Chapter 4

Contrast-Invariant Appearance-Based Detection

4.1 Introduction

In this chapter we propose two different models for contrast-invariant appearance detection. The contrast signal, which is the intensity variation around a mean brightness value, can be measured in two ways, either by an object-specific, global model or a generic, local method. Once the dataset is adjusted for contrast, a representation is built using the principal subspace (*within-space*) and its orthogonal complement (*out-of-subspace*). We experiment with principal subspaces obtained by PCA and S-PCA basis vectors. An important feature of our system is a *perceptual distortion measure* that we designed to compare the appearance of an object to its reconstruction from the principal subspace. By distortion we mean the reconstruction error caused by not including the out-of-subspace signal. We show how to use S-PCA basis trained on generic background patches to decide if two images are perceptually similar.

Our work builds upon four key observations on subspace-based representations. First, subspace methods involve least squares approximations which are notoriously sensitive to large outliers. Hence, it is important to account for outliers in training, and testing, subspace methods [17, 30]. A related issue is one of variation in the image contrast. In general, images with higher contrasts have larger variance. If some images have variance much larger than others then these images can potentially skew the estimation of the object-specific subspace. Often a contrast normalization step is performed, either by histogram equalization or by normalizing the variance (or the power) of an image so that it is of unit length. We offer two alternative normalization strategies which we feel are more appropriate for object-specific ensembles.

Second, S-PCA amounts to a rotation of input coordinate axes and, as such, it does not define a probability density model for the input data. However, subspaces can also be derived from the perspective of density estimation [75, 92, 109]. The advantage in estimating the input density is that it allows for the design of probabilistic methods to detect, recognize and/or classify test images as appearances of known objects. In particular, one strategy for object representation is to divide the signal space into a principal subspace and its orthogonal complement and then build probability models separately for the two subspaces. Then the detection strategy is to apply a threshold on the likelihood assigned by the combined density model to a test image [75].

The third key observation, as noted in [25, 74], is that the variance estimate *per–pixel* given by the subspace density models for the residual signal in the out–of–subspace is overly conservative [75, 109]. We defer the actual details of variance estimation used in our appearance-based detector to a later section.

Finally, the detection problem can be posed as one of assessing image similarity, where the issue is to develop a measure that captures the notion that two images are *perceptually* similar. It is well known that standard error norms, such as the mean-squared-error (MSE), are unsuitable for measuring perceptual distortion. Recent successes in formulating perceptual distortion norms (e.g., [105]) have come from analyzing the psychophysics of spatial pattern detection, particularly contrast and orientation masking, and understanding the functional properties of neurons in the primary visual cortex. A typical perceptual distortion model consists of a linear transformation of images by a "hand-crafted" wavelet representation that is tuned to different spatial orientations and scales, followed by a divisive normalization mechanism. The normalization scheme involves pooling information in adjacent wavelet channels (that is in neighbouring spatial locations, orientations and scales, eg. [18]). Normalization provides a context for local significance, in that a high sensor channel (wavelet) response is down-weighted if the adjacent channels are equally active but upgraded otherwise. The full set of normalized sensors tuned for different spatial positions, spatial frequencies, orientations and contrast discrimination bands provide a basis for assessing the perceptual similarity between two images. Our work generalizes this normalization scheme to object-specific multi-scale representations derived from S-PCA.

While we concentrate on sub-space methods, much work has also been done in building feature-based object detectors [98, 104], in particular systems where the features are simple to compute and hence the objects are fast to detect [3, 39, 82, 117].

4.2 Datasets

We investigate two different image datasets: eyes/non-eyes [25] and faces/non-faces [6]. The eye images are regions cropped from the FERET face database [87]. The face images were first scaled and rotated such that, in the warped image, the centers of left and right eyes have a horizontal separation of 40 pixels. From these warped images, we crop image regions around the eyes, each of size 20 × 25 pixels, to generate a database of 2392 eye patches (Fig. 4.1).

For non-eyes, we construct a generic background patch ensemble by running an interest point detector [97] on several different natural images and collecting image patches with detector responses above a certain threshold (Fig. 4.2). The interest point detector

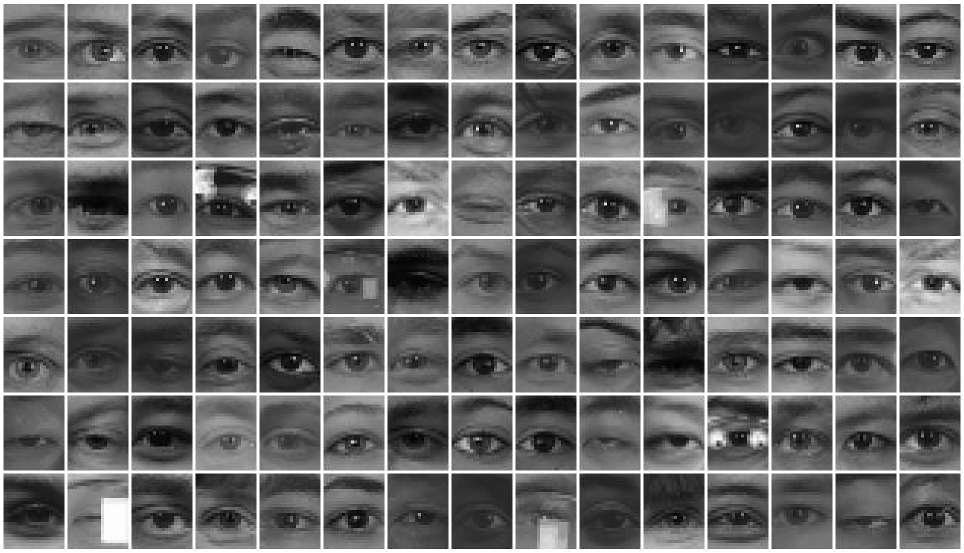


Figure 4.1: Eye images cropped from the FERET database [87]. Each image is of size 20 × 25 pixels.

can be seen as a first step in the detection hierarchy in that it eliminates blank, textureless regions from further consideration. To populate the 500 dimensional input space with a sufficiently large number of positive and negative examples, we symmetrize the ensemble. In particular, the eye images were flipped to generate mirror-symmetric pairs for a total of (2 × 2392) images. We take more liberties with the generic background patches, reflecting them about the x-/y-axis and around the origin, to make the original database of 3839 images four times as large. The datasets were randomly split in half to train and test the detection algorithm proposed here. We defer the details of the MIT face database [6] to the results section of this chapter.

4.3 Detection Model – I

The approach we take here is to use a global, object-specific model to normalize the ensemble for contrast variation and then build a low-dimensional description for the data. We then measure two statistics that we show are relevant for appearance detection.

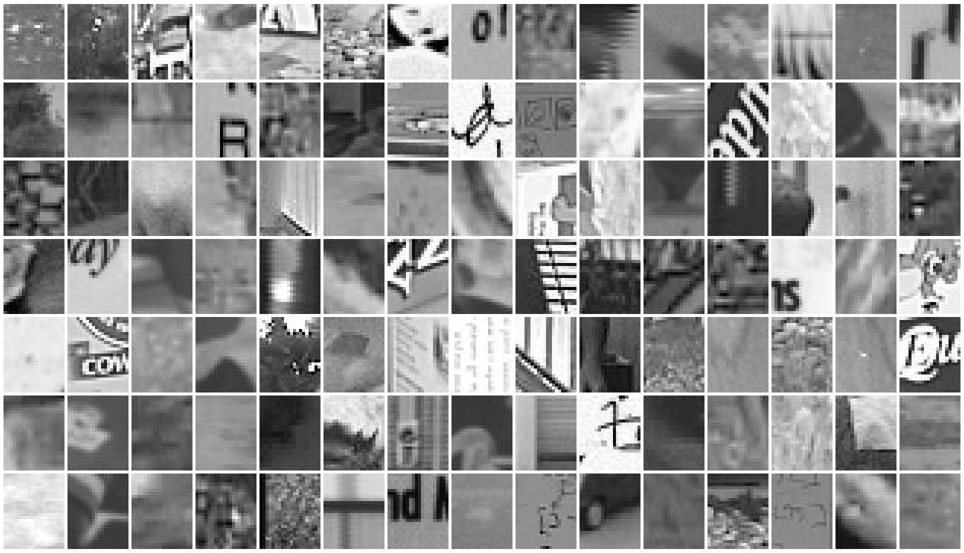


Figure 4.2: Generic background patches detected by an interest point detector [97]. The image size is 20 × 25.

4.3.1 Contrast Model for an Object-Specific Ensemble

Contrast is defined as the variation in image intensities around a mean brightness value. We can extend this contrast model to be global and explicit for an object-specific ensemble by using two basis vectors, namely a constant "DC" basis vector and a vector in the direction of the mean of the training set after removing the contribution from the DC basis. We define the training set as $\{\vec{t}_k\}_{k=1}^{K}$ where K is the total number of training images and $\vec{t}_k$ is the k^{th} training image stored in a column format. Each element of the DC vector $\vec{\xi}$ is given by

$$\xi_i = 1/\sqrt{N}, \tag{4.1}$$

where N is the total number of pixels in the image, and $\vec{\xi}$ is of unit length. The mean image $\vec{\mu}$ is derived from the training set after removing the DC contribution, that is

$$\vec{\mu} \propto \frac{1}{K} \sum_{k=1}^{K} \left[\vec{t}_k - \vec{\xi} \left(\vec{\xi}^T \vec{t}_k \right) \right], \tag{4.2}$$

and $\vec{\mu}$ is normalized to be of unit length. We define the components of $\vec{t_k}$ in the directions of the DC and the mean vector as

$$d_k = \vec{\xi}^T \vec{t_k} \tag{4.3}$$

and

$$m_k = \vec{\mu}^T \vec{t_k} \tag{4.4}$$

respectively.

The goal here is to generate a subspace representation for the spatial structure of the eye images that is insensitive to contrast changes. We wish to capture the spatial structure by a basis matrix B given by

$$B = \begin{bmatrix} \vec{b}_1 & \vec{b}_2 & \cdots & \vec{b}_N \end{bmatrix}. \tag{4.5}$$

It is possible that the spatial structure of eye images, and thus the basis matrix B, changes as the mean component m_k increases. Alternatively, these basis images may stay roughly the same and only the variation of their coefficients increases with the mean component m_k. The coefficient $c_{k,j}$ in the direction of j^{th} basis vector for the k^{th} image is given by,

$$c_{k,j} = \vec{b}_j^T \vec{t}_k. \tag{4.6}$$

Indeed, if the variation of the mean component is primarily due to lighting and imaging effects, then we might assume that the underlying signal, i.e. the spatial structure of eye images, is invariant to contrast. In this case we would expect the basis images $\vec{b}_j$ to be independent of m_k and only the variance of coefficients $c_{k,j}$ to scale as a function of m_k.

