Support Vector Machines for 3D Object Recognition

Massimiliano Pontil and Alessandro Verri

Abstract—Support Vector Machines (SVMs) have been recently proposed as a new technique for pattern recognition. Intuitively, given a set of points which belong to either of two classes, a linear SVM finds the hyperplane leaving the largest possible fraction of points of the same class on the same side, while maximizing the distance of either class from the hyperplane. The hyperplane is determined by a subset of the points of the two classes, named *support vectors*, and has a number of interesting theoretical properties. In this paper, we use linear SVMs for 3D object recognition. We illustrate the potential of SVMs on a database of 7,200 images of 100 different objects. The proposed system does not require feature extraction and performs recognition on images regarded as points of a space of high dimension without estimating pose. The excellent recognition rates achieved in all the performed experiments indicate that SVMs are well-suited for aspect-based recognition.

Index Terms—Support vector machines, optimal separating hyperplane, appearance-based object recognition, pattern recognition.

1 Introduction

Support Vector Machines (SVM) have been recently proposed as a very effective method for general purpose pattern recognition [16], [5]. Intuitively, given a set of points which belong to either of two classes, a SVM finds the hyperplane leaving the largest possible fraction of points of the same class on the same side, while maximizing the distance of either class from the hyperplane. According to [16], given fixed but unknown probability distributions, this hyperplane—called Optimal Separating Hyperplane (OSH)—minimizes the risk of misclassifying not only the examples in the training set but also the *yet-to-be-seen* examples of the test set.

The aim of this paper is to illustrate the potential of SVMs on a computer vision problem, the recognition of 3D objects from single images. To this purpose, an aspect-based method for the recognition of 3D objects which makes use of SVMs is described. In the last few years, aspect-based recognition strategies have received increasing attention from both the psychophysical [12], [6] and computer vision [11], [1], [13], [4], [10], [8], [17] communities. Although not naturally tolerant to occlusions, aspect-based recognition strategies appear to be well-suited for the solution of recognition problems in which geometric models of the viewed objects can be difficult, if not impossible, to obtain. To emphasize the generality of SVMs for classification tasks, the method (*a*) does not require feature extraction or data reduction, and (*b*) can be applied directly to images regarded as points of an N-dimensional object space, *without* estimating pose. The high dimensionality of the object space makes OSHs very effective decision surfaces, while the recognition stage is reduced to deciding on which side of an OSH lies a given point in object space.

The proposed method has been tested on the COIL database consisting of 7,200 images of 100 objects. Half of the images were used as training examples, the remaining half as test images. We discarded color information and tested the method on the remaining images corrupted by synthetically generated noise, bias, and occlusions. The remarkable recognition rates achieved in all the performed experiments indicate that SVMs are well-suited for aspect-based recognition. Comparisons with other pattern recognition methods, like perceptrons, show that the proposed method is far more robust in the presence of noise.

The paper is organized as follows. In Section 2, we review the basic facts of the theory of SVM. Section 3 discusses the implementation of SVMs adopted throughout this paper and describes the main features of the proposed recognition system. The obtained experimental results are illustrated in Section 4. Finally, Section 5 summarizes the conclusions that can be drawn from the presented research.

2 Theoretical Overview

In this section, we recall the basic notions of the theory of SVMs [16], [5]. We start with the simple case of linearly separable sets. Then we define the concept of support vectors and deal with the more general nonseparable case. Finally, we list the main properties of SVMs. Since we have only used linear SVMs we do not cover the generalization of the theory to the case of nonlinear separating surfaces.

2.1 Optimal Separating Hyperplane

In what follows, we assume we are given a set S of points $\mathbf{x}_i \in \mathbb{R}^n$ with $i = 1, 2, \ldots, N$. Each point $\mathbf{x}_i$ belongs to either of two classes and thus is given a label $y_i \in \{-1, 1\}$. The goal is to establish the equation of a hyperplane that divides S

- *M. Pontil is with the Center for Biological and Computational Learning, Massachusetts Institute of Technology, 45 Carleton Street E25-206, Cambridge, MA 02142. E-mail: pontil@ai.mit.edu.*
- *A. Verri is with INFM—Dipartimento di Informatica e Scienze dell'Informazione, Universitá di Genova, Via Dodecaneso 35, 16146 Genova, Italy. E-mail: verri@disi.unige.it.*

Manuscript received 11 June 1997; revised 16 Apr. 1998. Recommended for acceptance by P. Flynn.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number 106755.

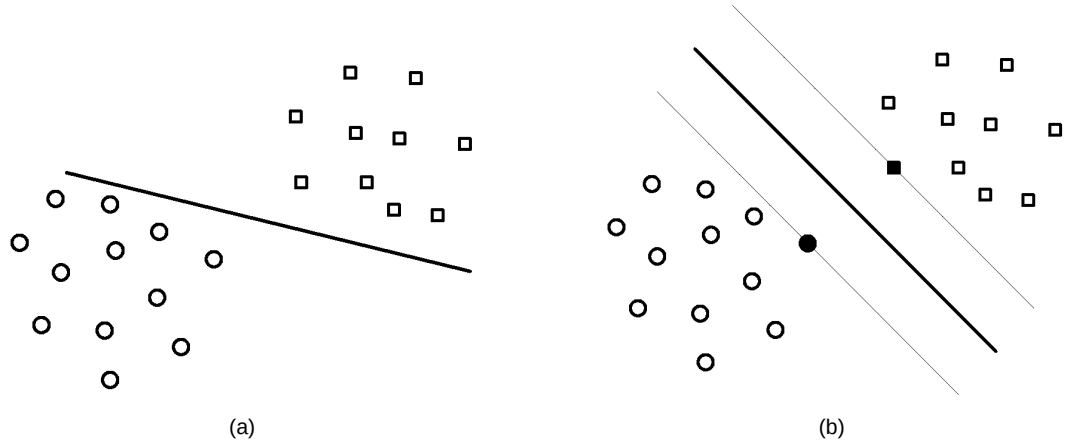


Fig. 1. Separating hyperplane (a) and OSH (b). The dashed lines in (b) identify the margin.

leaving all the points of the same class on the same side while maximizing the distance between the two classes and the hyperplane. To this purpose we need some preliminary definitions.

DEFINITION 1. *The set S is* linearly separable *if there exist* $\mathbf{w} \in \mathbb{R}^n$ *and* $b \in \mathbb{R}$ *such that*

$$y_i (\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1, \quad (1)$$

for $i = 1, 2, \ldots, N$.

