Independent Component Representations for Face Recognition

Marian Stewart Bartlett
University of California San Diego
The Salk Institute
La Jolla, CA 92037
marni@salk.edu

H. Martin Lades
Livermore, CA
lades@earthlink.net

Terrence J. Sejnowski
University of California San Diego
Howard Hughes Medical Institute
The Salk Institute, La Jolla, CA 92037
terry@salk.edu

Abstract

In a task such as face recognition, much of the important information may be contained in the high-order relationships among the image pixels. A number of face recognition algorithms employ principal component analysis (PCA), which is based on the second-order statistics of the image set, and does not address high-order statistical dependencies such as the relationships among three or more pixels. Independent component analysis (ICA) is a generalization of PCA which separates the high-order moments of the input in addition to the second-order moments. ICA was performed on a set of face images by an unsupervised learning algorithm derived from the principle of optimal information transfer through sigmoidal neurons [6]. The algorithm maximizes the mutual information between the input and the output, which produces statistically independent outputs under certain conditions. ICA was performed on the face images under two different architectures, one which separated images across spatial location, and a second which separated the feature code across images. The first architecture provided a statistically independent basis set for the face images that can be viewed as a set of independent facial feature images. The second architecture provided a factorial code, in which the probability of any combination of features can be obtained from the product of their individual probabilities. Both ICA representations were superior to representations based on principal components analysis for recognizing faces across sessions and changes in expression. A classifier combining the two ICA algorithms recognized faces from a gallery of 450 at 91% accuracy across a changes in expression, and 89% across sessions, and 81% across sessions with a change of expression.

Manuscript in perparation. Please reference Bartlett, M.S. (1998) *Face Image Analysis by unsupervised learning and Redundancy Reduction.* Doctoral Dissertation, University of California, San Diego. An abbreviated version of this work appears in Bartlett, M.S., Lades, H.M. & Sejnowski, T.J. (1998). "Independent component representations for face recognition," in Rogowitz, B. & Pappas, T., (Eds.) *Proceedings of the SPIE Symposium on Electonic Imaging: Science and Technology, vol. 3299, Human Vision and Electronic Imaging III* 528-539.

1 Introduction

Horace Barlow has argued that redundancy provides knowledge [3]. Redundancy in the sensory input contains structural information about the environment. What is important for the perceptual system to detect is “suspicious coincidences,” new statistical regularities in the sensory input that differ from the environment to which it has been adapted. Bars and edges, for example, are locations in the visual input at which there is phase alignment across multiple spatial scales, and therefore constitute a “suspicious coincidence” in Barlow's formulation [4]. Learning mechanisms that encode the redundancy that is expected in the input and remove it from the output enable the system to more reliably detect these new regularities. Incoming sensory stimuli are automatically compared against the null hypothesis of statistical independence, and suspicious coincidences signaling a new causal factor can be more reliably detected. Learning such a transformation is equivalent to modeling the prior knowledge of the statistical dependencies in the input. Independent codes are advantageous for encoding complex objects that are characterized by high-order combinations of features, because the prior probability of any particular high-order combination is low.

Redundancy reduction has been discussed in relation to the visual system at several levels. A first-order redundancy is mean luminance. Adaptation mechanisms take advantage of this nonrandom feature by using it as an expected value, and expressing values relative to it [3]. Contrast gain control, which has been observed in V1 [11], takes account of changes in the variance of the input signals. Principal component analysis is a way of encoding second order dependencies in the input by rotating the axes to correspond to directions of maximum covariance. Principal component analysis provides a dimensionality-reduced code that separates the correlations in the input. Atick and Redlich [2] have argued for such compact, decorrelated representations as a general coding strategy for the visual system.

Some of the most successful algorithms for face recognition, such as "Eigenfaces" [26], "Holons" [9], and “Local Feature Analysis” [21] are based on learning mechanisms that are sensitive to the correlations in the face images. These are data-driven representations based on principal component analysis of the image set. Principal component analysis removes the correlations in the input, but does not address the high-order dependencies the images, such as the relationships among three or more pixels. Edges are an example of a high-order dependency in an image, as are elements of shape and curvature. In a task such as face recognition, much of the important information may be contained in the high-order relationships among the image pixels.

Second-order statistics capture the amplitude spectrum of images but not the phase[1] [10, 7]. The Fourier transform of the autocorrelation function of a signal is equal to its power spectrum (square of the amplitude spectrum). The remaining information that is not captured by the autocorrelation function, the high order statistics, corresponds to the phase spectrum. Spatial phase contains the structural information in images that drives human recognition much more strongly than the amplitude spectrum [19, 24]. A face image synthesized from the amplitude spectrum of face A and the phase spectrum of face B will be perceived as an image of face B.

