SUPPORT VECTOR DATA DESCRIPTION BASED ON PCA FEATURES FOR FACE DETECTION

Verónica Vilaplana and Ferran Marqués

Image Processing Group, Universitat Politècnica de Catalunya
Jordi Girona, 1-3, 08034 Barcelona, SPAIN
phone: + (34)934011066, fax: + (34)934016447, email: {veronica,ferran}@gps.tsc.upc.edu
web: gps-tsc.upc.es/imatge

ABSTRACT

In this paper we deal with the problem of frontal face detection using Support Vector Data Description (SVDD) to characterize textural attributes of faces. The SVDD classifier relies on PCA features of face samples to obtain a decision boundary around the face data without using information of negative examples (outliers). We analyze the performance of the classifier for different dimensionalities of the feature space and for different selections of the SVDD parameters. Experimental results show that the SVDD can be adopted as an effective tool for making a face detector as a combination of multiple simple one-class classifiers.

1. INTRODUCTION

Face detection is usually approached, in pattern recognition terms, as a two-class classification problem. Two classes, face and no-face, are modeled using training samples of face and no-face patterns and a decision boundary between the two classes is inferred.

However, the two classes are not equally complex. Faces have a common structure with the same configuration of facial features, so it is natural to think of a face model. On the contrary, the no-face class is broader and richer. Although negative examples are abundant, those that are useful from a learning point of view are very difficult to define and characterize. One possible solution is to build a 'near-face' class model, using patterns obtained as misclassifications in early stages of the training phase [8]. In this case, an extremely large set of examples is needed in order to learn the task with the desired accuracy.

Our approach to face detection is as a one-class classification problem, where an object has to be classified as target object (face) or outlier (no-face). In one-class classification it is assumed that only information of the target class is available. In this context, we are developing a region-based face detection and segmentation technique (partially described in [4]) based on the combination of multiple one-class face classifiers.

Instead of looking at all possible pixel locations and all possible scales, the detection algorithm bases its analysis strategy on a reduced set of regions that represent the image content at different scales of resolution. For each candidate region, a set of simple classifiers that rely on different shape, color and texture attributes is evaluated. The outputs of the classifiers are combined into a final face likelihood.

In this paper we concentrate on textural attributes of faces. Statistical techniques based on texture, or 'appearance based methods', are widely adopted for face detection.

Moghadam and Pentland [6] and Sung and Poggio [8] make use of an eigenface method to model the face class using the 'distance in the feature space' (DIFS) and 'distance from the face space' (DFFS) criteria (see Section 2).

Osuna et al. [7] propose an SVM-based approach to frontal-view face detection. This method seeks to learn the boundary between face and no-face patterns. After learning, only examples of face and no-face patterns located on the boundary are selected to build the decision function.

One of the state of the art face detectors is the system presented by Viola-Jones [11]. Weak classifiers based on simple, local, Haar-like features are boosted into a single strong classifier. Strong classifiers are then combined in a coarse to fine cascade which allows background regions to be quickly discarded. The technique is very fast but requires a hard training with a large number of face and no-face patterns.

The techniques mentioned above detect faces by scanning the image at multiple scales and locations. The analysis is performed on each image sub-window. A complete discussion of other appearance based techniques and other approaches to face detection can be found in [12].

The goal of this paper is to analyze the use of the Support Vector Data Description classifier (SVDD) based on PCA features as an alternative to the texture classifier based on DIFF that has been extensively used in the literature and is currently implemented in our system. Sections 2 and 3 present the eigenface and SVDD formulation, respectively. Section 4 describes our experiments and results, and finally some conclusions are drawn in Section 5.

2. EIGENFACES

Given a collection of n by m pixel training images represented as vectors of size $N = nm$ in an N-dimensional space, Principal Component Analysis (PCA) defines a transformation from R^N to a lower dimensional space R^M, $M < N$, defined by $z = W^t(x - \quad)$, where is the sample mean.

The column vectors of W, $\{w_i\}_{i=1 \ldots M}$, are the orthonormal axes that capture most of the variance present in the data. These column vectors are the M eigenvectors of the data covariance matrix with largest eigenvalues.