The pair $(\mathbf{w}, b)$ defines a hyperplane of equation

$$\mathbf{w} \cdot \mathbf{x} + b = 0$$

named separating hyperplane (see Fig. 1a). If we denote with w the norm of $\mathbf{w}$, the signed distance d_i of a point $\mathbf{x}_i$ from the separating hyperplane $(\mathbf{w}, b)$ is given by

$$d_i = \frac{\mathbf{w} \cdot \mathbf{x}_i + b}{w} \quad (2)$$

with w norm of $\mathbf{w}$. Combining inequality (1) and (2), for all $x_i \in S$ we have

$$y_i d_i \geq \frac{1}{w}. \quad (3)$$

Therefore, $1/w$ is the lower bound on the distance between the points $\mathbf{x}_i$ and the separating hyperplane $(\mathbf{w}, b)$.

We now need to establish a one-to-one correspondence between separating hyperplanes and their parametric representation.

DEFINITION 2. *Given a separating hyperplane* $(\mathbf{w}, b)$ *for the linearly separable set S, the* canonical representation *of the separating hyperplane is obtained by rescaling the pair* $(\mathbf{w}, b)$ *into the pair* $(\mathbf{w}', b')$ *in such a way that the distance of the closest point, say* $\mathbf{x}_j$, *equals* $1/w'$.

Through this definition we have that

$$\min_{\mathbf{x}_i \in S} \left\{ y_i \left(\mathbf{w}' \cdot \mathbf{x}_i + b' \right) \right\} = 1.$$

Consequently, for a separating hyperplane in the canonical representation, the bound in inequality (3) is tight. In what follows we will assume that a separating hyperplane is always given a canonical representation and thus write $(\mathbf{w}, b)$ instead of $(\mathbf{w}', b')$. We are now in a position to define the notion of OSH.

DEFINITION 3. *Given a linearly separable set S, the* optimal separating hyperplane *is the separating hyperplane for which the distance of the closest point of S is maximum.*

Since the distance of the closest point equals $1/w$, the OSH can be regarded as the solution of the problem of maximizing $1/w$ subject to the constraint (1), or

Problem **P1**

Minimize $\frac{1}{2} \mathbf{w} \cdot \mathbf{w}$

subject to $y_i (\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1, \; i = 1, 2, \ldots, N$

Note that the parameter b enters in the constraints but not in the function to be minimized. The quantity $2/w$, the lower bound of the minimum distance between points of different classes, is named *margin*. Hence, the OSH can also be seen as the separating hyperplane which maximizes the margin (see Fig. 1b). From the quantitative viewpoint, the margin can be thought of as a measure of the difficulty of the problem (the smaller the margin the more difficult the problem). We now study the properties of the solution of the Problem **P1**.

2.2 Support Vectors

Problem **P1** is usually solved by means of the classical method of Lagrange multipliers. In order to understand the concept of support vectors it is necessary to go briefly through this method. For more details and a thorough review of the method see [2].

If we denote with $\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_N)$ the N nonnegative Lagrange multipliers associated with the constraints (1), the solution to Problem **P1** is equivalent to determining the *saddle point* of the function

$$L = \frac{1}{2} \mathbf{w} \cdot \mathbf{w} - \sum_{i=1}^{N} \alpha_i \left\{ y_i \left(\mathbf{w} \cdot \mathbf{x}_i + b \right) - 1 \right\} \quad (4)$$

with $L = L(\mathbf{w}, b, \alpha)$. At the saddle point, L has a minimum for $\mathbf{w} = \overline{\mathbf{w}}$ and $b = \overline{b}$ and a maximum for $\alpha = \overline{\alpha}$, and thus we can write

$$\frac{\partial L}{\partial b} = \sum_{i=1}^{N} y_i \alpha_i = 0 \quad (5)$$

$$\frac{\partial L}{\partial \mathbf{w}} = \mathbf{w} - \sum_{i=1}^{N} \alpha_i y_i \mathbf{x}_i = 0 \tag{6}$$

with

$$\frac{\partial L}{\partial \mathbf{w}} = \left(\frac{\partial L}{\partial w_1}, \frac{\partial L}{\partial w_2}, \ldots, \frac{\partial L}{\partial w_n} \right).$$

Substituting (5) and (6) into the right-hand side of (4), we see that Problem **P1** reduces to the maximization of the function

$$\mathcal{L}(\alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{N} \alpha_i \alpha_j y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j \tag{7}$$

subject to the constraint (5) with $\boldsymbol{\alpha} \geq 0$ (in what follows $\boldsymbol{\alpha} \geq 0$ means $\alpha_i \geq 0$ for every component α_i of any vector $\boldsymbol{\alpha}$). This new problem is called *dual problem* and can be formulated as

Problem **P2**
Minimize $-\frac{1}{2} \boldsymbol{\alpha}^{\mathrm{T}} D \boldsymbol{\alpha} + \sum \alpha_i$
subject to $\sum y_i \alpha_i = 0$
$\boldsymbol{\alpha} \geq 0,$

where both sums are for $i = 1, 2, \ldots, N$, and D is an $N \times N$ matrix such that

$$D_{ij} = y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j. \tag{8}$$

As for the pair $(\overline{\mathbf{w}}, \overline{b})$, from (6) it follows that

$$\overline{\mathbf{w}} = \sum_{i=1}^{N} \overline{\alpha}_i y_i \mathbf{x}_i,$$

while $\overline{b}$ can be determined from $\overline{\boldsymbol{\alpha}}$, solution of the dual problem, and from the Kühn-Tucker conditions

$$\overline{\alpha}_i \left(y_i \left(\overline{\mathbf{w}} \cdot \mathbf{x}_i + \overline{b} \right) - 1 \right) = 0, \qquad i = 1, 2, \ldots, N. \tag{9}$$

Note that the only $\overline{\alpha}_i$ that can be nonzero in (6) are those for which the constraints (1) are satisfied with the equality sign. This has an important consequence. Since most of the $\overline{\alpha}_i$ are usually null, the vector $\overline{\mathbf{w}}$ is a linear combination of a relatively small percentage of the points $\mathbf{x}_i$. These points are termed *support vectors* because they are the closest points from the OSH and the only points of S needed to determine the OSH (see Fig. 1b).

Given a support vector $\mathbf{x}_j$, the parameter $\overline{b}$ can be obtained from the corresponding Kühn-Tucker condition as

$$\overline{b} = y_j - \overline{\mathbf{w}} \cdot \mathbf{x}_j. \tag{10}$$

The problem of classifying a new data point $\mathbf{x}$ is now simply solved by looking at the sign of

$$\overline{\mathbf{w}} \cdot \mathbf{x} + \overline{b}.$$

Therefore, the support vectors condense all the information contained in the training set S which is needed to classify new data points.