Independent component analysis [8] is a generalization of principal component analysis that separates the high-order dependencies in the input, in addition to the second-order dependencies. Bell and Sejnowski [6] recently developed an algorithm for separating the statistically independent components of a dataset through unsupervised learning. This algorithm has proven successful for separating randomly mixed auditory signals (the cocktail party problem), and has recently been applied to separating EEG signals [14], fMRI signals [15], and finding image filters that give independent outputs from natural scenes [7].

[1] Given a translation invariant input, it is not possible to compute any statistics of the phase from the amplitude spectrum (Dan Ruderman, personal communication.)

This paper presents methods for representing face images for face recognition based on the statistically independent components of the image set. We performed independent component analysis on the image set under two architectures. The first architecture separated images across space (pixel location), and found a set of statistically independent source images for a set of face images. These source images comprised a set of independent basis images for the faces, and can be viewed as set of statistically independent facial feature images, in which the pixel values in one feature image cannot be predicted from the pixel values of the other feature images. The face representation consisted of the coefficients for the linear combination of independent basis images that comprised each face image. This architecture corresponded to the one used to perform blind separation of a mixture auditory signals [6] and to examine the independent sources of EEG [14] and fMRI data [15]. Under this architecture, the basis images were independent, but the coding variables that represented each face image were not. The second architecture separated pixels across images, and corresponded to the architecture used to find image filters that produced statistically independent outputs from natural scenes [7]. This architecture defined a set of statistically independent coding variables for representing the face images. In other words, we used ICA to find a factorial face code.

Face recognition performance was tested using the FERET database [23]. Face recognition performances using the ICA representations were benchmarked by comparing them to performances using principal component analysis, which is equivalent to the "Eigenface" representation [26, 22].

2 Independent Component Analysis (ICA)

Bell and Sejnowski's ICA algorithm is an unsupervised learning rule that was derived from the principle of optimal information transfer through sigmoidal neurons [12, 6]. Consider the case of a single input, x, and output, y, passed through a nonlinear squashing function, g.

$$u = wx + w_0 \qquad\qquad y = g(u) = \frac{1}{1 + e^{-u}} \tag{1}$$

As illustrated in Figure 1, the optimal weight w on x for maximizing information transfer is the one that best matches the probability density of x to the slope of the nonlinearity. The optimal w produces the flattest possible output density, which in other words, maximizes the entropy of the output.

The optimal weight is found by gradient ascent on the entropy of the output, y with respect to w. When there are multiple inputs and outputs, maximizing the joint entropy of the output encourages the individual outputs to move towards statistical independence. When the form of the nonlinear transfer function g is the same as the cumulative density functions of the underlying independent components (up to scaling and translation) it can be shown that maximizing the mutual information between the input X and the output Y also minimizes the mutual information between the u_i [16, 7]. Many natural signals, such as sound sources, have been shown to have a super-Gaussian distribution, meaning that the kurtosis of the probability distribution exceeds that of a Gaussian [6]. For mixtures of super-Gaussian signals, the logistic transfer function has been found to be sufficient to separate the signals [6].

The update rule for the weight matrix, W, for multiple inputs and outputs is given by

$$\Delta W = (I + y' u^T) W \tag{2}$$

where $y' = \frac{\partial}{\partial y_i} \frac{\partial y_i}{\partial u_i} = \frac{\partial}{\partial u_i} ln \frac{\partial y_i}{\partial u_i}$.