We test this hypothesis by analyzing the relationship between the mean component m_k and the contrast variance l_k, given by

$$l_k = \frac{1}{N} \left\| \vec{t_k} - d_k \vec{\xi} - m_k \vec{\mu} \right\|_2^2, \tag{4.7}$$

where $\|\cdot\|_2^2$ is the expression for the square of the two-norm and l_k determines the variance-per-pixel that remains unaccounted for, after using the DC and the mean vectors to describe the ensemble. In Fig. 4.3(Left) we plot the contrast variation $\sqrt{l_k}$ as a function of the mean coefficient m_k for each eye image. Notice that there are a large range of variances in the training images, with those images with larger values of the mean coefficient having significantly larger variances. As a rough approximation this relationship can be modeled by a straight line fit to the data, shown by the black line going through the origin in Fig. 4.3(Left). Note, the straight line fit is biased by the constraint that it passes through the origin (denoting perfect contrast invariance). From Fig. 4.3(Left) there appears to be a trend away from the line for smaller values of the mean coefficient. We suspect this is due to the use of a single mean image, which may not model the lower contrast images particularly well.

To balance the variances across different image contrasts, we rescale the training images to get

$$\vec{q}_k = \frac{\left[\vec{t}_k - d_k\vec{\xi} - m_k\vec{\mu}\right]}{s(m_k)}, \tag{4.8}$$

where $s(m_k)$ is a scaling factor for training image $\vec{t}_k$ with mean coefficient m_k. It is convenient to assume a minimum value for the left-over variance, say due to independent pixel noise, and use this in the estimate for scaling:

$$s(m_k) = \sqrt{\sigma_{\min}^2 + f(m_k)}. \tag{4.9}$$

The red curve in Fig. 4.3(Left) is a plot of $s(m_k)$ with $f(m_k) = p \times m_k^2$, where p is the slope of the straight line fit (the black line in Fig. 4.3(Left)), and $\sigma_{\min} = 8$. After normalization, all the scaled images falling on the red curve shown in Fig. 4.3(Right) will have a left-over-variance value of one. We discuss next how to use the rescaled images for training a principal subspace.

4.3.2 Principal Subspace and its Complement

Once the images are rescaled into $\vec{q}_k$ for $k = 1, \cdots, K$, we first trim the database by removing extreme images that either have a low or negative mean image coefficient (≤ 20) and/or very large left-over variances per pixel (> 25). Most of these extreme images have

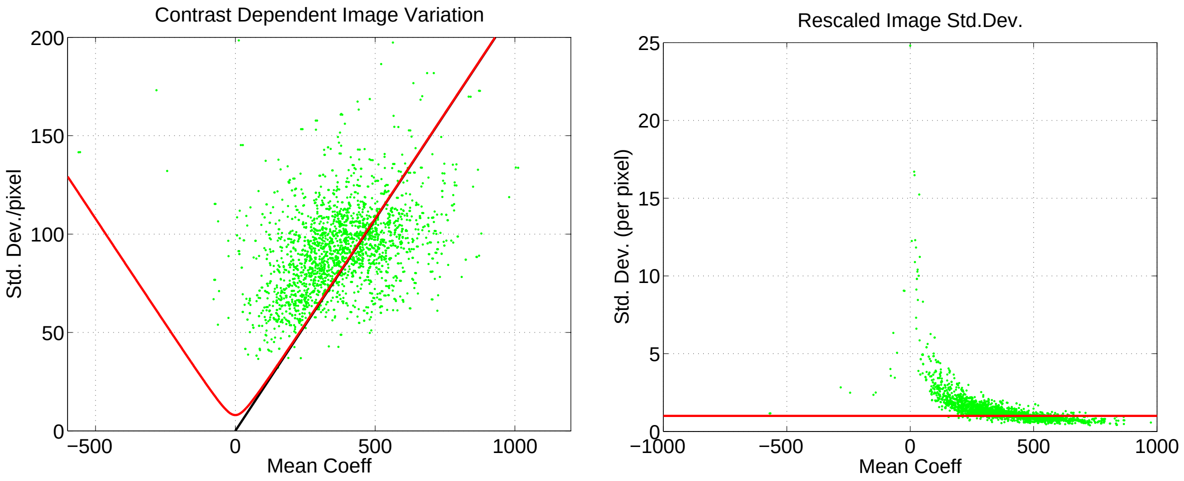


Figure 4.3: Contrast dependency of the Eye Dataset.

identifiable problems, such as the eye being closed, not centered, having a different scale, or having reflections from glasses. For the database we consider here they constitute only a small fraction of the input data: 46 out of 2392 eye images in the training set. We then use singular value decomposition (SVD) to perform principal component analysis. The PCA bases are shown in Fig. 4.4. The PCA basis in turn are used as a starting point for training the S-PCA representation. The S-PCA basis is trained for a complete representation of the input space, that is, the basis matrix is square. The resulting S-PCA basis are shown in Fig. 4.5.

Let $\vec{b}_j$ denote the basis images obtained either by PCA or S-PCA and σ_j^2 the variance obtained by projecting the data onto the basis direction $\vec{b}_j$. Suppose we approximate the normalized images $\vec{q}_k$ with just the M leading basis images. We can compute the residual signal $\vec{e}_k$ in the complement space as

$$\vec{e}_k = \vec{q}_k - \sum_{j=1}^{M} c_{k,j} \vec{b}_j, \tag{4.10}$$

where the expansion coefficient is given by

$$c_{k,j} = \vec{b}_j^T \vec{q}_k, \tag{4.11}$$

and M is taken to be much less than the total number of training images K and the total

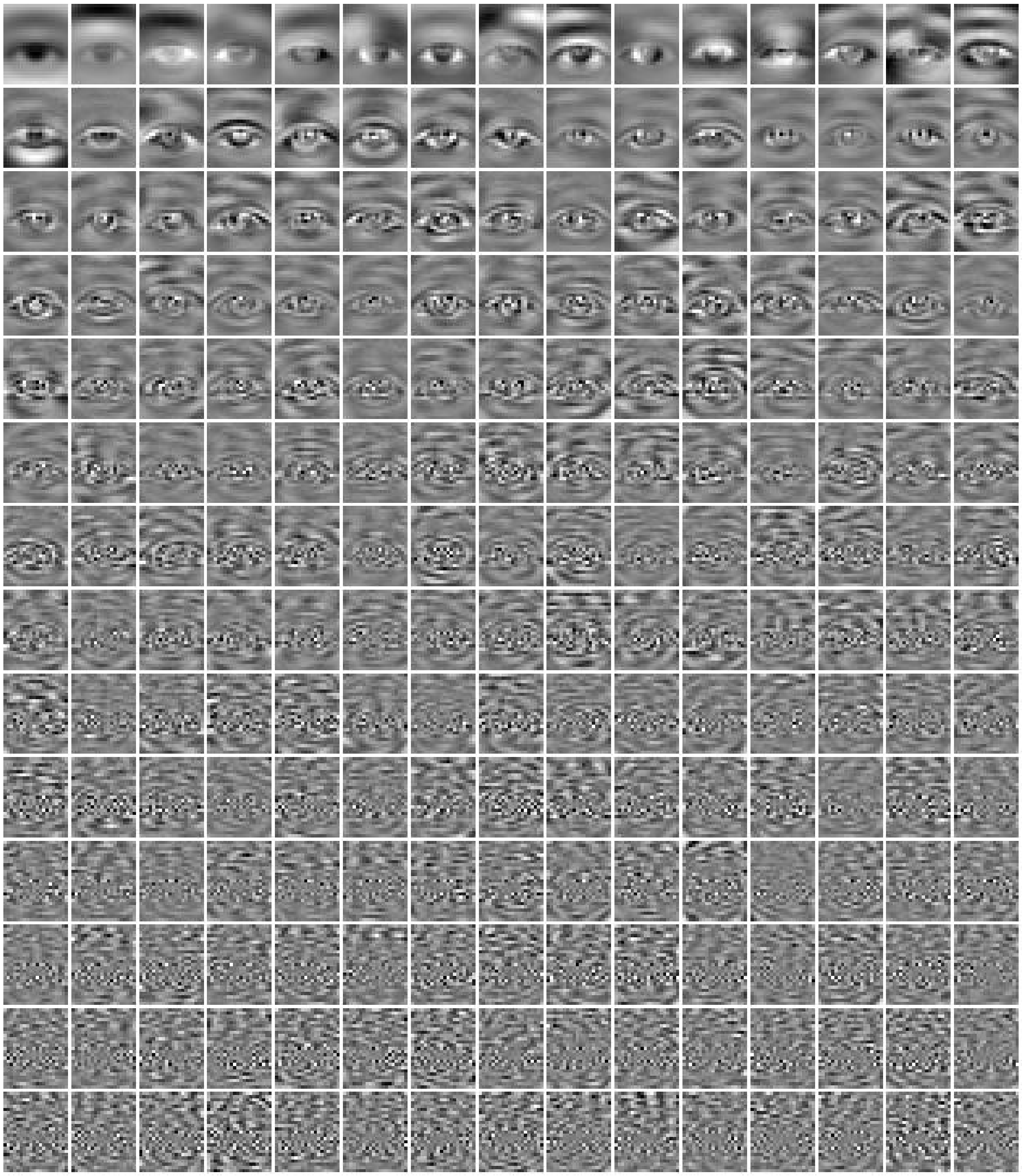


Figure 4.4: PCA basis for the eye space, arranged from left to right in decreasing order of the input variance captured. The first sub-image on the left is the mean image.