2.3 Linearly Nonseparable Case

If the set S is not linearly separable or one simply ignores whether or not the set S is linearly separable, the problem of searching for an OSH is meaningless (there may be no separating hyperplane to start with). Fortunately, the previous analysis can be generalized by introducing N nonnegative variables $\boldsymbol{\xi} = (\xi_1, \xi_2, \ldots, \xi_N)$ such that

$$y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1 - \xi_i, \quad i = 1, 2, \ldots, N. \tag{11}$$

The purpose of the variables ξ_i is to allow for a small number of misclassified points. If the point $\mathbf{x}_i$ satisfies inequality (1), then ξ_i is null and (11) reduces to (1). Instead, if the point $\mathbf{x}_i$ does not satisfy inequality (1), the extraterm $-\xi_i$ is added to the right hand side of (1) to obtain inequality (11). The generalized OSH is then regarded as the solution to

Problem **P3**
Minimize $-\frac{1}{2} \mathbf{w} \cdot \mathbf{w} + C \sum \xi_i$
subject to $y_i (\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1 - \xi_i \quad i = 1, 2, \ldots, N$
$\boldsymbol{\xi} \geq 0.$

The purpose of the extraterm $C\sum\xi_i$, where the sum is for $i = 1, 2, \ldots, N$, is to keep under control the number of misclassified points. Note that this term leads to a more robust solution, in the statistical sense, than the intuitively more appealing term $C \sum \xi_i^2$. In other words, the term $C\sum\xi_i$ makes the OSH less sensitive to the presence of outliers in the training set. The parameter C can be regarded as a regularization parameter. The OSH tends to maximize the minimum distance $1/w$ for small C, and minimize the number of misclassified points for large C. For intermediate values of C the solution of problem **P3** trades errors for a larger margin.

In analogy with what was done for the separable case, Problem **P3** can be transformed into the *dual*

Problem **P4**
Maximize $-\frac{1}{2} \boldsymbol{\alpha}^{\mathrm{T}} D \boldsymbol{\alpha} + \sum \alpha_i$
subject to $\sum y_i \alpha_i = 0$
$0 \leq \alpha_i \leq C, \qquad i = 1, 2, \ldots, N$

with D the same $N \times N$ matrix of the separable case. Note that the dimension of **P4** is given by the size of the training set, while the dimension of the input space gives the rank of D. From the constraints of Problem **P4** it follows that if C is sufficiently large and the set S linearly separable, Problem **P4** reduces to **P2**.

As for the pair $(\overline{\mathbf{w}}, \overline{b})$, it is easy to find

$$\overline{\mathbf{w}} = \sum_{i=1}^{N} \overline{\alpha}_i y_i \mathbf{x}_i,$$

while $\overline{b}$ can again be determined from $\overline{\boldsymbol{\alpha}}$, solution of the dual problem **P4**, and from the new Kuhn-Tucker conditions

$$\overline{\alpha}_i \left(y_i \left(\overline{\mathbf{w}} \cdot \mathbf{x}_i + \overline{b} \right) - 1 + \overline{\xi}_i \right) = 0 \tag{12}$$

$$\left(C - \overline{\alpha}_i \right) \overline{\xi}_i = 0 \tag{13}$$

where the $\overline{\xi}_i$ are the values of the ξ_i at the saddle point. Similarly to the separable case, the points $\mathbf{x}_i$ for which $\overline{\alpha}_i > 0$ are termed *support vectors*. The main difference is

that here we have to distinguish between the support vectors for which $\overline{\alpha}_i < C$ and those for which $\overline{\alpha}_i = C$. In the first case, from condition (13) it follows that $\overline{\xi}_i = 0$, and hence, from condition (12), that the support vectors lie at a distance $1/\overline{w}$ from the OSH. These support vectors are termed *margin vectors*. The support vectors for which $\overline{\alpha}_i = C$, instead, are misclassified points (if $\xi_i > 1$), points correctly classified but closer than $1/\overline{w}$ from the OSH (if $0 < \xi \leq 1$), or margin vectors (if $\xi_i = 0$). Neglecting this last rare (and degenerate) occurrence, we refer to all the support vectors for which $\alpha_i = C$ as *errors*. All the points that are not support vectors are correctly classified and lie outside the margin strip.

Finally, we point out that the entire construction can also be extended rather naturally to include nonlinear decision surfaces [16]. However, since for the research described in this paper this extension was not needed, we do not further discuss this issue here.

2.4 Mathematical Properties

We conclude this section listing the three main mathematical properties of SVMs.

The first property distinguishes SVMs from previous nonparametric techniques, like nearest-neighbors or neural networks. Typical pattern recognition methods are based on the minimization of the *empirical risk*, that is on the attempt to minimize the misclassification errors on the training set. Instead, SVMs minimize the *structural risk*, that is the probability of misclassifying a previously unseen data point drawn randomly from a fixed but unknown probability distribution. If the VC-dimension [15] of the family of decision surfaces is known, the theory of SVMs provides an upper bound for the probability of misclassification of the test set for any possible probability distributions of the data points [16].

Second, SVMs condense all the information contained in the training set relevant to classification in the support vectors. This (*a*) reduces the size of the training set identifying the most important points, and (*b*) makes it possible to efficiently perform classification.

Third, SVMs are quite naturally designed to perform classification in high dimensional spaces, even in the presence of a relatively small number of data points. The real limitation to the employment of SVMs in high dimensional spaces is computational efficiency. In practice, for each particular problem a trade-off between computational efficiency and success rate must be established.

3 The Recognition System

We now describe the recognition system we devised to assess the potential of the theory. We first review the implementation developed for determining the support vectors and the associated OSH given a training set of points belonging to either of two classes.

3.1 Implementation

In Section 2, we have seen that the problem of determining the OSH reduces to Problem **P4**, a typical problem of quadratic programming. The vast literature of nonlinear programming covers a multitude of problems of quadratic programming and provides a plethora of methods for their solution. Our implementation makes use of the equivalence between quadratic programming problems and *Linear Complementary* Problems (LCPs) and is based on the *Complementary Pivoting Algorithm* (CPA), a classical algorithm able to solve LCPs [2].