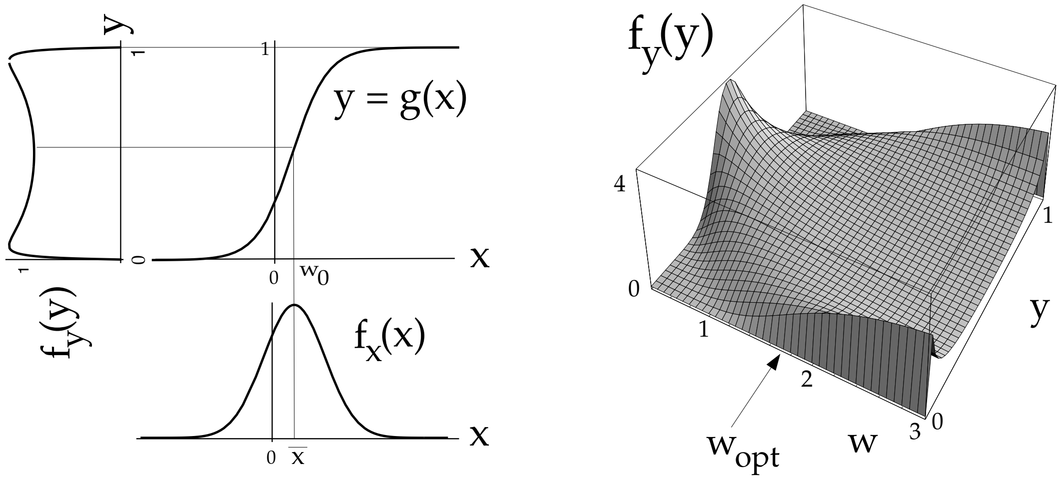

Figure 1: Optimal information flow in sigmoidal neurons. The input x is passed through a nonlinear function, $g(x)$. The information in the output density $f_y(y)$ depends on matching the mean and variance of $f_x(x)$ to the slope and threshold of $g(x)$. Right: $f_y(y)$ is plotted for different values of the weight, w. The optimal weight, w_{opt} transmits the most information. Figure from Bell & Sejnowski (1995), reprinted with permission from *Neural Computation*, copyright 1995, MIT Press.

We employed the logistic transfer function, $g(u) = \frac{1}{1+e^{-u}}$, giving $y' = (1 - 2y_i)$. The algorithm includes a "sphering" step prior to learning [7]. The row means are subtracted from the dataset, X, and then X is passed through the zero-phase whitening filter, W_z, which is twice the inverse square root of the covariance matrix:

$$W_z = 2 * \langle XX^T \rangle^{-\frac{1}{2}}. \tag{3}$$

This removes both the first and the second-order statistics of the data; both the mean and covariances are set to zero and the variances are equalized. The full transform from the zero-mean input was calculated as the product of the sphering matrix and the learned matrix, $W_I = W * W_Z$. The pre-whitening filter in the ICA algorithm has the Mexican-hat shape of retinal ganglion cell receptive fields which remove much of the variability due to lighting [7].

The difference between ICA and PCA is illustrated as follows. Consider a set of data points derived from two underlying distributions as shown in Figure 2. Principal component analysis encodes second order dependencies in the data by rotating the axes to correspond to directions of maximum covariance. PCA models the data as a multivariate Gaussian and would place an orthogonal set of axes such that the projections of the two distributions would be completely overlapping. Independent component analysis does not constrain the axes to be orthogonal, and attempts to place them in the directions of statistical dependencies in the data. Each weight vector in ICA attempts to encode a portion of the dependencies in the input, so that the dependencies are removed from between the elements of the output. The projection of the two distributions onto the ICA axes would have less overlap, and the output distributions of the two weight vectors would be kurtotic [10].

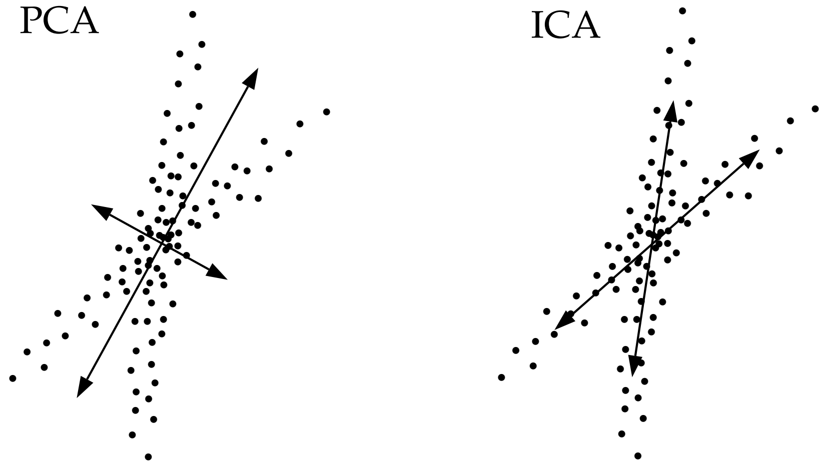

Figure 2: Example 2-D data distribution and corresponding principal component and independent component axes. Figure inspired by Lewicki & Sejnowski (submitted).