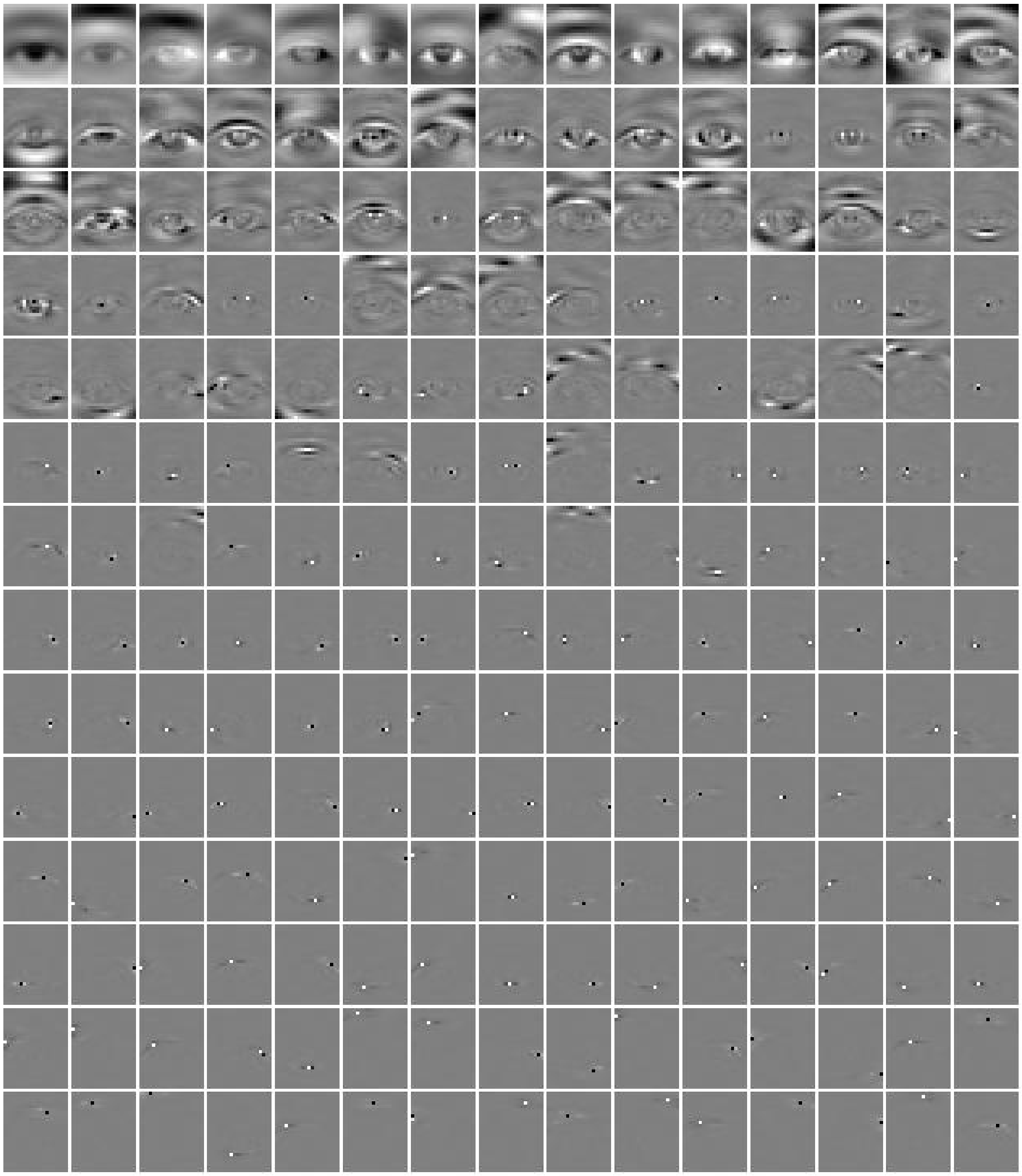


Figure 4.5: S-PCA basis for the eye space, arranged from left to right in decreasing order of the input variance captured. The first sub-image on the left is the mean image.

number of pixels N. We can now define the residual variance to be

$$v_M(\vec{x}) = \frac{1}{K}\sum_{k=1}^{K} e_k^2(\vec{x}), \tag{4.12}$$

where $v_M(\vec{x})$ denotes the variance at pixel $\vec{x}$ resulting from approximating the input with the leading M-dimensional subspace. This expression is exact for PCA on the training set, that is the residual variance at pixel $\vec{x}$ after projecting out the first M basis directions (along with the DC and the mean images) in the training set is just $v_M(\vec{x})$. The residual variances at nearby pixels are likely to be correlated. We ignore this correlation issue for now, but this is an important issue and we will return to it in a future section.

4.3.3 Detection Strategy

The detection strategy is to expand test images in terms of the DC and mean images, along with the first M leading basis images and measure two statistics namely, m_k and S_k^{oos}. Here m_k is the coefficient of the mean image, and

$$S_k^{\text{oos}} = \frac{1}{N}\sum_{\vec{x}} \frac{\left[\vec{t}_k(\vec{x}) - d_k\vec{\xi}(\vec{x}) - m_k\vec{\mu}(\vec{x}) - \sum_{j=1}^{M} c_{k,j}\vec{b}_j(\vec{x})\right]^2}{\vec{v}_M(\vec{x})}, \tag{4.13}$$

measures the variance of the residual error in expanding the test image $\vec{t}_k$ by the leading M-dimensional basis images (along with $\vec{\xi}$ and $\vec{\mu}$), as compared to the residual variance $\vec{v}_M$ obtained by expansions of the same form over the training set.

The variance plot drawn in Fig. 4.6(LEFT) shows $\sqrt{S_k^{\text{oos}}}$ computed with just the DC and the mean images (i.e. $M = 0$) and it is clear from the figure that the feature spaces of S_k^{oos} and m_k are amenable to a simple classification strategy. In particular, we use the constraint that

$$m_k > m_{\min} \tag{4.14}$$

where $m_{\min}$ is a small positive value ($= 20$). Negative values of m_k correspond to contrast reversals and small positive values for m_k generate a mean image component which varies only by a few gray levels. Additionally, we apply a simple contrast invariant threshold of the form

$$\arctan(m_k, \sqrt{S_k^{\text{oos}}}) \leq \tau_{\text{oos}}, \tag{4.15}$$

which requires that the distribution of the eye dataset be underneath a line drawn through the origin at an angle τ_{oos} for successful detection. A particular detector is then defined by the number of basis vectors used for the principal subspace and the choice of the parameter τ_{oos}.

4.3.4 Results

We first discuss results from using the PCA basis and then show the improvement in performance obtained by the S-PCA basis. While the eye data is split into training and testing sets, the generic background patch ensemble is considered as one large test dataset.

In Figs. 4.6 and 4.7 we show the separation of the test datasets: eyes (green) and non-eyes (red), in the feature space of $\sqrt{S_k^{\text{oos}}}$ vs m_k, as a function of the increasing subspace dimensionality ($M = 0, 20, 50,$ and 100). The detection threshold lines drawn as black lines in each one of the plots generate roughly 5% false positives. The true detection/rejection rates and the corresponding ROC curve obtained by changing the value of the τ_{oos} parameter can be seen in more detail in Fig. 4.8. The simplest strategy of using just the DC and the mean images (i.e. $M = 0$) results in a detection rate of 93% and a false target rate of 7%. With increasing subspace dimensionality, the datasets appear to be more separable visually, however the improvement in the ROC curves is marginal. In particular, a 50-dimensional PCA subspace is suitable for describing the eye manifold, as the true detection rate grows to 95% and the false positive rate reduces to $\approx 5.5\%$ and increasing M further causes over-fitting on the training set, as seen from the ROC plots in Fig. 4.9.

The discriminative ability of the PCA representation is not high. Is there any im-

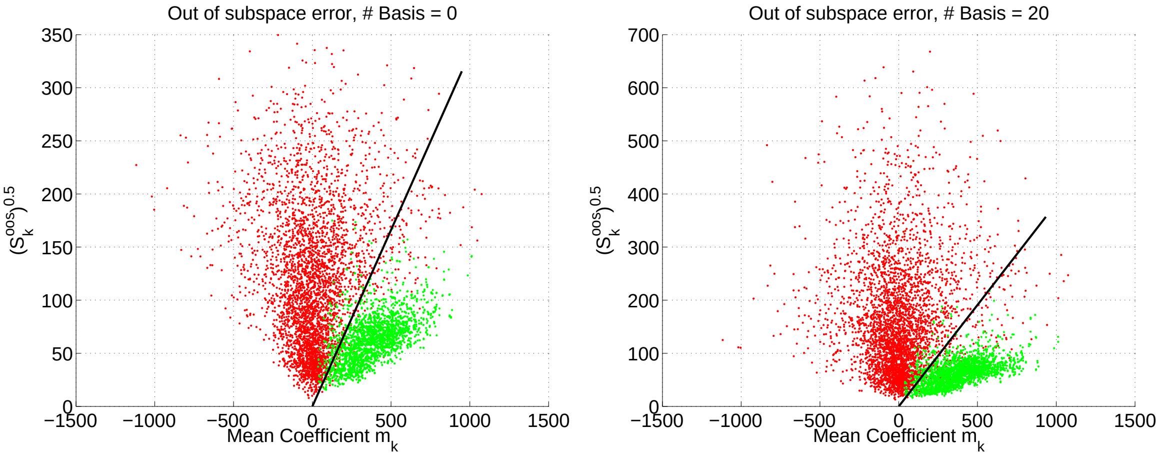


Figure 4.6: Separation of the eye and the non-eye clouds in the feature space of $\sqrt{S_k^{\text{oos}}}$ vs m_k with the addition of 0 (LEFT) and 20 (RIGHT) PCA basis.

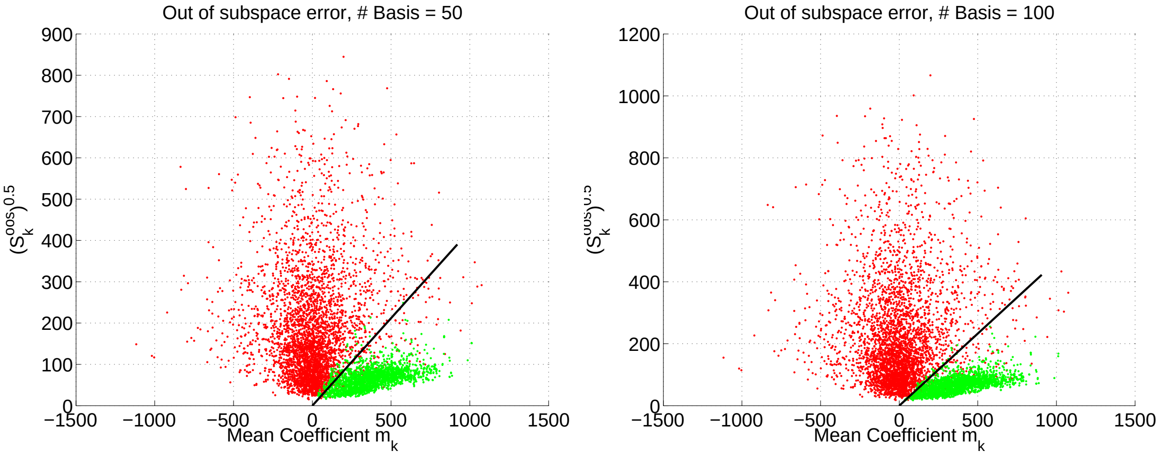


Figure 4.7: Separation of the eye and the non-eye clouds in the feature space of $\sqrt{S_k^{\text{oos}}}$ *vs* m_k with the addition of (LEFT) 50 and (RIGHT) 100 PCA basis.

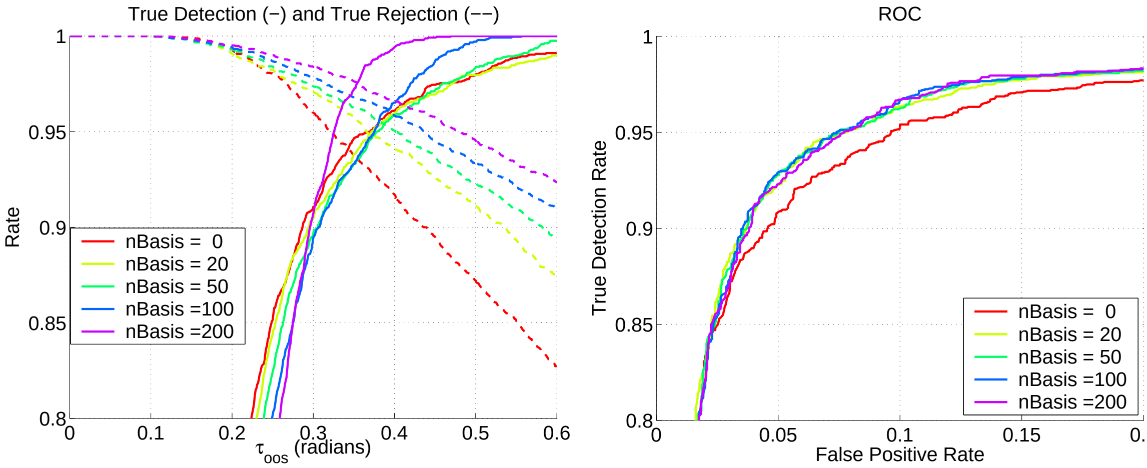


Figure 4.8: PCA basis applied to the test dataset. (Left) True detection and rejection rates. (Right) ROC curve.