Since the spatial complexity of CPA goes with the square of the number of examples, the algorithm cannot deal efficiently with much more than a few hundreds of examples. This has not been a fundamental issue for the research described in this paper, but for problems of larger size one has definitely to resort to more sophisticated techniques [9].

3.2 Recognition Stages

We have developed a recognition system based on three stages:

1) Preprocessing
2) Training set formation
3) System testing

Before describing these three stages in detail, we first illustrate the main features of the COIL database.

3.2.1 The COIL Images

The COIL (Columbia Object Image Library) database we used consists of 7,200 images of 100 objects (72 views for each of the 100 objects). The COIL images are color images (24 bits for each of the RGB channels) of 128×128 pixels. The 7,200 images were downloaded via anonymous ftp from the site `www.cs.columbia.edu`. As explained in detail in [8], the objects are positioned in the center of a turntable and observed from a fixed viewpoint. For each object, the turntable is rotated of 5° per image. Fig. 2 and Fig. 3 show a selection of the objects in the database and one every three views (or images) of a specific object, respectively. In all COIL images we inspected, the object region appears to be re-sampled so that the larger of the two object dimensions fits the image size. Consequently, the apparent size of an object may change considerably from image to image (see Fig. 3, for example), especially for the objects which are not symmetric with respect to the turntable axis.

3.2.2 Preprocessing

In the preprocessing stage each COIL image $I = (R, G, B)$ was first transformed into a gray-level image E through the conversion formula

$$E = .31R + .59G + .10B,$$

rescaling the obtained gray-level in the range between zero and 255. Then, the image spatial resolution was reduced to 32×32 by averaging the gray levels over 4×4 pixel patches. The aim and relevance of this last transformation will be discussed in the experimental section. Unless stated otherwise, it can thus be assumed that each COIL image is transformed into an eight-bit vector of $32 \times 32 = 1{,}024$ components.

Fig. 2. Images of 32 objects of the COIL database.

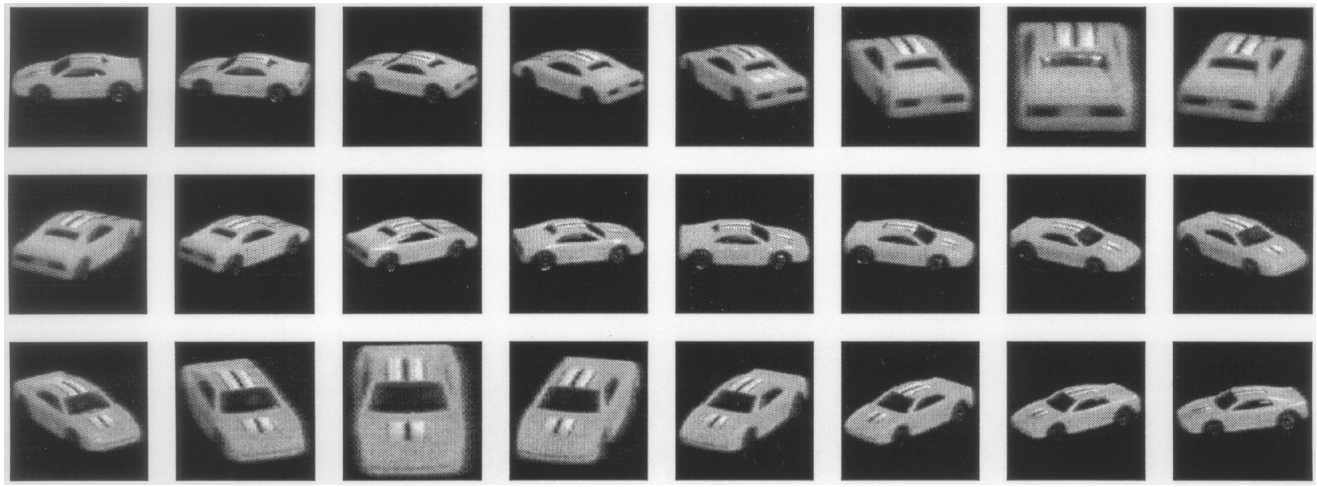


Fig. 3. Twenty-four of the 72 images of a COIL object.

3.2.3 Forming the Training Set

Since the final goal is to build an aspect-based recognition system, the training sets used in each experiment consist of 36 images (one every 10°) for each object.

For each experiment, a subset σ of the 100 objects (typically chosen randomly) has been considered. Then, the OSHs associated to each pair of objects i and j in σ were computed, the support vectors identified, and the obtained parameters, $\mathbf{w}(i, j)$ and $b(i, j)$, stored in a file. In all cases, because of the high dimensionality of object space compared to the small number of examples, we have never come across errors in the classification of the training sets.

The images corresponding to some of the support vectors for a specific pair of objects are shown in Fig. 4. These images can be thought of as representative views (or aspects) of the objects to be recognized. Notice that unlike the case of aspect graph approaches, these views do not characterize an object per se, but one object relatively to another.

As a final remark, we observe that for each object pair we have typically found a number of support vectors ranging from 1/3 to 2/3 of the 72 training images. This relatively large fraction of support vectors can again be explained by the high dimensionality of the object space combined with the small number of examples.

3.2.4 System Testing

Given a subset σ of the 100 objects and the associated training set of 36 images for each object in σ, the test set consists of the remaining 36 images for each object in σ. Recognition was performed following the rules of a tennis tournament. Each object in σ is regarded as a *player*, and in each *match* the system temporarily classifies an image of the test set according to the OSH relative to the pair of players involved in the match. If in a certain match the players are