3 Independent Component Representations of Face Images

3.1 Statistically independent basis images

To find a set of statistically independent basis images for the set of faces, we separated the independent components of the face images according to the image synthesis model of Figure 3. The face images in X were assumed to be a linear mixture of an unknown set of statistically independent source images S, where A is an unknown mixing matrix. The sources were recovered by a matrix of learned filters, W_I, which produced statistically independent outputs, U. This synthesis model is related to the one used to perform blind separation on an unknown mixture of auditory signals [6] and to separate the sources of EEG signals [14] and fMRI images [15].

The images comprised the rows of the input matrix, X. With the input images in the rows of X, the ICA outputs in the rows of $W_I X = U$ were also images, and provided a set of independent basis images for the faces (Figure 4). These basis images can be considered a set of statistically independent facial features, where the pixel values in each feature image cannot be predicted from the pixel values in the other feature images. The ICA representation consisted of the coefficients for the linear combination of independent basis images in U that comprised each face image, as shown in Figure 4. The matrix of coefficients, B, was obtained from the mixing matrix $A \stackrel{\Delta}{=} W_I^{-1}$.

The number of independent components found by the ICA algorithm corresponds to the dimensionality of the input. In order to have control over the number of independent components extracted by the algorithm, instead of performing ICA on the n original images, we performed ICA on a set of m linear combinations of those images, where $m < n$. Recall that the image synthesis model assumes that the images in X are a linear combination of a set of unknown statistically independent sources. The image synthesis model is unaffected by replacing the original images with some other linear combination of the images.

Adopting a method that has been applied to independent component analysis of fMRI data [15], we chose for these linear combinations the first m principal component eigenvectors of the image set. Principal component analysis on the image set in which the pixel locations are treated as observations and each face image a measure, gives the linear combination of the parameters (images) that accounts for the maximum variability in the observations (pixels).

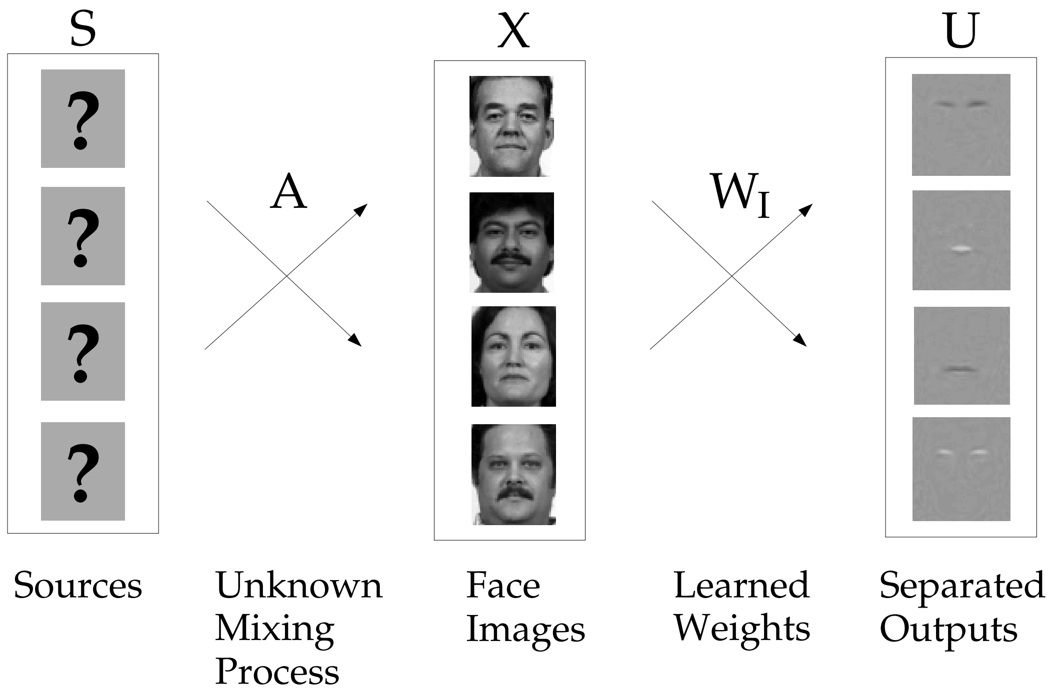

Figure 3: Image synthesis model. For finding a set of independent component images, the images in X are considered to be a linear combination of statistically independent basis images, S, where A is an unknown mixing matrix. The basis images were recovered by a matrix of learned filters, W_I, that produced statistically independent outputs, U.

The use of PCA vectors in the input did not throw away the high-order relationships. These relationships still existed in the data but were not separated.