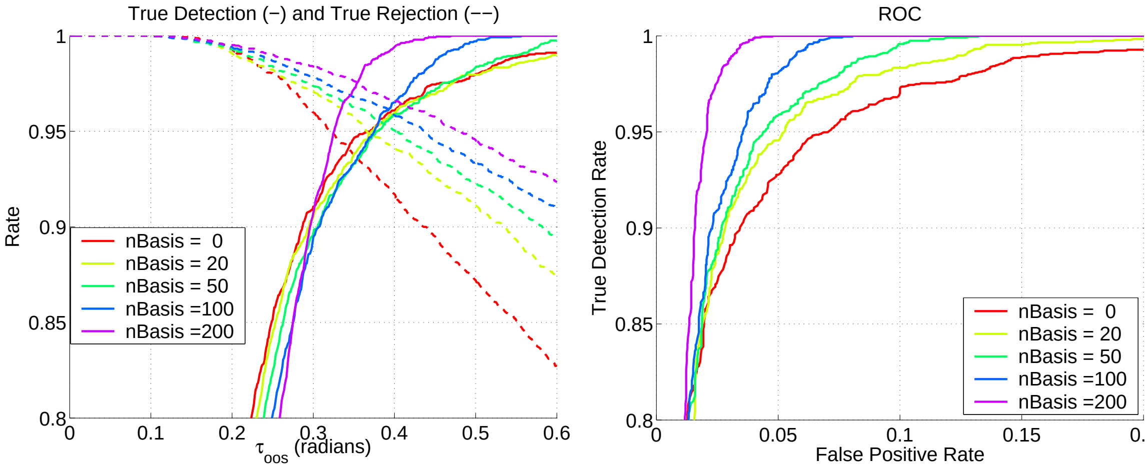


Figure 4.9: PCA basis applied to the training dataset. (Left) True detection and rejection rates. (Right) ROC curve. Note the difference from Fig. 4.8 in the curves for nBasis = 100 and 200, indicating overfitting.

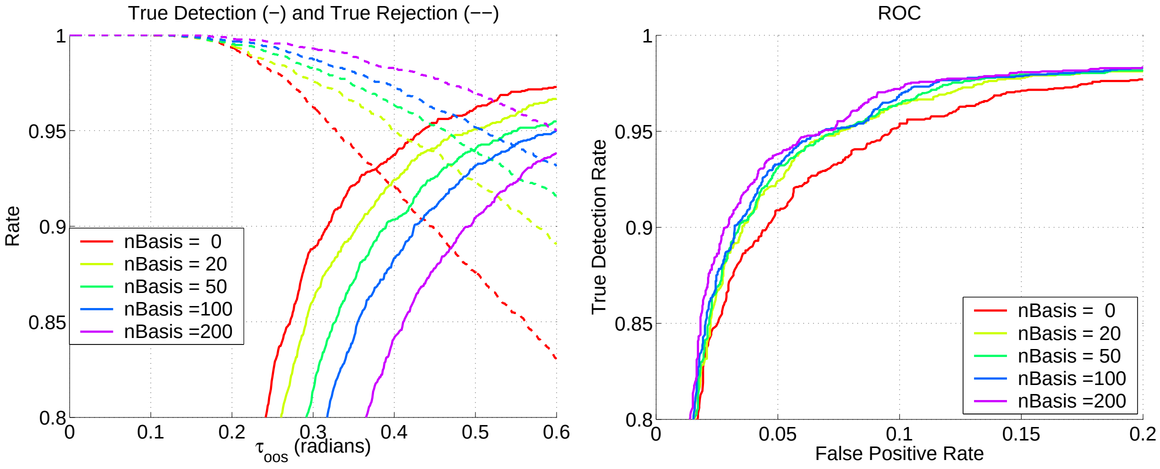


Figure 4.10: S-PCA basis applied to the test dataset. (Left) True detection and rejection rates. (Right) ROC curve.

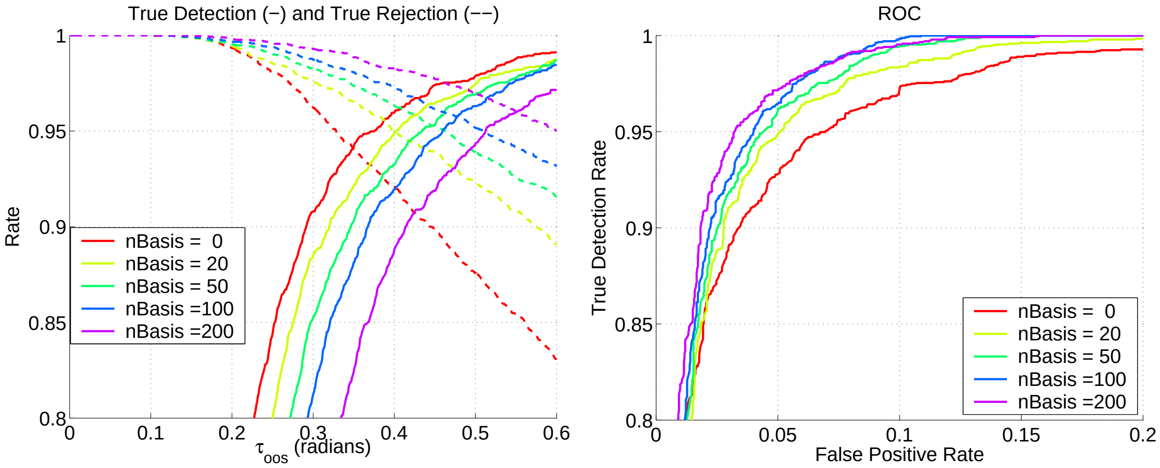


Figure 4.11: S-PCA basis applied to the training dataset. (Left) True detection and rejection rates. (Right) ROC curve.

provement in using the S-PCA basis? The results from applying the S-PCA basis on the test datasets is shown in Fig. 4.10. Compared with the results obtained from using the PCA basis, as shown in Fig. 4.8, the improvement with S-PCA basis is at most a few percentage points. It is not surprising that PCA and S-PCA do about the same because the reconstruction errors are roughly the same. While there is an advantage in using S-PCA basis, in that they facilitate fast detection (see § 3.6), the recognition rates for both PCA and S-PCA are less than satisfactory.

A closer inspection of the eyes rejected as false negatives shows that these are images that look extreme, in that many have highlights caused by the specularities from the eye glasses or pixel outliers such as the hair falling over the eye brows etc. It is possible to improve on the false negative rates by taking into account only relevant portions of an image in detecting eyes, as we have done in [25]. However, the high false positives rates are a major cause for concern and we present a vastly improved detection model next.

4.4 Detection Model – II

The problem of detecting known appearances is analogous to the problem of measuring image similarity, in that we need a perceptual error norm for meaningful results. In § 4.3 we used a mean-squared-error (MSE) norm for measuring image distortions and we ignored the fact that the residual errors in the out-of-subspace signal may be correlated spatially. Thus, the detector model we proposed earlier is unlikely to capture perceptual distances between image pairs.

Motivated by the work in [105], we propose a new detector model outlined in Fig. 4.12. For this new detector model, we find it convenient to employ a generic, local contrast-normalization scheme, so that there is no bias introduced by the object-specific mean image into the dataset (Eq. 4.2). There are five steps involved: (1) contrast-normalize ($\mathcal{WCN}$) the test image $\vec{y}$ to obtain $\vec{t}$; (2) project $\vec{t}$ onto the wavelet-like space W derived

$$\begin{tikzcd}
\vec{t} \arrow[r, leftarrow, "\mathcal{WCN}"] \arrow[d, "W^T"] & \vec{x} & \\
\vec{d} \arrow[r, "B^T"] \arrow[d, "\mathcal{PDN}"] & \vec{b} \arrow[r, "B"] & \hat{\vec{d}} \arrow[d, "\mathcal{PDN}"'] \\
\vec{z} & & \hat{\vec{z}}
\end{tikzcd}$$

Figure 4.12: Detector Model – II with perceptual distance normalization ($\mathcal{PDN}$). See § 4.4 for more details.

from training S-PCA on generic background patches and obtain $\vec{d}$ as the coefficient vector; (3) build a low-dimensional approximation $\hat{\vec{d}}$ to the coefficient vector $\vec{d}$ using S-PCA basis B constructed for the object-specific ensemble in the "wavelet" space; (4) apply perceptual distance normalization $\mathcal{PDN}$ on the coefficient vector $\vec{d}$ and its reconstruction $\hat{\vec{d}}$ to obtain normalized vectors $\vec{z}$ and $\hat{\vec{z}}$; and finally (5) apply a simple detection strategy to $\vec{z}$ and $\hat{\vec{z}}$. We explain these details next.

4.4.1 Weber-Contrast Normalization

Weber-contrast is a measure of the relationship between the response of a pixel and that of its neighborhood. In particular, if x_i is the response of a pixel at location i and μ_i is an estimate of the mean response value in its neighborhood, then the Weber contrast signal c_i is defined as:

$$c_i = \frac{x_i - \mu_i}{\mu_i}. \tag{4.17}$$

The mean signal value μ_i can be obtained by convolving the image with a two-dimensional radially-symmetric Gaussian filter $G(i\,;\,\sigma)$. The neighborhood size is determined by the standard deviation σ of the Gaussian function. While this contrast computation removes shading variations, there are pixel outliers, such as the specularities from the eye glasses or the hair falling over the eye brows, that can bias the computation (Fig. 4.1). To reduce

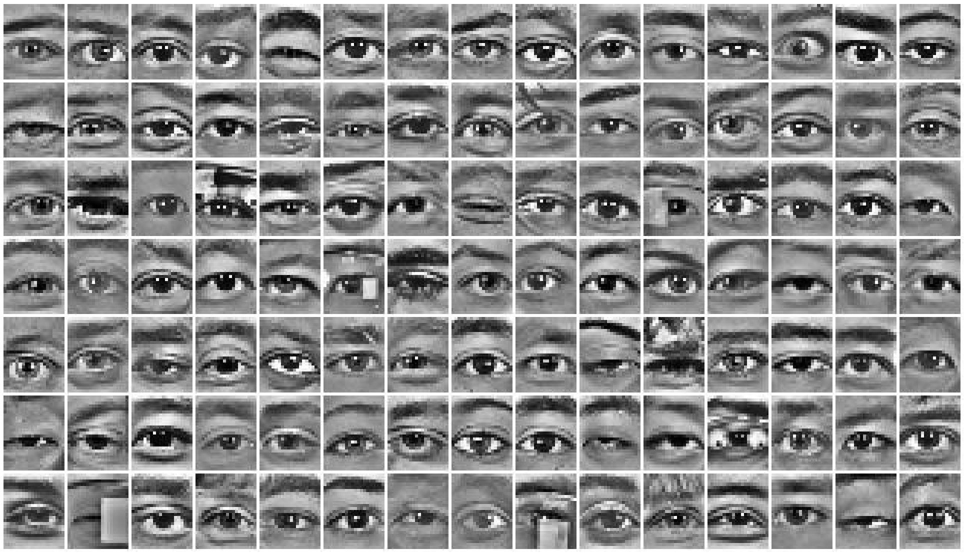


Figure 4.13: Weber-contrast normalized eye dataset. Compare this with Fig. 4.1.

the effect of outliers we normalize the contrast values using the following expression:

$$\begin{aligned} t_i &= \mathcal{F}(c_i\,;\,\beta), \\ &= \frac{1-\exp(-\beta c_i)}{1+\exp(-\beta c_i)}, \end{aligned} \tag{4.18}$$

where β is chosen such that for a predefined contrast value $c_i = c_{\text{def}}$, the normalized contrast t_i takes a value of 0.5.