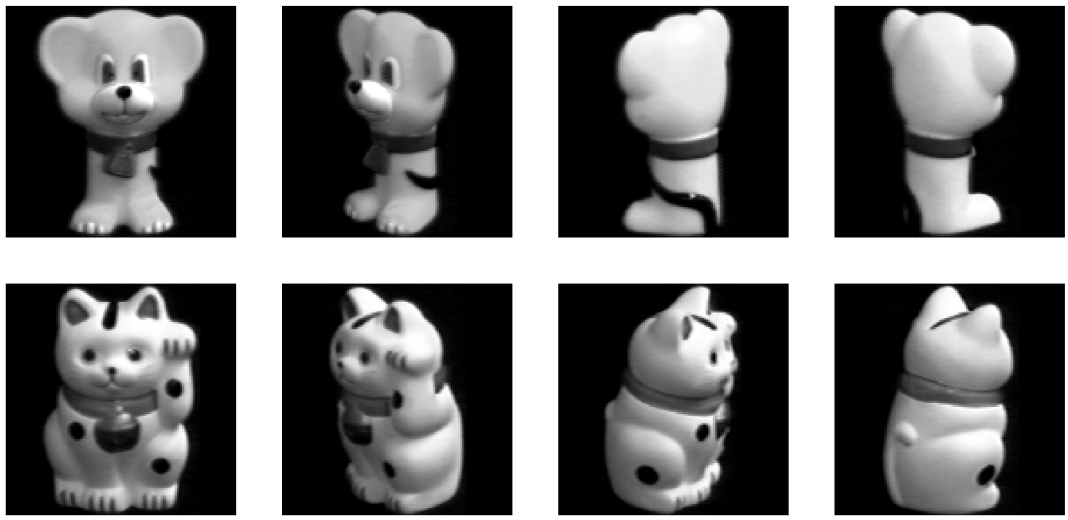
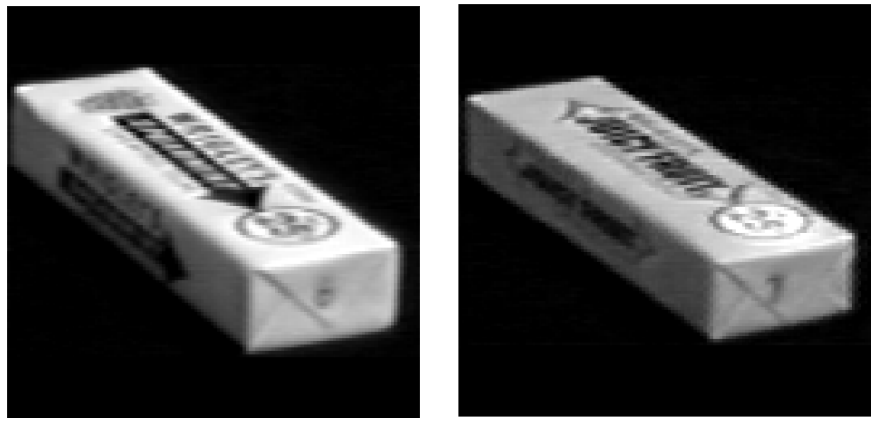


Fig. 4. Eight of the support vectors for a specific object pair.



(a) (b)

Fig. 5. The only misclassified image (a) and corresponding erroneously recognized object (b).

objects i and j, the system classifies the viewed object of image $\mathbf{x}$ as object i or j depending on the sign of

$$\mathbf{w}(i, j) \cdot \mathbf{x} + b(i, j).$$

If, for simplicity, we assume there are 2^K players, the first round 2^{K-1} matches are played and the 2^{K-1} losing players are out. The 2^{K-1} match winners advance to the second round. The $(K-1)$th round is the final between the only two players that won all the previous matches. This procedure requires $2^K - 1$ classifications. Note that the system recognizes object identity without estimating pose.

We are now ready to present the obtained experimental results.

4 Experimental Results

In order to verify the effectiveness and robustness of the proposed recognition system, we performed experiments on the COIL images under increasingly difficult conditions. We first considered the images exactly as extracted from the COIL database, then we added pixel-wise random noise, bias in the registration of the test images, and a combination of the two. Finally, we studied the sensitivity of the system to moderate amount of occlusions and compared the obtained results against a simple perceptron.

4.1 Plain Images

We first tested the proposed recognition system on sets of 32 of the 100 COIL objects. As already mentioned, the training sets consisted of 36 images (one every 10°) for each of the 32 objects and the test sets of the remaining 36 images for each object. For all the 20 random choices of 32 of the 100 objects we tried, the system reached perfect score. Therefore, we decided to select by hand the 32 objects *more difficult* to recognize (i.e., the groups of objects which appeared more similar). By doing so the system finally mistook a view of a packet of chewing gum (see Fig. 5a) for another very similar packet of chewing gum (a view of which is shown in Fig. 5b).

To gain a better understanding of how an SVM actually perform recognition, it may be useful to look at the relative weights of the components of $\mathbf{w}$. A gray-valued encoded representation of the absolute value of the components of $\mathbf{w}$ relative to the OSH of the two objects of Fig. 4

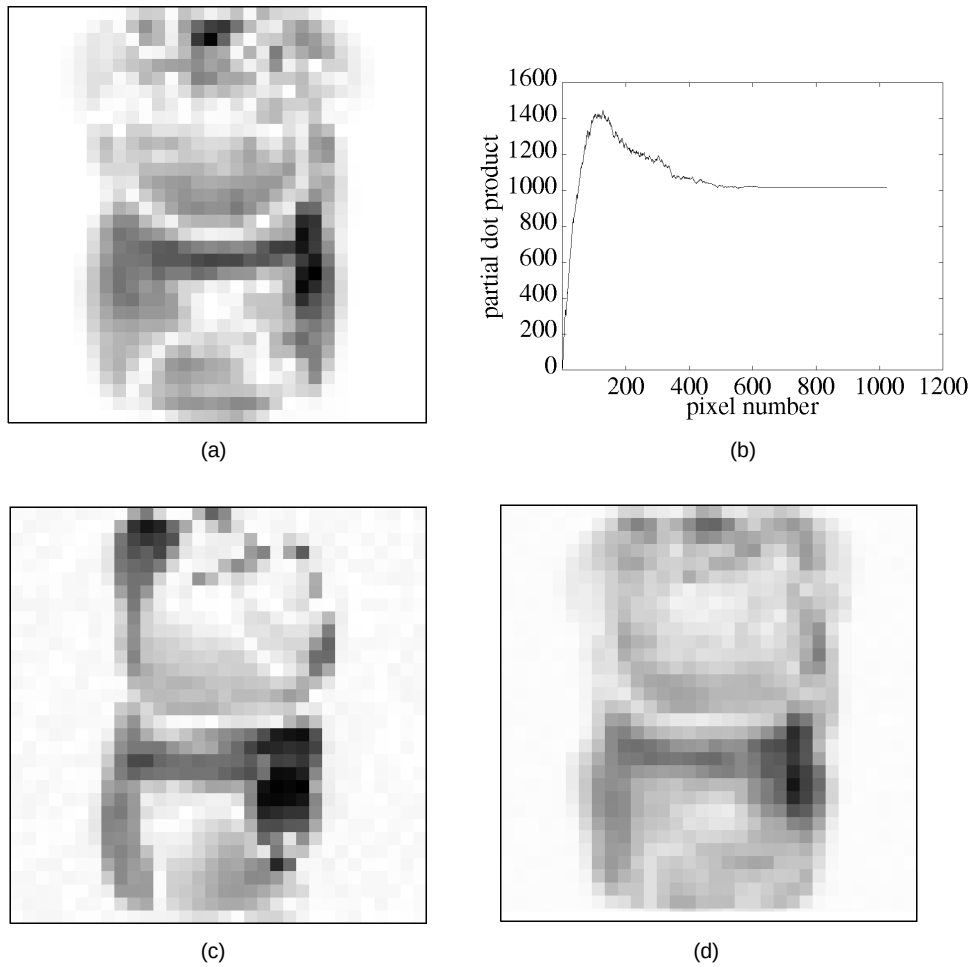


Fig. 6. Image of **w** for a SVM (a). Relative weights of the components of **w** (b). (See text for details.) Images of **w** for a perceptron (c) and an average of 10 perceptrons (d).