When the PCA is applied to face images, the eigenvectors are called *eigenfaces*. It is assumed that the face class lies in the subspace $F = < w_i >_{i=1 \ldots M}$ spanned by the first M eigenvectors of a PCA computed on the training dataset. Traditionally, two measures are used for face detection:

- **Distance in the feature space (DIFS):** A Gaussian model is assumed for the face class in the subspace. The similarity measure between a candidate x and the face class is the Mahalanobis distance in the subspace (the distance between x and the sample mean $\bar{x}$).

$$DIFS(x) = \sum_{i=1}^{M} \frac{y_i^2}{\lambda_i} \quad (1)$$

where y_i is the projection of the mean normalized vector $x - \bar{x}$ on the ith-eigenvector and λ_i is the ith-eigenvalue.

- **Distance from the feature space (DFFS):** Another similarity measure between a candidate and the face class is the reconstruction error, the Euclidean distance between the candidate and its projection on the subspace.

$$DFFS(x) = \sum_{i=M+1}^{N} y_i^2 = \|x - \bar{x}\|^2 - \sum_{i=1}^{M} y_i^2 \quad (2)$$

DIFS defines, in the subspace F, concentric ellipses of points that are equidistant from the sample mean (in the sense of the Mahalanobis distance). Applying a threshold on this distance, an elliptical boundary of the face class is obtained. DFFS is a distance in the orthogonal subspace $F^{\perp}$. Note that DIFS and DFFS provide complementary information and may thus be combined [6].

3. SUPPORT VECTOR DATA DESCRIPTION

Support Vector Data Description was developed by Tax and Duin [10] to solve one-class classification problems.

Inspired by the Support Vector Machine learning theory, SVDD obtains a boundary around the target data set; this boundary is used to decide whether new objects are target objects or outliers.

Given a set of training target data $\{x_i\}, i = 1,...,N$, the simplest form of SVDD defines an hypersphere around the data. The sphere is characterized by a center a and a radius R. The goal is to minimize the volume of the sphere -minimize R^2- keeping all the training objects inside its boundary.

The structural error to minimize is:

$$F(R,a) = R^2 \quad (3)$$

with the constraints:

$$\|x_i - a\|^2 \leq R^2, \ \forall i \quad (4)$$

To allow outliers in the training set, the constraints are relaxed by introducing slack variables ξ_i; the minimization problem turns into:

$$F(R,a) = R^2 + C \sum_i \xi_i \quad (5)$$

with the constraints:

$$\|x_i - a\|^2 \leq R^2 + \xi_i, \quad \xi_i \geq 0 \ \forall i \quad (6)$$

The parameter C controls the tradeoff between errors and the volume of the description.

By introducing Lagrange multipliers α_i and γ_i, the following Lagrangian is obtained:

$$L(R,a,\alpha,\gamma,\xi) = R^2 + C \sum_i \xi_i - \sum_i \alpha_i (R^2 + \xi_i - (x_i.x_i - 2a.x_i + a.a)) - \sum_i \gamma_i \xi_i \quad (7)$$

L has to be minimized with respect to R, a and ξ_i and maximized with respect to α_i and γ_i. Solving the partial derivatives of L the following constraints are found:

$$\sum_i \alpha_i = 1 \quad (8)$$

$$a = \sum_i \alpha_i x_i \quad (9)$$

$$0 \leq \alpha_i \leq C \quad (10)$$

Replacing equations (8-10) into equation (7), the Lagrangian is:

$$L = \sum_i \alpha_i (x_i.x_i) - \sum_{i,j} \alpha_i \alpha_j (x_i.x_j) \quad (11)$$

The maximization of (11) gives a set $\{\alpha_i\}$. Objects x_i with $\alpha_i > 0$ are called the *support vectors* (SV) of the description. Support vectors lie on the boundary (if $0 < \alpha_i < C$) or outside the boundary (if $\alpha_i = C$) of the sphere that contains the data. Equation (9) shows that the center of the sphere is a linear combination of the support vectors.