The contrast normalization results on a subset of the eye and the non-eye datasets are shown in Figs. 4.13 and 4.14 respectively. We set $\sigma = 3$ for estimating the contrast and $c_{\text{def}} = 0.3$ for normalization. In Fig. 4.15 we show histograms of the pixel intensities for the two datasets before and after normalization. The normalization removes the DC signal for generic background patches. But for the eye ensemble, we see a small non-zero average, caused by pixel intensities being brighter than their neighbors more often than not. The range of intensities after normalization is compressed, that is the tails of the distributions are clipped. In general, we observe larger values of σ improve the performance of the detector, but the detector is less susceptible to the actual setting of c_{def}.

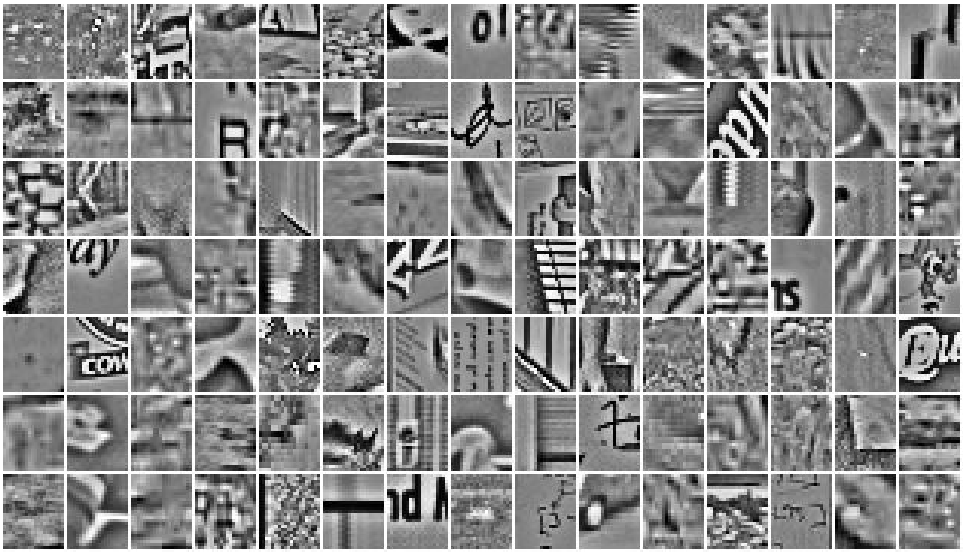


Figure 4.14: Weber-contrast normalized non-eye images (compare with Fig. 4.2).

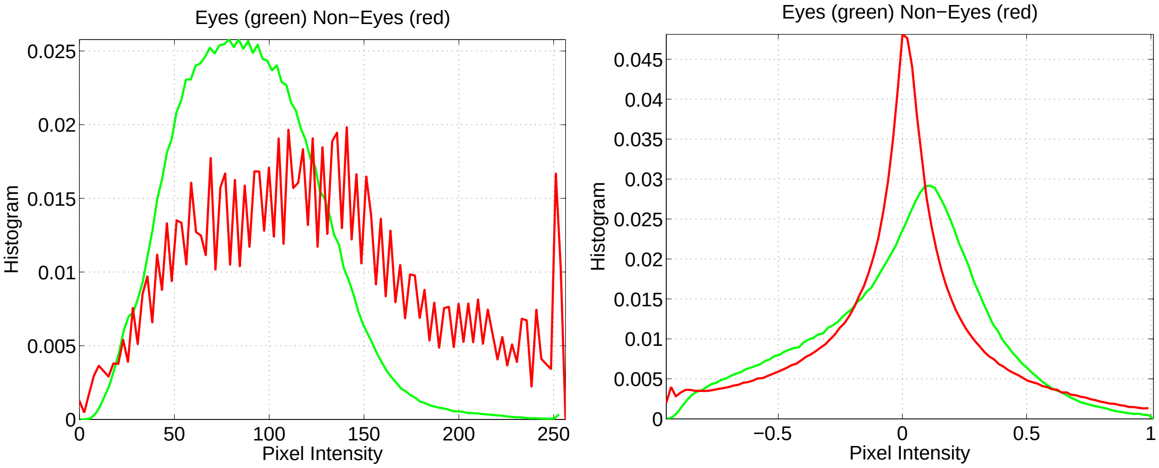


Figure 4.15: Weber-contrast normalization: histograms of the pixel intensities in the eye (green) and non-eye (red) datasets before (Left) and after normalization (Right).

4.4.2 S-PCA Representation: W, B

The S-PCA basis matrix trained on generic background patches is given by W, an $N \times N$ matrix, shown in Fig. 4.16. For an N-dimensional contrast-normalized image $\vec{t}$ the N-dimensional S-PCA coefficient vector is given by

$$\vec{d} = W^T \vec{t}.$$

Because the S-PCA basis look like wavelets, we abuse the notation slightly to call $\vec{d}$ a wavelet coefficient vector. Next, S-PCA is trained separately on the wavelet coefficients generated for the images in the object-specific ensemble (eg. eyes). For the following step, we build a low-dimensional representation for the wavelet coefficient vector using the leading M object-specific S-PCA basis vectors. In particular let B be the object-specific S-PCA basis matrix of size $N \times M$, then projecting the wavelet coefficient $\vec{d}$ gives

$$\vec{b} = B^T \vec{d}$$

and the wavelet coefficient vector can be reconstructed as

$$\begin{aligned} \vec{\hat{d}} &= B\vec{b} \\ &= BB^T \vec{d}, \end{aligned}$$

which is again N-dimensional. Because the basis matrix B resides in the wavelet space, it is hard to interpret it and hence in Fig. 4.17 we show the matrix: $W \times B$ obtained by pre-multiplying object-specific S-PCA basis B by the generic background patch S-PCA basis W. Notice, the basis $W \times B$ is sparse, spatially local and multi-scale.

4.4.3 Perceptual Distance Normalization

The coefficient vectors $\vec{d}$ and $\vec{\hat{d}}$ are now subjected to a perceptual distance normalization process. The idea is to normalize each wavelet coefficient by the pooled amplitude of wavelet coefficients tuned to similar spatial frequencies and similar spatial neighborhoods

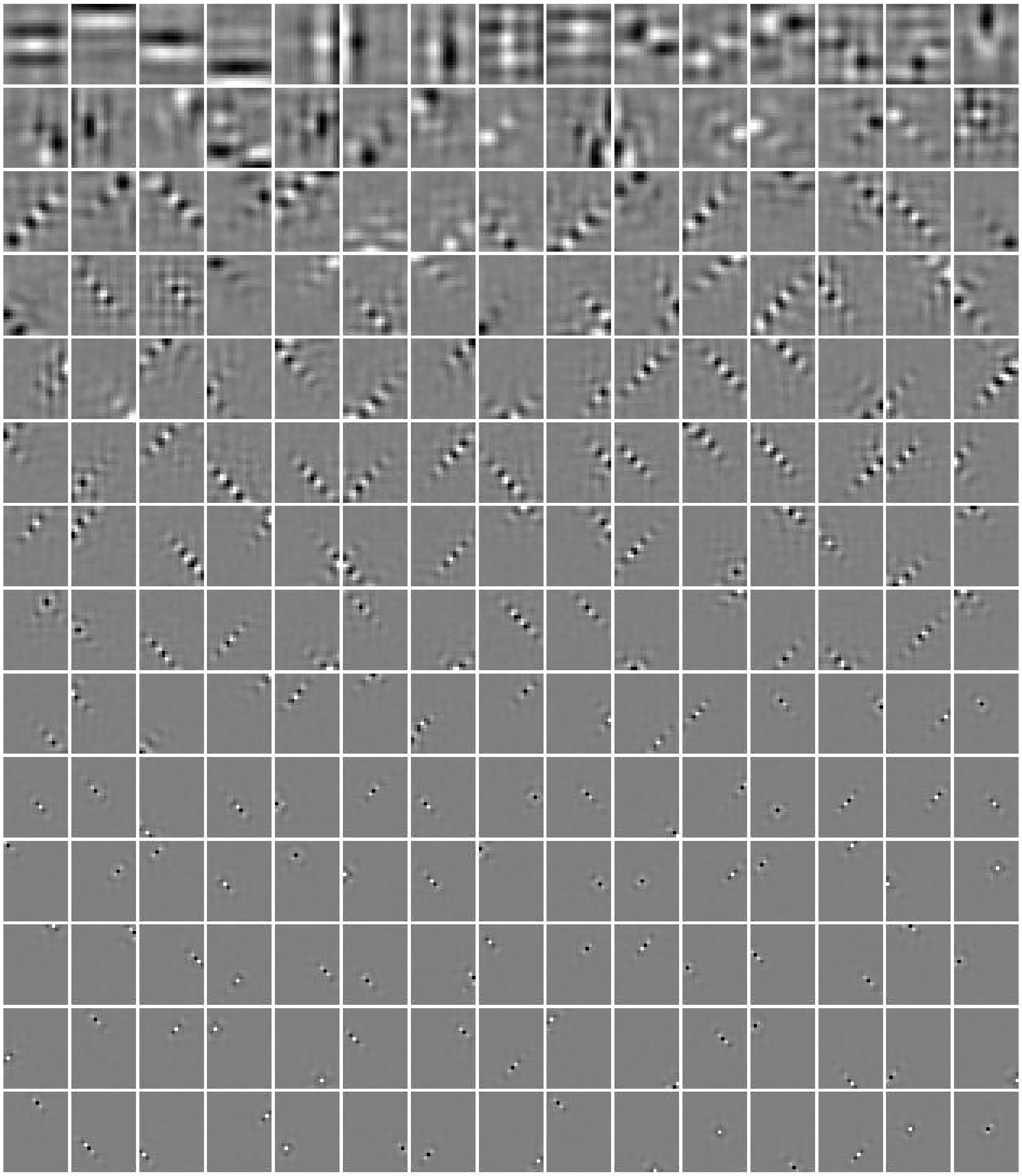


Figure 4.16: S-PCA basis for Weber-contrast normalized generic background patches.