is displayed in Fig. 6a (the darker a point, the higher the corresponding **w** component). Note that the background is essentially irrelevant, while the larger components (in absolute value) can be found in the central portion of the *image*. Interestingly, the image of Fig. 6a resembles the visual appearance of both the "dog" and "cat" of Fig. 4. The graph of Fig. 6b shows the convergence of $\sum w_i x_i$ to the dot product $\mathbf{w} \cdot \mathbf{x}$ for one of the "cat" image, with the components w_i sorted in decreasing order. From the graph, it clearly follows that less than half of the 1,024 components are all what is needed to reach almost perfect convergence, while a reasonably good approximation is already obtained using only the first 100 larger components. Qualitatively and with a few exceptions corresponding to very similar object pairs, the graph of Fig. 6b is typical.

In conclusion, as reported in Table 1, the proposed method performs recognition with excellent percentages of success even in the presence of very similar objects. It is worthwhile noticing that while the recognition time is practically negligible (requiring the evaluation of 31 dot products), the training stage (in which all the $32 \times 31/2 = 496$ OSHs must be determined) takes about 15 minutes on a SPARC10 workstation.

TABLE 1
AVERAGE ERROR RATES (A.E.R.) FOR IMAGES FROM THE COIL DATABASE

spat. res.	a.e.r.
32×32	0.03%

4.2 Noise Corrupted Images

In order to assess the robustness of the method, we added zero mean random noise uniformly distributed in the interval $[-n, +n]$ to the gray value of each pixel. Restricting the analysis to the 32 objects more difficult to recognize, the system performed equally well for noise up to ±100 gray levels and degrades gracefully for higher amount of noise (see Table 2, middle column). Notice that since the image gray levels are bound to be between zero and 255, adding noise up to $\pm n$ gray levels means that the noisy images were actually rescaled within the range [0, 255]. Some of the noise corrupted images from which the system has been able to identify the viewed object are displayed in Fig. 7.

By inspection of the obtained results, we note that most of the errors were due the three chewing gum packets of Fig. 2. which become practically indistinguishable as the noise increases. The same experiments leaving out two of

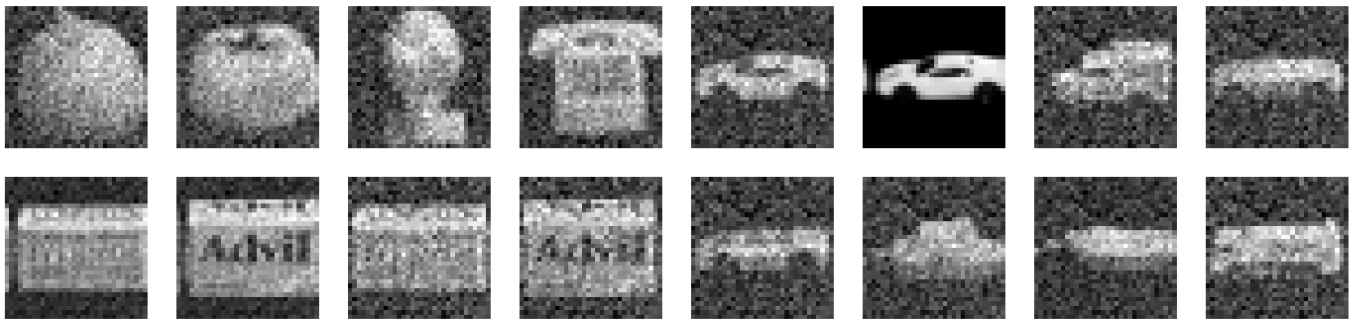


Fig. 7. Sixteen images of the test sets synthetically corrupted by noise and spatially misregistered, but correctly classified by the proposed system.

the three packets produced much better performances (see Table 2, right column). It must be said that the very good statistics of Table 2 are partly due to the "filtering effects" of the reduction of the image size from 128×128 to 32×32 pixels obtained by spatial averaging.

In order to study the behavior of SVMs for different dimensionality of the input data, we perform the same experiments by using images at different spatial resolution. The obtained results are summarized in Table 3 which shows the error rates obtained with spatial resolution ranging from 8×8 to 128×128 (the maximum resolution available) and noise uniformly distributed in the interval [–100, 100] gray levels. Better results, not reported here, were obtained by low-pass filtering the input vectors regarded as "images"). Intuitively, this can be explained in terms of the local correlations established by the filtering step. Note that, as expected from the theoretical considerations, the recognition rates increase with the spatial resolution. However, since also the recognition time increases with spatial resolution, the spatial resolution of 32×32 pixels appeared to be the best trade off between recognition rate and efficient classification. This is why, in what follows, we only report results relative to the case of 32×32 images.

From the obtained experimental results, it can easily be inferred that the method achieves very good recognition rates even in the presence of large amount of noise.

4.3 Shifted Images

In a third series of experiments, we checked the dependence of the system on the precision with which the available images are spatially registered. We thus shifted each image of the test set by n pixels in the horizontal direction (in a wrapping around style) and repeated the same recognition experiments of this section on the set of the 32 more difficult objects. As can be appreciated from Table 4, the system performs equally well for small shifts (n = 3, 5) and degrades gracefully for larger displacements (n = 7, 10).

We have obtained very similar results, reported in Table 5, when combining noise and shifts. Here again it must be noted that the quality of the results is partly due to the "filtering effects" of the preprocessing step.

It is concluded that the spatial registration of images is important but that spatial redundancy makes it possible to achieve very good performances even in the presence of a combination of additive noise and moderate amount of misregistration between the model image and the current image.