Let P_m denote the matrix containing the first m principal component axes in its columns. We performed ICA on P_m^T, producing a matrix of m independent source images in the rows of U. The coefficients, $\mathbf{b}$, for the linear combination of basis images in U that comprised the face images in X were determined as follows:

The principal component representation of the set of zero-mean images in X based on P_m is defined as $R_m = X * P_m$. A minimum squared error approximation of X is obtained by $X_{rec} = R_m * P_m^T$.

The ICA algorithm produced a matrix $W_I = W * W_Z$ such that

$$W_I * P_m^T = U \qquad \Rightarrow \qquad P_m^T = W_I^{-1} U. \tag{4}$$

Therefore

$$X_{rec} = R_m * P_m^T \qquad \Rightarrow \qquad X_{rec} = R_m * W_I^{-1} U. \tag{5}$$

where W_Z was the sphering matrix defined in Equation 3. Hence the rows of $R_m * W_I^{-1}$ contained the coefficients for the linear combination of statistically independent sources U that comprised X_{rec}, where X_{rec} was a minimum squared error approximation of X, just as in PCA. The independent component representation of the face images based on the set of m statistically independent feature images, U was therefore given by the rows of the matrix

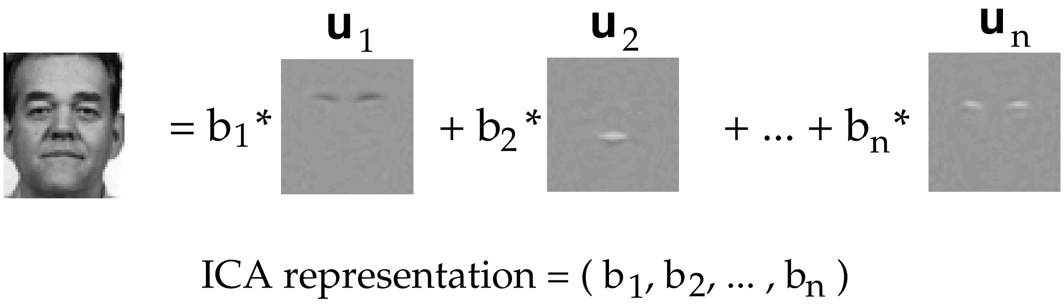

Figure 4: The independent basis image representation consisted of the coefficients, $\mathbf{b}$, for the linear combination of independent basis images, $\mathbf{u}$, that comprised each face image $\mathbf{x}$.

$$B = R_m * W_I^{-1}. \tag{6}$$

A representation for test images was obtained by using the principal component representation based on the training images to obtain $R_{test} = X_{test} * P_m$, and then computing $B_{test} = R_{test} * W_I^{-1}$.

3.2 Independence in face space versus pixel space

The analysis in Section 3.1 produced statistically independent basis images. The ICA algorithm separated images across pixel location (see Figure 5 Top Left.) Each pixel location was an observation which took on different grayvalues for each of the faces. This is illustrated in Figure 5 (Bottom Left), in which the pixels are plotted according to their grayvalues for each face image. ICA in Architecture 1 finds weight vectors in the directions of statistical dependencies in the population of face images over the pixel locations. Projecting the data onto these weights produced a set of independent images, where the pixel grayvalues in one image could not be predicted from the grayvalues of the other images. These independent images spanned the space of the face images, and each face was represented by the coefficients for the linear combination of these independent template images that comprised each face image.

The correspondence of the ICA-basis representation (ICA1) with the principal component representation is that the Eigenface images are pixelwise uncorrelated. PCA represents faces as a linear combination of uncorrelated template images, whereas ICA represents faces as a linear combination of independent template images. The Eigenfaces are uncorrelated because 1. They are orthogonal by definition, and 2. The Eigenfaces are themselves a set of PCA coefficients obtained by considering each each pixel location an observation and each face image a measure. Let v_i denote the eigenvectors of X^TX. The eigenvectors u_i of XX^T can be obtained by Xv_i [26].

Although the basis images obtained in Architecture 1 were spatially independent, the coefficients that coded each face were not. By altering the architecture of the independent component analysis, we defined a second representation in which the *coefficients* were statistically independent. In other words, the second ICA architecture found a factorial code for the face images. The alteration in architecture corresponded to transposing the input (see Figure 5 Top Right). Each face image was treated as an observation coded by the grayvalues at each of the pixel locations. ICA in Architecture 2 finds weight vectors in the directions of statistical dependencies in the face code across the population of faces. Projecting the data onto these weights produced a set of independent coding variables to replace "pixel location", where the value of any given coding variable could not be predicted from the