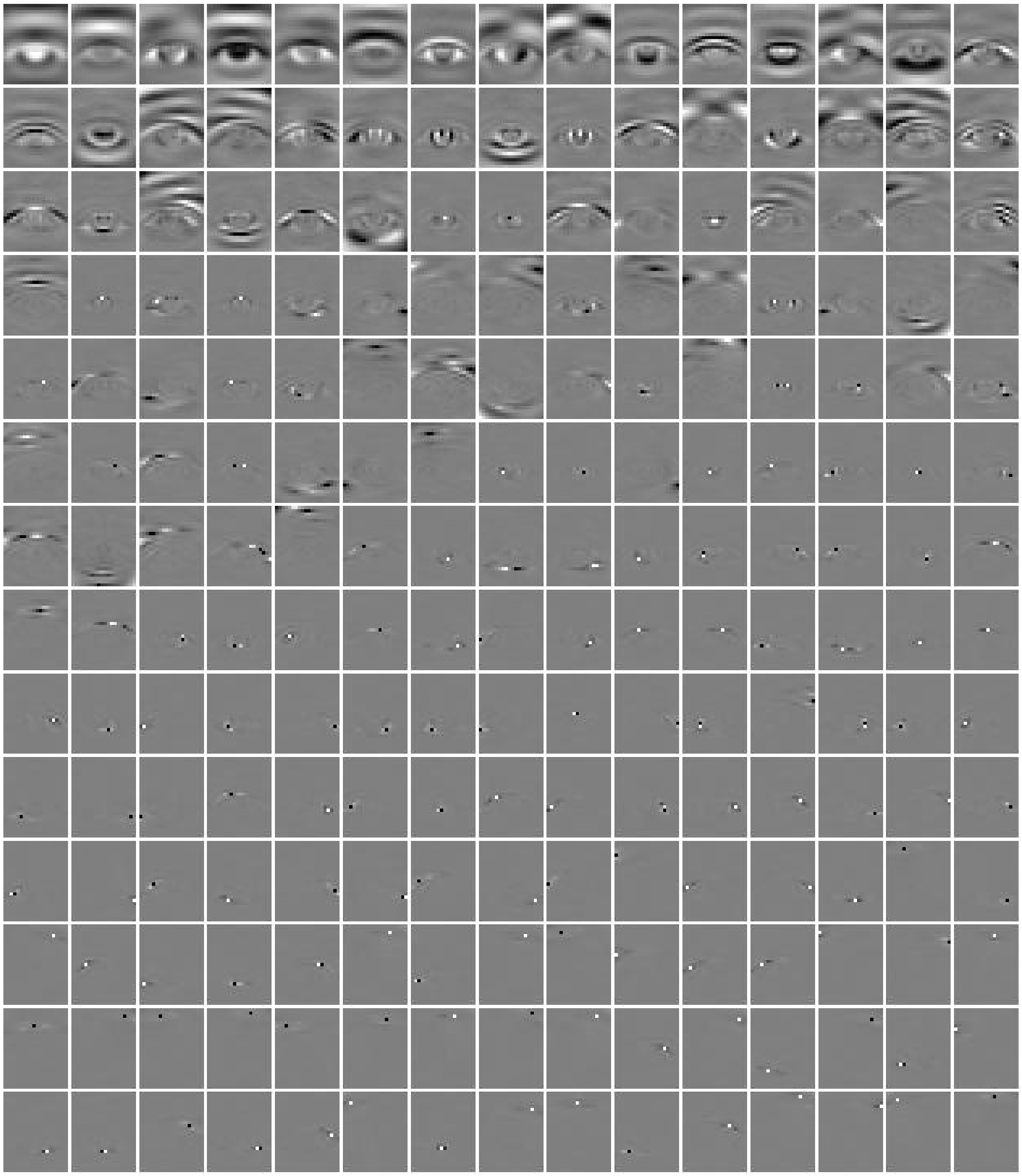


Figure 4.17: S-PCA basis for Weber-contrast normalized eye images.

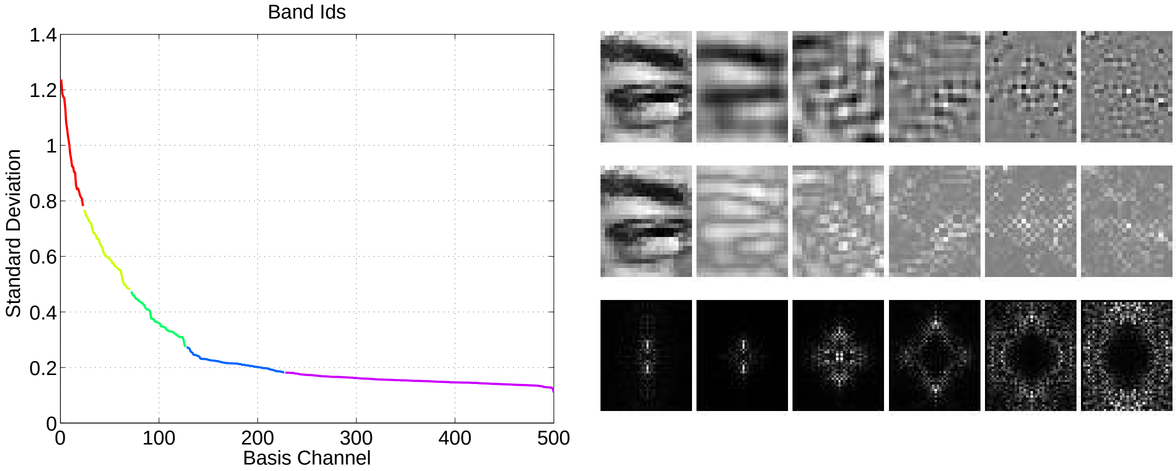


Figure 4.18: (Left) Partitioning the variance spectrum into five different subsets (shown here in different colors) each tuned to roughly similar spatial frequencies. (Right) An eye image and its bandpass components from the five different basis bands (Top), the corresponding amplitude maps (Middle) and the corresponding Fourier magnitude maps for the bandpass images (Bottom).

[18, 105]. Because S-PCA basis are learned from the data, as opposed to being hand-crafted, we need to find what the adjacent scales and orientations are for each S-PCA basis function in an automatic fashion. We outline a simple strategy next.

The spatial frequency tuning of the wavelets in W was observed to be related to the variance of the wavelet coefficients over the training set. Therefore by partitioning the variance spectrum we obtain subsets of wavelets tuned to roughly similar spatial frequencies. We automated this partitioning using K-means on the log variance spectrum. The result is shown in Fig. 4.18(Left), where the spectrum is partitioned into 5 groups, each drawn in a different color for the generic background patch ensemble.

To identify the local amplitude of image structure in each frequency band we form the amplitude image using a band of S-PCA basis indexed from l to h,

$$\vec{p} = \left| \sum_{l \leq k \leq h} \vec{w}_k d_k \right| . \tag{4.21}$$

Here $\vec{w}_k$ denotes the k^{th} basis vector, d_k is the corresponding wavelet coefficient value. In

Fig. 4.18(Right, Top) we show an eye image and the bandpass images that result from the summation in Eq. 4.21 in each of the five different basis bands. In Fig. 4.18(Right, Middle) we show the eye image along with its amplitude maps that result from the absolute operation over the bandpass images shown before, as given in Eq. 4.21. To provide a better intuition for the bandpass images, the images in Fig. 4.18(Right, Bottom) show the Fourier magnitudes of the bandpass images. To estimate the portion of this amplitude image within the spatial support of the k^{th} wavelet $\vec{w}_k$, we compute:

$$s_k = \left|\vec{w}_k^T\right| \vec{p}. \tag{4.22}$$

It can be shown that $s_k \geq |d_k|$ with the equality holding when $d_j = 0$ for $j \neq k$.

We can finally express the perceptual distance normalization ($\mathcal{PDN}$) of the k^{th} element of the coefficient vector as

$$z_k = \frac{d_k}{(s_k + \upsilon_{lh})}. \tag{4.23}$$

The constant υ_{lh} is a saturation parameter for the basis band indexed from l to h. It is determined empirically by processing random images with a predetermined noise level (= 4 gray levels) and measuring the statistics of the resulting S-PCA coefficients. In particular, the random images are contrast normalized and for each wavelet band a corresponding amplitude map is generated. The amplitude maps are then projected back into the wavelet space and the saturation constant υ_{lh} is set to the median value of the coefficients of the amplitude map in each wavelet band. The perceptual distance normalized coefficients of a wavelet coefficient vector $\vec{d}$ and its reconstruction $\vec{\hat{d}}$ are given by vectors $\vec{z}$ and $\vec{\hat{z}}$ respectively.

4.4.4 Detection Strategy

For the purpose of detection we measure two numbers: (1) the wavelet norm given by the L_1 norm of $\vec{z}$; and (2) the error norm given by the L_1 norm of the error vector $\vec{z} - \vec{\hat{z}}$.

We study the variation of these two numbers as a function of the increasing subspace dimensionality M, the number of columns in the basis matrix B shown in Fig. 4.13 (for relevant discussion see §4.4.2). We expect the error norm to be high for generic image patches because the subspace was built for the object-specific ensemble. Also, we expect that the higher the wavelet norm, the higher will be the error norm for generic image patches. In fact, as we discuss next, what we observe is that the generic background patches and the object-specific ensemble appear as two distinguishable clouds with a small amount of overlap. We next present results using a straightforward detection strategy.

4.4.5 Results

Eyes/Non-Eyes:

In Figs. 4.19 and 4.20, we show the results of applying the new detection method on the test set of eyes/non-eyes by varying $M = \{20, 50, 100, 200\}$. For clarity the plots have been scaled in such a way as to show all of the eye images (green points) at the expense of omitting a portion of the non-eye images (red points). As a detection strategy, we adopted a very simple approach of using a line aligned with the principal axis of the generic image patch cloud as shown in Fig. 4.21(Left) for the test set and Fig. 4.22(Left) for the training set. Points below the line are taken as positive detections. The ROC curve for a given M is obtained by adjusting the y-intercept of the line and these are shown for both the test and train sets. For example, in Fig. 4.21(Left) the two black lines correspond to two different points on the ROC curve ($M = 50$) shown in Fig. 4.21(Right).