TABLE 2
ERROR RATES (E.R.) FOR COIL IMAGES CORRUPTED BY NOISE

Noise	e.r. (32 objs)	e.r. (30 objs)
±25	0.3%	0.0%
±50	0.8%	0.1%
±75	1.1%	0.2%
±100	1.6%	0.2%
±150	2.7%	0.7%
±200	6.2%	1.8%
±250	11.0%	5.8%

The noise is in gray levels (see text).

TABLE 3
ERROR RATES (E.R.) FOR COIL IMAGES CORRUPTED BY NOISE UNIFORMLY DISTRIBUTED IN THE INTERVAL [–100, 100] AT DIFFERENT SPATIAL RESOLUTIONS

spat. res.	e.r. (32 objs)	e.r. (30 objs)
8×8	2.8%	0.5%
16×16	2.1%	0.3%
32×32	1.6%	0.2%
64×64	0.9%	0.1%
128×128	0.3%	0.0%

TABLE 4
ERROR RATES (E.R.) FOR SHIFTED COIL IMAGES (SHIFTS ARE IN PIXEL UNITS)

shift	e.r. (32 objs)	e.r. (30 objs)
3	0.6%	0.1%
5	2.0%	0.8%
7	6.7%	4.8%
10	18.6%	12.5%

TABLE 5
ERROR RATES (E.R.) IN THE PRESENCE OF BOTH NOISE (IN GRAY LEVELS) AND SHIFTS (IN PIXEL UNITS)

shift	noise	e.r. (32 objs)	e.r. (30 objs)
3	±25	0.6%	0.1%
3	±50	0.8%	0.1%
3	±100	1.8%	0.2%
3	±150	3.0%	0.5%
5	±25	2.1%	0.6%
5	±50	2.7%	0.8%
5	±100	4.1%	1.3%
5	±150	7.3%	2.7%

TABLE 6
ERROR RATES (E.R.) FOR COIL IMAGES OCCLUDED BY A RANDOMLY PLACED K × K WINDOW OF UNIFORMLY DISTRIBUTED RANDOM NOISE

k	e.r. (32 objs)	e.r. (30 objs)
4	0.7%	0.4%
6	2.0%	1.2%
8	5.7%	4.3%
10	12.7%	10.8%

TABLE 7
ERROR RATES (E.R.) FOR COIL IMAGES IN WHICH N COLUMNS AND M ROWS (RANDOMLY SELECTED) WERE REPLACED BY UNIFORMLY DISTRIBUTED RANDOM NOISE

n	m	e.r. (32 objs)	e.r. (30 objs)
1	1	2.1%	1.3%
1	2	3.2%	1.9%
2	1	4.5%	2.8%
2	2	6.1%	3.2%

4.4 Occlusions

In order to verify the robustness of the system against occlusions, we performed two more series of experiments. In the first series we randomly selected a subwindow in the test images and assigned a random value between zero and 255 to the pixels inside the subwindow. The obtained error rates are summarized in Table 6. In the second experiment, we randomly selected *n* columns and *m* rows in the rescaled images and assigned a random value to the corresponding pixels. The obtained error rates are summarized in Table 7.

Some of the images from which the system was able to identify partially occluded objects are displayed in Fig. 8. Comparing the results in Tables 6 and 7, it is evident that the occlusion concentrated in a subwindow of the image poses more problems. In both cases, however, we conclude that the system appears to tolerate small amounts of occlusion.

4.5 Comparison With Perceptrons

In order to gain a better understanding of the relevance of the obtained results, we run a few experiments using perceptrons instead of SVMs. We considered a subset formed by two objects (the first two toy cars in Fig. 2, formed the training set of 72 images, and run recognition experiments by adding uniformly distributed random noise to the test set (also consisting of 72 images). The obtained results are summarized in Table 8.

TABLE 8
AVERAGE ERROR RATES (A.E.R.) OBTAINED BY AVERAGING THE A.E.R. OF 10 SINGLE PERCEPTRONS (SP) AND BY COMPUTING THE AVERAGE PERCEPTRON (AP) COMPARED WITH THE CORRESPONDING A.E.R. OF SVMS IN THE PRESENCE OF NOISE (SEE TEXT)

noise	SP a.e.r.	AP a.e.r.	SVM
±50	2.8%	1.7%	0.0%
±100	7.1%	4.3%	0.0%
±150	16.1%	9.6%	0.0%
±200	25.4%	14.8%	0.0%
±250	29.3%	20.2%	0.0%
±300	33.3%	25.4%	4.2%

The second column of Table 8 gives the average of the results obtained with ten different perceptrons (corresponding to 10 different random choices of the initial weights). The poor performance of perceptrons can be easily explained in terms of the margin associated to the separating hyperplane of each perceptron as opposed to the SVM margin. In this example, the perceptron margin is between two and 10 times smaller than the SVM margin. This means that both SVMs and perceptrons separate exactly the training set, but that the perceptron margin makes it difficult to classify correctly novel images in the presence of noise. Intuitively, this fact can be explained by thinking of noise perturbation as a motion in object space: if the margin is too small, even a slight perturbation can bring a point across the separating hyperplane (see Fig. 1). For the sake of comparison, the "image" of the normal vector associated with one of the perceptrons is displayed in Fig. 4c. The normal vector obtained by averaging the 10 computed perceptron, instead, is shown in Fig. 4d. While Figs. 4a and 4c are clearly different, Figs. 4a and 4d are qualitatively similar. However, quantitative analysis shows that even the average perceptron, if compared with the OSH, is still quite an ineffective classifier (Table 8, third and fourth columns).

5 DISCUSSION

In this final section we compare our results with the work of [8], discuss a few issues arising from our analysis, and summarize the obtained results.

5.1 Comparison With Previous Work

The images of the COIL database have been originally used by Murase and Nayar as a benchmark for testing their appearance-based recognition system, based on the notion of parametric eigenspace [8]. Our results seem to compare favorably with respect to the results reported in [8] especially in terms of computational cost and given the fact that we discarded color information. Note that SVMs allow for the construction of training sets of much smaller size than

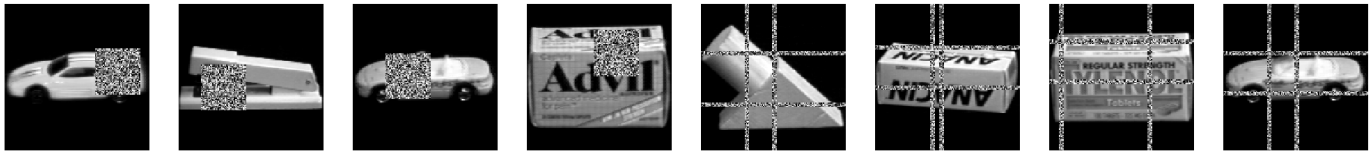


Fig. 8. Eight partially occluded objects correctly classified by the system.

the training sets of [8]. Unlike Murase and Nayar's, however, our method does not identify object's pose.