A new object z is accepted as a target object if it is inside the description. Hence, the following condition has to be verified:

$$\|z - a\|^2 = (z.z) - 2 \sum_i \alpha_i (z.x_i) + \sum_{i,j} \alpha_i \alpha_j (x_i.x_j) \leq R^2 \quad (12)$$

This formulation of SVDD can be extended to obtain a more flexible description. Data is mapped nonlinearly into a higher dimensional space where a hyperspherical description can be found. The mapping is performed implicitly, replacing the inner products in (11) by a kernel function:

$$K(x_i,x_j) = (\Phi(x_i).\Phi(x_j)) \quad (13)$$

Several kernel functions have been proposed in [10]. In our experiments we use a Gaussian kernel of the form:

$$K(x_i,x_j) = \exp\left(\frac{-\|x_i - x_j\|^2}{s^2}\right) \quad (14)$$

This kernel is independent of the position of the data with respect to the origin since only uses distances between objects. The parameter s is a width parameter that controls how tight the description is around the data.

Using this kernel, a new object z is accepted if:

$$\sum_i \alpha_i \exp\left(\frac{-\|z - x_i\|^2}{s^2}\right) \geq \frac{1}{2}(-R^2 + B) \quad (15)$$

where B depends only on the support vectors.

This function is a threshold on a weighted sum of Gaussians, so the boundary description is strongly influenced by the width parameter s. For small values of s all the objects tend to be support vectors. The description approximates a Parzen density estimation with a small width parameter. For large values of s, the SVDD description approximates the original spherical solution. For intermediate values of s, a weighted Parzen density is obtained.

Figure 1 shows the descriptions obtained for different values of s, using as feature space the first two PCA projections of a set of face images (see Section 4).

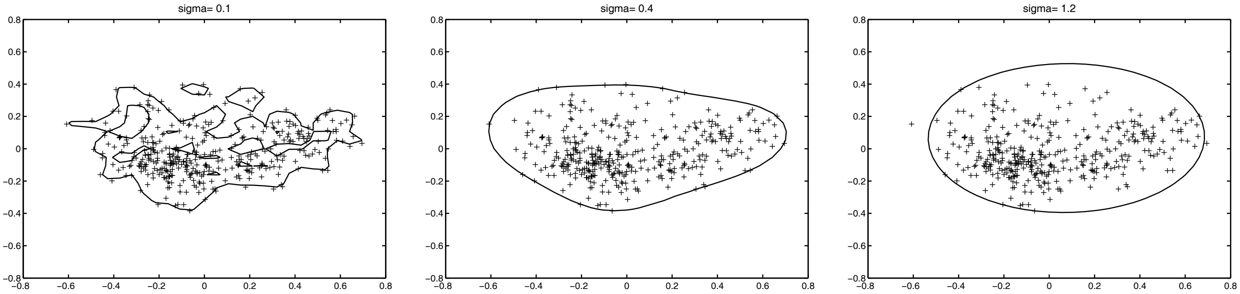

Figure 1: Scatterplots and data descriptions for the first two PCA projections of a training set of face images. Gaussian kernels with different widths (s = 0.1, 0.4, 1.2) are used. The solid lines show the description boundaries.

By increasing s, the number of support vectors decreases. When the training set represents the true target distribution, the fraction of support vectors gives an estimate of the error on the target set. Then, as s increases, the number of support vectors and hence the error on the target set decreases. However, when the description boundary is larger, more outliers become inside the description, so the number of false negatives also increases.

4. EXPERIMENTS

The use of eigenfaces to describe facial texture is appealing because it reduces the dimensionality of the input feature space and thus less samples are needed to train the classifier. Another reason is that eigenfaces proved to be robust features in real face applications.

We propose to use SVDDs trained on PCA features to model the face class boundary. We want to compare the performance of SVDDs with the elliptical boundaries obtained with the DIFS classifier and analyze the number of features that are needed to build the face class boundary and the selection of parameters for the SVDDs.

All the experiments with SVDDs were performed with Matlab PRTools4 [2] and dd_tools1.11 [9].

4.1 Training and test sets

The first set of experiments uses for training a subset of 800 images from the XM2VTS [5] database (200 different individuals, 4 samples per individual). Faces were manually cropped and rescaled to 40x60 pixels. This set is used to train the PCA and the SVDDs.