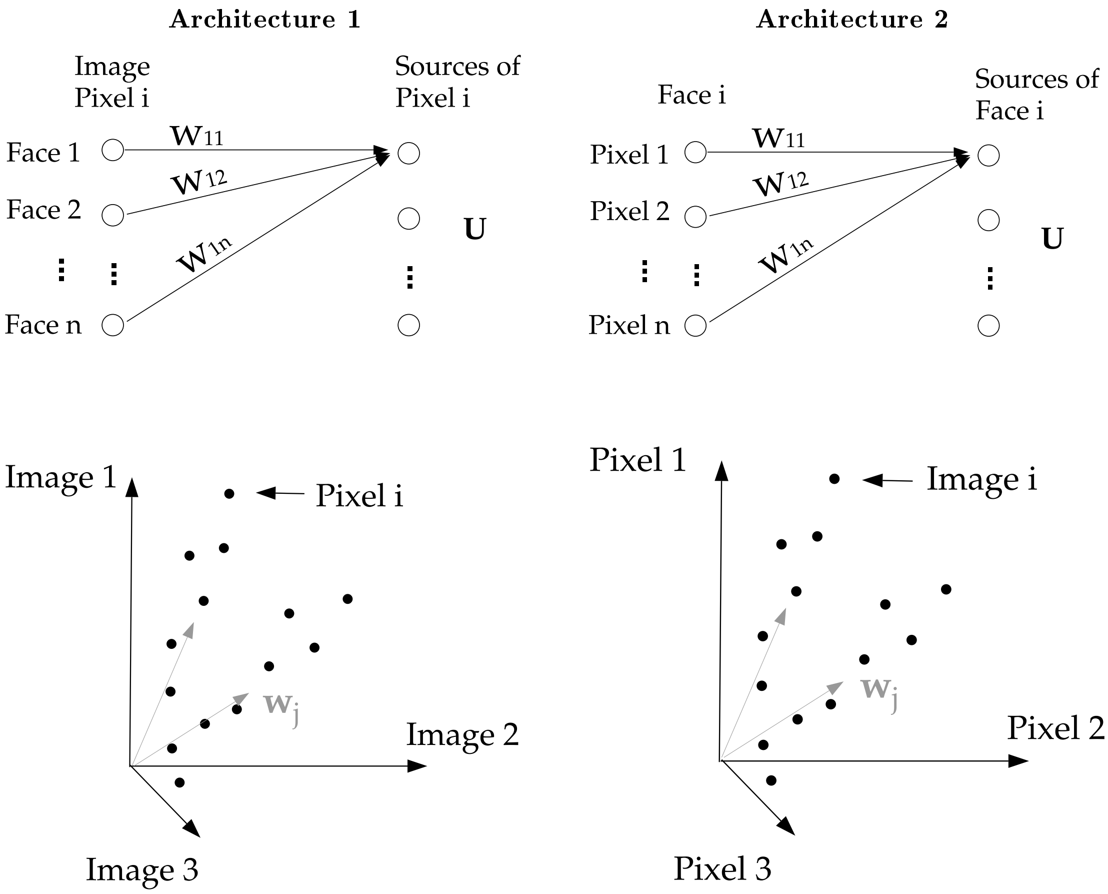

Figure 5: Two architectures for performing ICA on images. LEFT: Architecture for finding statistically independent basis images. Top Left: Performing source separation on the face images produced independent component images in the rows of U. Bottom left: The gray-values at pixel location i are plotted for each face image. ICA in architecture 1 finds weight vectors in the directions of statistical dependencies among the pixel locations. RIGHT: Architecture for finding a factorial code. Top Right: Performing source separation on the pixels produced a factorial code in the columns of the output matrix, U. Bottom Right: Each face image is plotted according to the grayvalues taken on at each pixel location. ICA in architecture 2 finds weight vectors in the directions of statistical dependencies among the face i mages.