It would be interesting to compare our method with the classification strategy suggested in [8] on the same data points. After the construction of parametric eigenspaces, Murase and Nayar classify an object computing the minimum of the distance between the point representative of the object and the manifold of each object in the database. A possibility could be the use of SVMs for this last stage.

5.2 Scalability

An important issue is how the proposed system *scale* with the number of views and objects. This point is also connected with the need of considering nonlinear SVMs when dealing with more difficult classification tasks.

If the number of objects is constant, more views under different geometric and illumination conditions eventually require the implementation of nonlinear SVMs. However, the structure of the system remains exactly the same. If more objects are added, instead, the proposed method runs out of memory space (the spatial complexity of the proposed system grows with the square of the number of objects). This is because the system computes an OSH for each object pair. An alternative strategy, of spatial complexity linear in the number of objects, was proposed by Vapnik and coworkers [5]. Instead of a tennis tournament, the OSHs separating each of the N objects from the remaining $N-1$ are first computed. Then, a test image is classified relatively to each of the N computed OSHs, and the final classification decided upon the outcome of the these N classifications. Although more appealing from the viewpoint of memory requirements, this method often leads to ambiguous classification. Empirical evidence indicates that our approach, which is never ambiguous, produces better and more reproducible results.

6 Conclusions

In this paper, we have assessed the potential of linear SVMs in the problem of recognizing 3D objects from a single view. As shown by the comparison with other techniques, it appears that SVMs can be effectively *trained* even if the number of examples is much lower than the dimensionality of the object space. This agrees with the theoretical expectation that can be derived by means of *VC*-dimension considerations [16]. The remarkably good results which we have reported indicate that SVMs are likely to be very useful for direct 3D object recognition.

Acknowledgments

This work has been partially supported by a grant from the Agenzia Spaziale Italiana and a national project funded by MURST.

References

[1] S. Akamatsu, T. Sasaki, H. Fukumachi, and Y. Suenaga, "A Robust Face Identification Scheme - KL Expansion of an Invariant Feature Space," *SPIE Proc. Intelligent Robts and Computer Vision X: Algorithms and Techniques*, vol. 1,607, pp. 71-84, 1991.

[2] M. Bazaraa and C.M. Shetty, *Nonlinear Programming*. New York, NY: John Wiley, 1979.

[3] V. Blanz, B. Scholkopf, H. Bulthoff, C. Burges, V.N. Vapnik, and T. Vetter, "Comparison of View–Based Object Recognition Algorithms Using Realistic 3D Models," *Proc of ICANN'96*, LNCS, vol. 1,112, pp. 251-256, 1996.

[4] R. Brunelli and T. Poggio, "Face Recognition: Features Versus Templates," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 15, pp. 1,042-1,052, 1993.

[5] C. Cortes and V.N. Vapnik, "Support Vector Network," *Machine Learning*, vol. 20, pp. 1-25, 1995.

[6] S. Edelman, H. Bulthoff, and D. Weinshall, "Stimulus Familiarity Determines Recognition Strategy for Novel 3-D Objects," *AI Memo*, no. 1,138, Cambridge, Mass.: Massachusetts Institute of Technology, 1989.

[7] D.P. Huttenlocher, G.A. Klanderman, and W.J. Rucklidge, "Comparing Images Using the Hausdorff Distance," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 15, pp. 850-863, 1993.

[8] H. Murase and S.K. Nayar, "Visual Learning and Recognition of 3-D Object From Appearance," *Int. J. Computer Vision*, vol. 14, pp. 5-24, 1995.

[9] E. Osuna, R. Freund, and F. Girosi, "Training Support Vector Machines: An Applications to Face Detection," *Proc. CVPR*, 1997.

[10] A. Pentland, B. Moghaddam, and T. Starner, "View-Based and Modular Eigenspaces for Face Recognition," *Proc. CVPR*, pp. 84-91, 1994.

[11] T. Poggio and S. Edelman, "A Network That Learns to Recognize Three-Dimensional Objects," *Nature*, vol. 343, pp. 263-266, 1990.

[12] M. Tarr and S. Pinker, "Mental Rotation and Orientation-Dependence in Shape Recognition," *Cognitive Psychology*, vol. 21, pp. 233-282, 1989.

[13] M. Turk and A. Pentland, "Eigenfaces for Recognition," *J. Cognitive Neuroscience*, vol. 3, pp. 71-86, 1991.

[14] B. Scholkopf, K. Sung, C. Burges, F. Girosi, P. Niyogi, T. Poggio, and V. Vapnik, "Comparing Support Vector Machines With Gaussian Kernels to Radial Basis Function Classifiers," *AI Memo*, no. 1599; *CBCL Paper*, no. 142. Cambridge, Mass.: Massachusetts Institute of Technology, 1996.

[15] V.N. Vapnik and A.J. Chervonenkis, "On the Uniform Convergence of Relative Frequencies of Events to Their Probabilities," *Theory Probability Appl.*, vol. 16, pp. 264-280, 1971.

[16] V.N. Vapnik, *The Nature of Statistical Learning Theory*. New York, NY: Springer-Verlag, 1995.

[17] J. Weng, "Cresceptron and SHOSLIF: Toward Comprehensive Visual Learning." S.K. Nayar and T. Poggio, eds. *Early Visual Learning*. Oxford Univ. Press, 1996.

Massimiliano Pontil received the Laurea and PhD in theoretical physics from the University of Genova in 1994 and 1998. In 1998, he visited the Center for Biological and Computational Learning at MIT. His main research interests are theoretical and practical aspects of learning techniques in pattern recognition and computer vision.

Alessandro Verri received the Laurea and PhD in theoretical physics from the University of Genova in 1984 and 1988. Since 1989, he has been Ricercatore at the University of Genova (first at the Physics Department and, since the fall of 1997, at the Department of Computer and Information Science). He has published approximately 50 papers on various aspects of visual computation in man and machines—including stereo, motion analysis, shape representation, and object recognition—and coauthored a textbook on computer vision with Dr. E. Trucco. He has been visiting fellow at MIT, International Computer Science Institute, INRIA/IRISA (Rennes), and Heriot-Watt University. His current interests range from the study of computational problems of pattern recognition and computer vision to the development of visual systems for industrial inspection and quality control.