The ROC curve in Fig. 4.21(Right) for the test set makes one thing very clear: the false positives can be kept very low, namely a value less than 0.8%, for a true acceptance rate of $\approx 95\%$ in a $M = 50$ dimensional subspace. This is a significant improvement over the previously reported detection model results (see Figs. 4.8 and 4.10). For comparison,

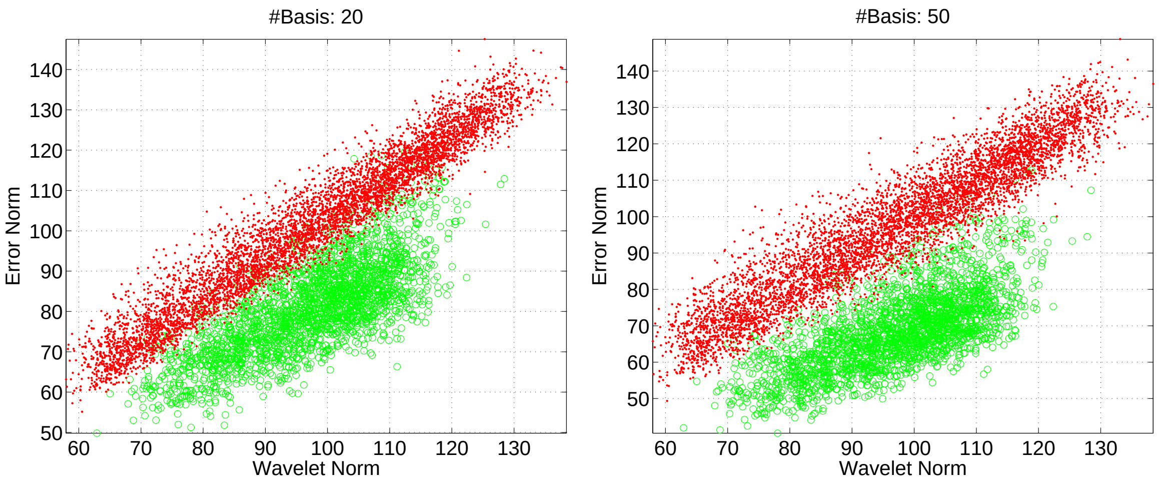


Figure 4.19: Separation of the eye and the non-eye clouds in the test set with the addition of (Left) 20 and (Right) 50 SPCA basis.

the ROC curve with $M = 50$ from Fig. 4.8 is reproduced here as a black line. In particular, for $M = 50$ the gain in false positive rate with $\mathcal{PDN}$ is nine-fold for a true detection rate of 95% and is 24-times better for a true detection rate of 90%. Some of the false positives admitted by the detection algorithm are shown in Fig. 4.23. The false positives were collected over the training and testing datasets and displayed in a single figure for convenience.

Faces/Non-Faces:

The MIT face database [6] consists of 2429/472 training/testing face images and 4548/23573 training/testing non-face images. Informally, most of the images in the training set are: cropped above the eyebrows, cropped just below the bottom lip, centered, roughly frontal views, and relatively well exposed. However, very few of the images in the test set appear to satisfy all these properties. We therefore created "mixed" training and testing sets by merging all these face images, adding the mirror symmetric versions, then randomly selecting half the dataset for mixed-training and the other half for mixed-testing. In Fig. 4.24(Left) we plot the perceptually normalized space for the newly created mixed-

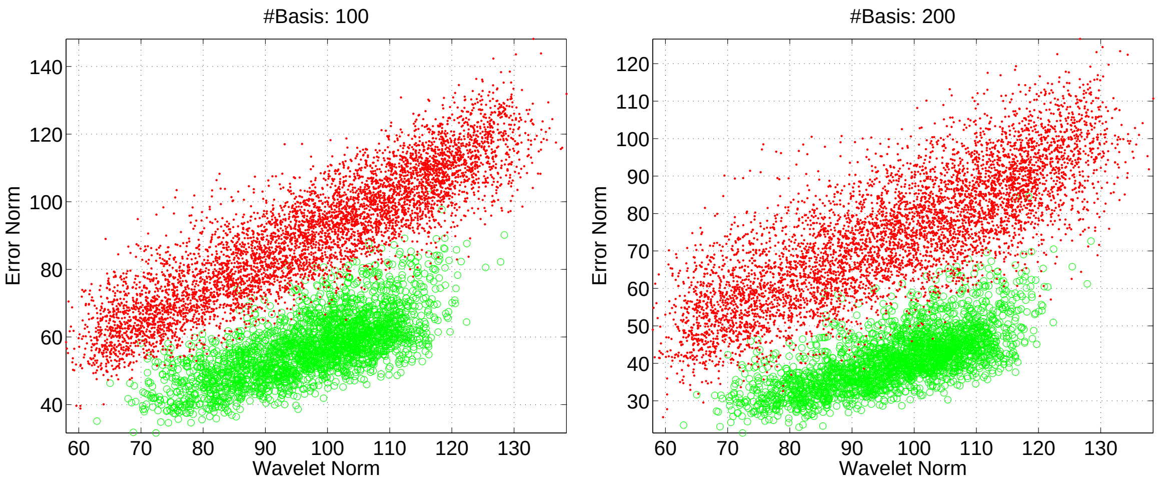


Figure 4.20: Separation of the eye and the non-eye clouds in the test set with the addition of (Left) 100 and (Right) 200 SPCA basis.

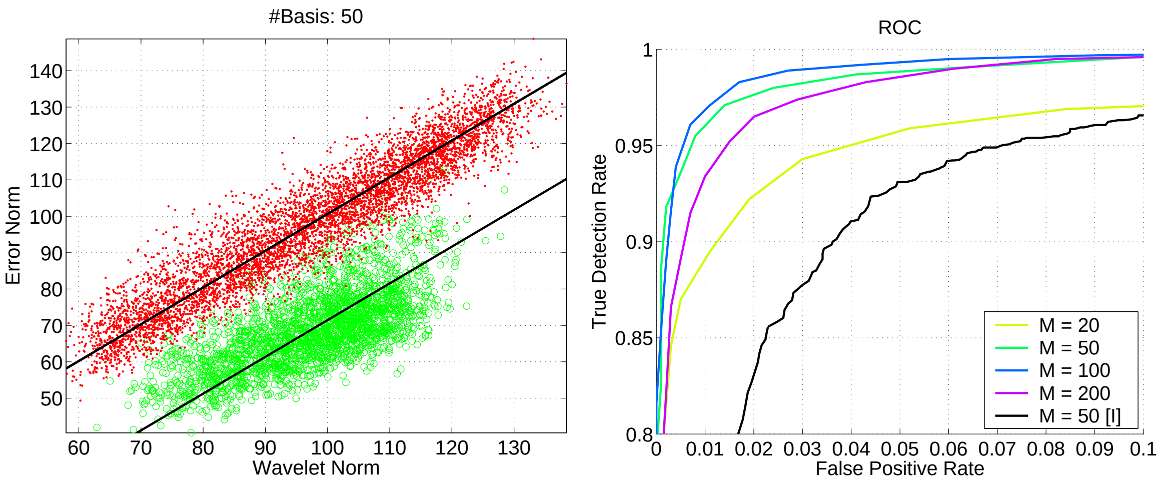


Figure 4.21: Detection results on the test set. (Left) Strategy is to adjust the y-intercept of a line aligned with the principal axis of the non-eye cloud (red). Points below the line are taken as positive detections. The two black lines correspond to two different points on the ROC curve for $M = 50$ (green) shown on the (Right). For comparison, the ROC curve for detector model I with $M = 50$ (from Fig. 4.8) is reproduced here on the (Right) as a black line (labeled $M = 50$ (I)).

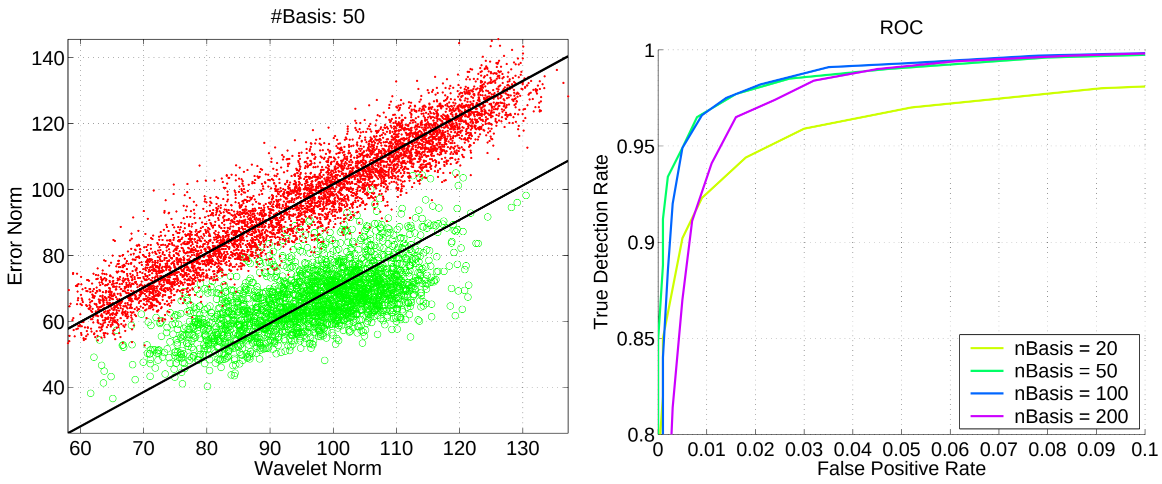
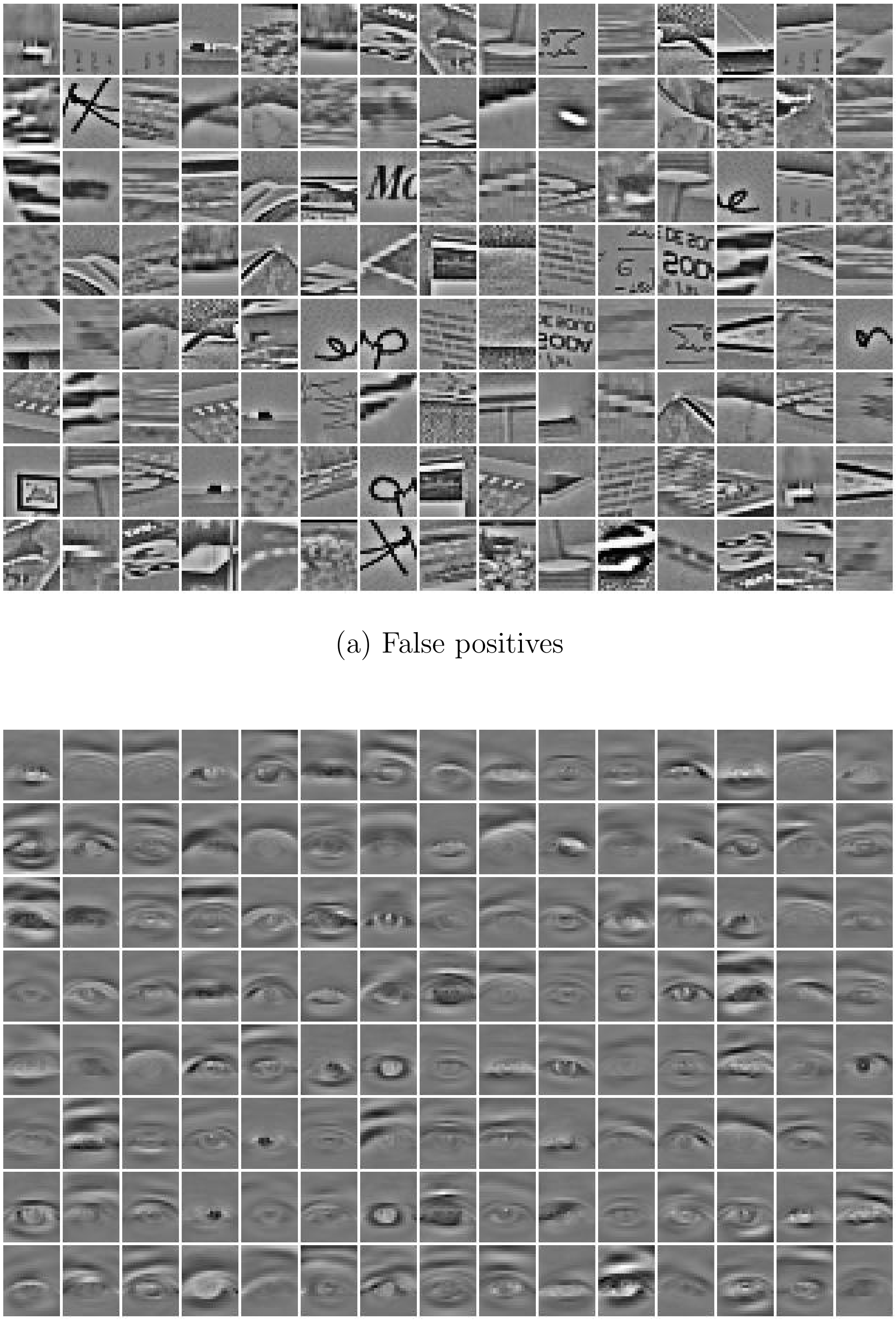