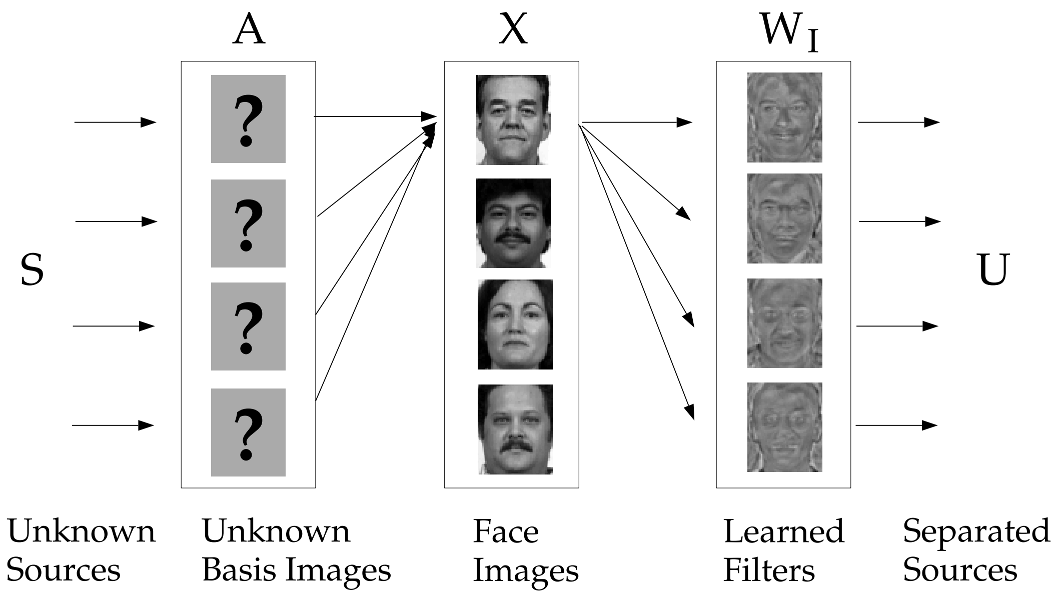

Figure 6: Image synthesis model for Architecture 2, based on Olshausen & Field (1996) and Bell & Sejnowski (1997). Each image in the dataset was considered to be a linear combination of underlying basis images in the matrix A. The basis images were each associated with a set of independent "causes", given by a vector of coefficients in S. The causes were recovered by a matrix of learned filters, W_I, which attempts to invert the unknown basis functions to produce statistically independent outputs, U.

other coding variables. Each face was represented by the values taken on by this new set of independent coding variables.

The correspondence of the ICA-factorial representation (ICA2) with the principal component representation is direct. The principal component coefficients constitute an uncorrelated face code, whereas the ICA2 coefficients constitute an independent face code.

3.3 A factorial face code

A factorial face code was obtained by performing source separation on the face images under Architecture 2. The alteration in architecture corresponded to transposing the input matrix X such that the images were in columns and the pixels in rows (see Figure 5 Right). Under this architecture, the filters (rows of W_I) were images, as were the columns of $A = W_I^{-1}$. The columns of A formed a new set of basis images for the faces, and the coefficients for reconstructing each face were contained in the columns of the ICA outputs, U.

Architecture 2 is associated with the image synthesis model in Figure 7. This model is similar to the the model in Figure 3, except that we now assume that the faces are comprised of a set of independent coefficients, S, for a set of basis images in A, whereas in the model in Figure 3 it was the other way around: The independent sources S were basis images, and the coefficients were in A. This model was based on the image synthesis model of Olshausen and Field [18], and was also employed by Bell and Sejnowski [7] to find image filters that produced statistically independent outputs from natural scenes. The ICA algorithm attempts to recover the source coefficients by finding a set of filters W_I that produce statistically independent outputs, U.

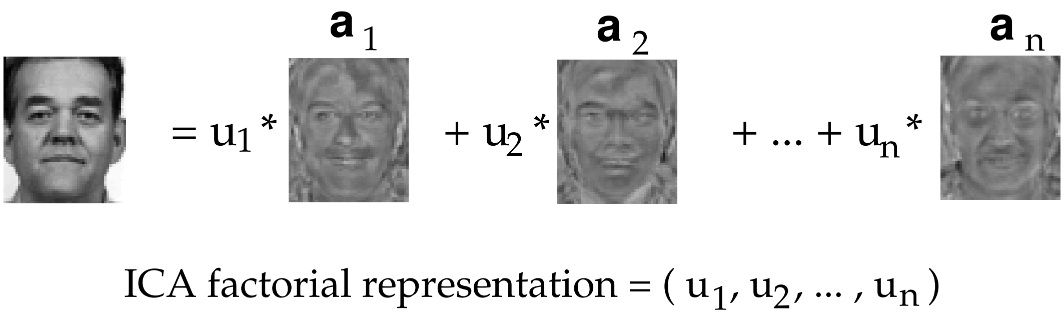

Figure 7: The factorial code representation consisted of the independent coefficients, $\mathbf{u}$, for the linear combination of basis images in A that comprised each face image $\mathbf{x}$.