In the second set of experiments, the training set consists of 1040 face images from the Spanish part of Banca database (g1 sets from all the controlled sessions).

The test set contains faces from the following databases: BioID [3] (400 images), Banca [1] (1040 images in g2 sets from all controlled sessions in the Spanish part) and XM2VTS (400 images not included in the training set).

The negative examples for all the experiments were collected from various images not containing faces and from the background of images that contained a face. We use a total of 12000 no-face images. These samples are only used to test the classifiers.

4.2 Results

The SVDD formulation presented in Section 3 corresponds to a hard-decision classifier; the output is 1 or 0. To compare the performances of SVDDs and DIFS and to be able to combine the SVDD with other classifiers we transform it into a soft-decision classifier.

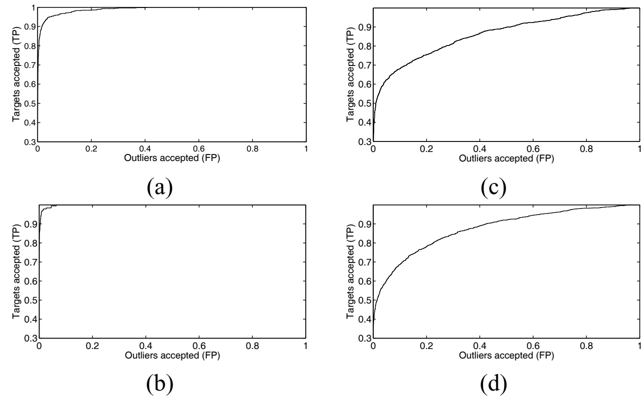

Figure 2: ROC curves: (a) DIFS with M=10 and XM2VTS test set, (b) SVDD with M=10, s=0.2, and XM2VTS test set, (c) DIFS with M=10 and complete test set, (d) SVDD with M=10 and complete test set.

For an input pattern z, the output of the SVDD is now:

$$d(z) = \underset{i}{} {}_i \exp\left(\frac{-\|z-x_i\|^2}{s^2}\right) \tag{16}$$

Varying a threshold on this distance, we may compute the ROC curve (targets accepted vs. outliers accepted) for the classifier.

First, we use the same database XM2VTS for training and for test. In this case, the training set is a representative sample from the face class distribution; both classifiers perform well and SVDD outperforms DIFS. Figure 2-a shows the best ROC obtained for DIFS, which corresponds to a feature space of dimension 10. For the same dimensionality, the performance of the SVDD (for s=0.2) is clearly better (Figure 2-b).

In the second case, XM2VTS is used for training but the test set contains images from other databases (BIO, Banca). XM2VTS images were acquired under quite controlled conditions. Faces are frontal and have a neutral expression. Banca and BIO faces, on the contrary, present a large variability in illumination and facial expressions. Now the training data distribution is very different from the test set dis-

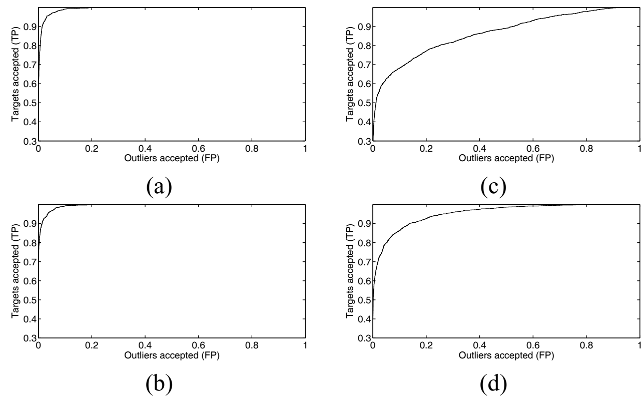

Figure 3: ROC curves: (a) DIFS with M=100 and Banca test set, (b) SVDD with M=50, s=0.2, and Banca test set, (c) DIFS with M=100 and complete test set, (d) SVDD with M=50, s=0.2 and complete test set.

tribution, so the SVDD overtrains on the training data and its performance degrades. DIFS, which finds an elliptical boundary around the data, performs slightly better than SVDD. Figures 2-c and 2-d show the ROC curves for DIFS and DFFS using the first 10 features, respectively.