Figure 4.22: Detection results on the training set. (Left) Strategy is to adjust the y-intercept of a line aligned with the principal axis of the non-eye cloud (red). Points below the line are taken as positive detections. The two black lines correspond to two different points on the ROC curve for $M = 50$ shown in the (Right).

testing set, where the non-faces are indicated by red dots and the faces by the blue and green dots. In particular, the green dots indicate faces from the original training set and blue dots indicate faces from the original testing set. Using a subspace of $M = 50$ dimensions, for a false positive rate of 0.1% we observe 16% false negatives, out of which 96% belong to the original testing set. In fact, the face images in the original testing set make up 16% of the mixed dataset and, given the separation between the original training and original testing face images in the perceptually normalized space, this is not a surprise. In Fig. 4.24(Right) the black curve denotes the recognition rates obtained using just the original training set, and omitting the original test set, in a $M = 50$ dimensional subspace. The recognition rates are near perfect.

Comparison with Support Vector Machine (SVM)

We compare the performance of our detector with a *support vector machine* (SVM) classifier [115]. SVM is parameterized by a kernel function and a C value which is the cost



(a) False positives

(b) Reconstruction from a 50 dimensional subspace

Figure 4.23: (Top) False positives collected over the training and testing sets and displayed in one figure. (Top) Reconstruction results from a 50 dimensional subspace.

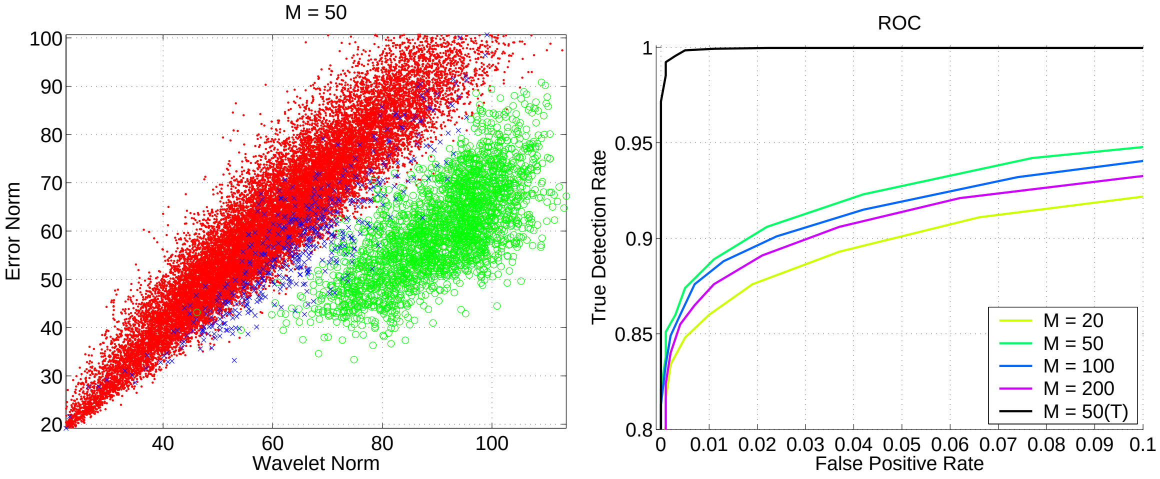


Figure 4.24: Detection results on the MIT face database. (Left) Characterizing test-faces (blue), train-faces (green) (which together constitute the mixed-test set) and non-faces in the test set (red) in the perceptually normalized space. (Right) ROC curves for the "mixed" test set. The black curve denotes recognition rate for just the train-faces in the "mixed" test set.

per unit violation of the classifier margin. We used a publicly available implementation of SVM [6]. We chose a Gaussian kernel and varied the σ parameter. The C value was set to 1, other values were tried but did not have a significant effect on the classifier.

On the eye dataset for different values of σ we observed a large variation in the total number of support vectors returned. In particular, for $\sigma = [3, 5, 10, 20]$ the number of support vectors returned on the training set are $[5267, 1564, 1140, 1616]$ respectively. Each support vector involves a dot product and hence, for a fair comparison the number of support vectors returned should be comparable to the number of inner products performed with the $\mathcal{PDN}$ model for a suitable choice of M. Thus, we selected $\sigma = [5, 10]$ and $M = [50, 100]$.

In Fig. 4.25(Left) we compare the detection results from SVM to our $\mathcal{PDN}$ model on the eye dataset. The green and the blue curves denote the use of $M = [50, 100]$ dimensional subspaces with the $\mathcal{PDN}$ model. The $\mathcal{PDN}$ graphs are identical to the

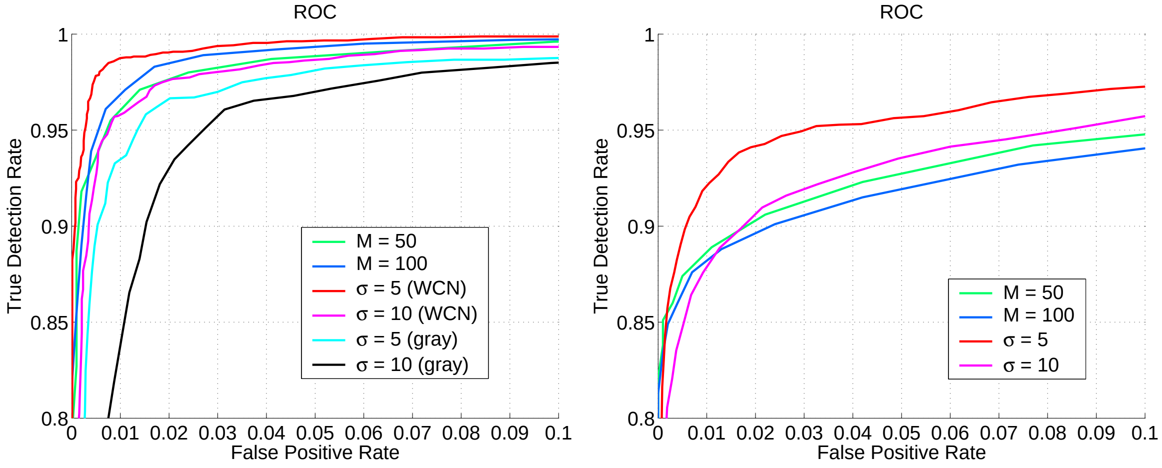


Figure 4.25: Comparing SVM with $\mathcal{PDN}$. (Left) ROC curves for the eye test set. The green and the blue curves correspond to $M = [50, 100]$ dimensional subspaces with the $\mathcal{PDN}$ model. The $\mathcal{PDN}$ graphs are identical to the ones shown in Fig. 4.21. The red and magenta curves show results from SVM, with $\sigma = [5, 10]$, on the contrast normalized ($\mathcal{WCN}$) dataset. The SVM performance on unnormalized images (gray) is given by the black and cyan curves. (Right) ROC curves for the "mixed" face testing set. The green and the blue curves correspond to $M = [50, 100]$ dimensional subspaces with the $\mathcal{PDN}$ model. The $\mathcal{PDN}$ graphs are identical to the ones shown in Fig. 4.24. The red and magenta curves show results from using SVM with $\sigma = [5, 10]$.

ones shown in Fig. 4.21. The red and magenta curves show results from using SVM with $\sigma = [5, 10]$. The performance of SVM with $\sigma = 10$ is similar to using $\mathcal{PDN}$ with $M = 50$. Increasing the total number of support vectors, i.e. reducing σ from 10 to 5, improves the performance of SVM. In addition, we tested the performance of the SVM on the original gray-level images (i.e. without $\mathcal{WCN}$), and the results are given by the black and cyan curves in Fig. 4.25(Left). It is clear that contrast normalization causes a significant improvement in the performance of SVM.

We ran a similar experiment on the mixed training and testing sets that we created for the MIT face database. The number of support vectors obtained on the mixed training set for $\sigma = [5, 10]$ are $[1549, 1434]$ respectively. In Fig. 4.25(Right) we compare the detection results from SVM to our $\mathcal{PDN}$ model on the mixed testing set. The green and the blue curves denote the use of $M = [50, 100]$ dimensional subspaces with the $\mathcal{PDN}$ model. The $\mathcal{PDN}$ graphs are identical to the ones shown in Fig. 4.24. The red and magenta curves show results from using SVM with $\sigma = [5, 10]$. The performance of SVM with $\sigma = 10$ is similar to using $\mathcal{PDN}$ with $M = 50$. The best detection result we obtained was for $\sigma = 5$, which required 1549 support vectors.

A detailed comparison of the different methods is beyond the scope of this thesis, for several reasons: (1) The $\mathcal{PDN}$ normalization is *not* optimal in terms of computational efficiency. It was designed for simplicity. The normalization of each wavelet should depend only on a few "neighbouring" wavelets, and there is likely to be a more efficient way to do this than by generating the amplitude map; (2) It is not clear that the SVM implementation we have used is optimal (e.g. see [107]). If neither method is optimal, a detailed comparison may not be very revealing. Perhaps, most interesting is the use of several detectors (e.g. an eye, a nose, a mouth, and a face detector) within a single system. For such a system the wavelet transform used in our approach is common to all detectors, and hence the cost of the wavelet transform, in terms of the underlying hardware, can be amortized.

4.5 Conclusion

Using PCA models, in place of S-PCA, we have observed detection results similar to the ones shown in Fig. 4.21 for both the object-specific and background ensembles with the same $\mathcal{PDN}$ formulation as provided in Eq. 4.21–4.23. The improved performance with $\mathcal{PDN}$ comes with a price, in that the images have to be represented in the full N-dimensional wavelet domain. However, we expect wavelet decomposition of signals to be a standard pre-processing tool. The simplicity of the detector in the wavelet domain is striking. In particular, after applying a linear model of eyes and doing perceptual normalization we can simply use the L_1 norm to separate classes.