Next, we train the classifiers with the Banca training set. As expected, when we use the same database for testing both classifiers perform well. However, SVDD finds good descriptions using less features than DIFS. Figures 3-a and 3-b show the ROC curve for DIFS using 100 features (the best result obtained for DIFS), and a very similar ROC for SVDDs using only 50 features. In this case the target class is more complex and we need more features than in the previous experiment to reach the same performance.

Finally, we test the classifiers with images from different databases. The ROC curves for DIFS trained with 100 features, and for SVDD trained with 50 features are presented in Figures 3-c and 3-d. SVDD performs better, using only half the features.

We have also studied the influence of the feature space dimensionality in the performance of the SVDD. We have trained several SVDDs keeping s constant and varying the number of eigenfaces, and we have found that at a certain point, increasing the number of eigenvectors does not improve much the results ($M \simeq 30$ and $M \simeq 70$ for XM2VTS and Banca training sets, respectively). In high dimensions, most of the points become support vectors (for any value of s), and this suggests that more data is required. In those cases we need more training points to find a reliable description of the boundary.

5. CONCLUSIONS

In this paper, we have focused on the use of the Support Vector Data Description classifiers based on PCA features for face detection and have compared SVDDs with classifiers based on DIFS.

SVDD finds a more flexible boundary description than DIFS. Our experiments have shown that when the training data is a representative sample of the target distribution or when the training data does not model its distribution but covers the target class area, SVDD outperforms DIFS.

We have obtained good results for relatively low dimensions of the feature space. In high dimensions, more training data is required to find a reliable boundary estimation.

Based on these results we are replacing the DIFS classifier by a SVDD classifier as the texture-based classifier included in our face detector.

6. ACKNOWLEDGMENTS

This work has been partially supported by the IST programme of the EU in the project IST-2000-32795 SCHEMA and the project ACERCA TEC2004-0194 of the Spanish Governement.

REFERENCES

[1] E. Bailly-Bailliere, S. Bengio, F. Bimbot, M. Hamouz, J.V. Kittler, J. Mariethoz, J. Matas, J. and K. Messer, V. Popovici, F. Poree, B. Ruiz and J.P. Thiran, The BANCA Database and Evaluation Protocol, *AVBPA03*, pp. 625–638, 2003.

[2] R.P.W. Duin, P. Juszczak, P. Paclik, E. Pekalska, D. de Ridder and D.M.J. Tax, PRTools4, A Matlab Toolbox for Pattern Recognition, Delft University of Technology, 2004.

[3] O. Jesorsky, K. Kirchberg and R. Frischholz, Robust Face Detection Using the Hausdorff Distance, *AVBPA01*, pp 90-95, 2001.

[4] F. Marqués and V. Vilaplana, Face Segmentation and Tracking Based on Connected Operators and Partition Projection, *Pattern Recognition*,vol. 35, no. 3, pp. 601–614, 2002.

[5] K. Messer, J. Matas, J.V. Kittler, J. Luettin and G. Maitre, XM2VTSDB: The Extended M2VTS Database, *AVBPA99*, 1999.

[6] B. Moghaddam and A. Pentland, Probabilistic Visual Learning for Object Representation, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no 7, pp. 696–710, 1997.

[7] E. Osuna, R. Freund and F. Girosi, Training Support Vector Machines: An Application to Face Detection, *Proc. Computer Vision and Pattern Recognition*, 1997.

[8] K. K. Sung and T. Poggio, Example-Based Learning for View-Based Human Face Detection, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 1, pp. 39–51, 1998.

[9] D. Tax, dd_tools 1.11, A Matlab toolbox for data description, outlier and novelty detection, Delft University of Technology, 2003.

[10] D. Tax and R. Duin, Support Vector Data Description, *Machine Learning*, 54, pp.45–66, 2004.

[11] P. Viola and M. Jones, Rapid Object Detection Using a Boosted Cascade of Simple Features, *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, 2001.

[12] M. Yang and D. Kriegman and N. Ahuja, Detecting Faces in Images: A Survey, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 1, pp. 34–58, 2002.