The columns of the ICA output matrix, $W_I X = U$ provided a factorial code for the training images in X. Each column of U contained the coefficients of the the basis images in A for reconstructing each image in X (Figure 7). The representational code for test images was found by $W_I X_{test_i} = U_{test_i}$, where X_{test} was the zero-mean matrix of test images, and W_I was the weight matrix found by performing ICA on the training images.

4 Face Recognition Performance

Face recognition performance was evaluated for the two ICA representations using the FERET face database [23]. The data set contained images of 425 individuals. There were up to four frontal views of each individual: A neutral expression and a change of expression from one session, and a neutral expression and change of expression from a second session that occurred up to two years after the first. Examples of the four views are shown in Figure 8. The two algorithms were trained on a single frontal view of each individual. The training set was comprised of 50% neutral expression images and 50% change of expression images. The algorithms were tested for recognition under three different conditions: same session, different expression; different session, same expression; and different session, different expression (see Table 1).

Figure 8: Example from the FERET database of the four frontal image viewing conditions: Neutral expression and change of expression from Session 1; Neutral expression and change of expression from Session 2.

Coordinates for eye and mouth locations were provided with the FERET database. These coordinates were used to center the face images, and then crop and scale them to 60×50 pixels. Scaling was based on the area of the triangle defined by the eyes and mouth. The luminance was normalized by linearly rescaling each image to the interval $[0, 255]$. For the subsequent analyses, the rows of the images were concatenated to produce 1×3000 dimensional vectors.

Image Set	Condition		Number of Images
Training Set	Session I	50% neutral 50% other	425
Test Set 1	Same Session	Different Expression	421
Test Set 2	Different Session	Same Expression	45
Test Set 3	Different Session	Different Expression	43

Table 1: Image sets used for training and testing.

4.1 Independent basis architecture

The principal component axes of the Training Set were found by calculating the eigenvectors of the pixelwise covariance matrix over the set of face images. Independent component analysis was then performed on the first 200 of these eigenvectors, P_{200}, where the first 200 principal components accounted for over 98% of the variance in the images. The 1×3000 eigenvectors in P_{200} comprised the rows of the 200×3000 input matrix X. The input matrix X was sphered according to Equation 3, and the weights, W, were updated according to Equation 2 for 1600 iterations. The learning rate was initialized at 0.001 and annealed down to 0.0001. Training took 90 minutes on a Dec Alpha 2100a. Following training, a set of statistically independent source images were contained in the rows of the output matrix U.

Figure 9 shows a subset of 25 source images. A set of principal component basis images (PCA axes), are shown in Figure 10 for comparison. The ICA basis images were more spatially local than the principal component basis images. Two factors contribute to the local property of the ICA basis images: The majority of the statistical dependencies were in spatially proximal image locations, and ICA algorithm produces sparse outputs [7].

These source images in the rows of U were used as the basis of the ICA representation. The coefficients for the zero-mean training images were contained in the rows of $B = R_{200} * W_I^{-1}$ according to Equation 6, and coefficients for the test images were contained in the rows of $B_{test} = R_{Test} * W_I^{-1}$ where $R_{Test} = Xtest_i * P_{200}$.

Face recognition performance was evaluated for the coefficient vectors $\mathbf{b}$ by the nearest neighbor algorithm. Coefficient vectors in each test set were assigned the class label of the coefficient vector in the training set that was most similar as evaluated by the cosine of the angle between them:

$$d = \frac{\mathbf{b}_{test} \cdot \mathbf{b}_{train}}{\| \mathbf{b}_{test} \| \| \mathbf{b}_{train} \|}. \quad (7)$$

Face recognition performance for the principal component representation was evaluated by an identical procedure, using the principal component coefficients contained in the rows of R_{200}. Figure 11 gives face recognition performance with both the ICA and the PCA based representations. Recognition performance is also shown for the PCA based representation using the first 20 principal component vectors, which was a recent Eigenface representation used by Pentland, Moghaddam and Starner [22]. Best performance for PCA was obtained using 200 coefficients. Excluding the first 1, 2, or 3 principal components did not improve PCA performance, nor did selecting intermediate ranges of components from 20 through 200. There was a trend for the ICA representation to give superior face recognition performance to the PCA representation with 200 components. The difference in performance was marginally significant for Test Set 3 ($Z = 1.94, p = 0.05$). The difference in performance between the ICA representation and the Eigenface representation with 20 components was statistically significant over all three test sets ($Z = 2.5, p < 0.05$) for Test sets 1 and 2, and ($Z = 2.4, p < 0.05$) for Test Set 3.

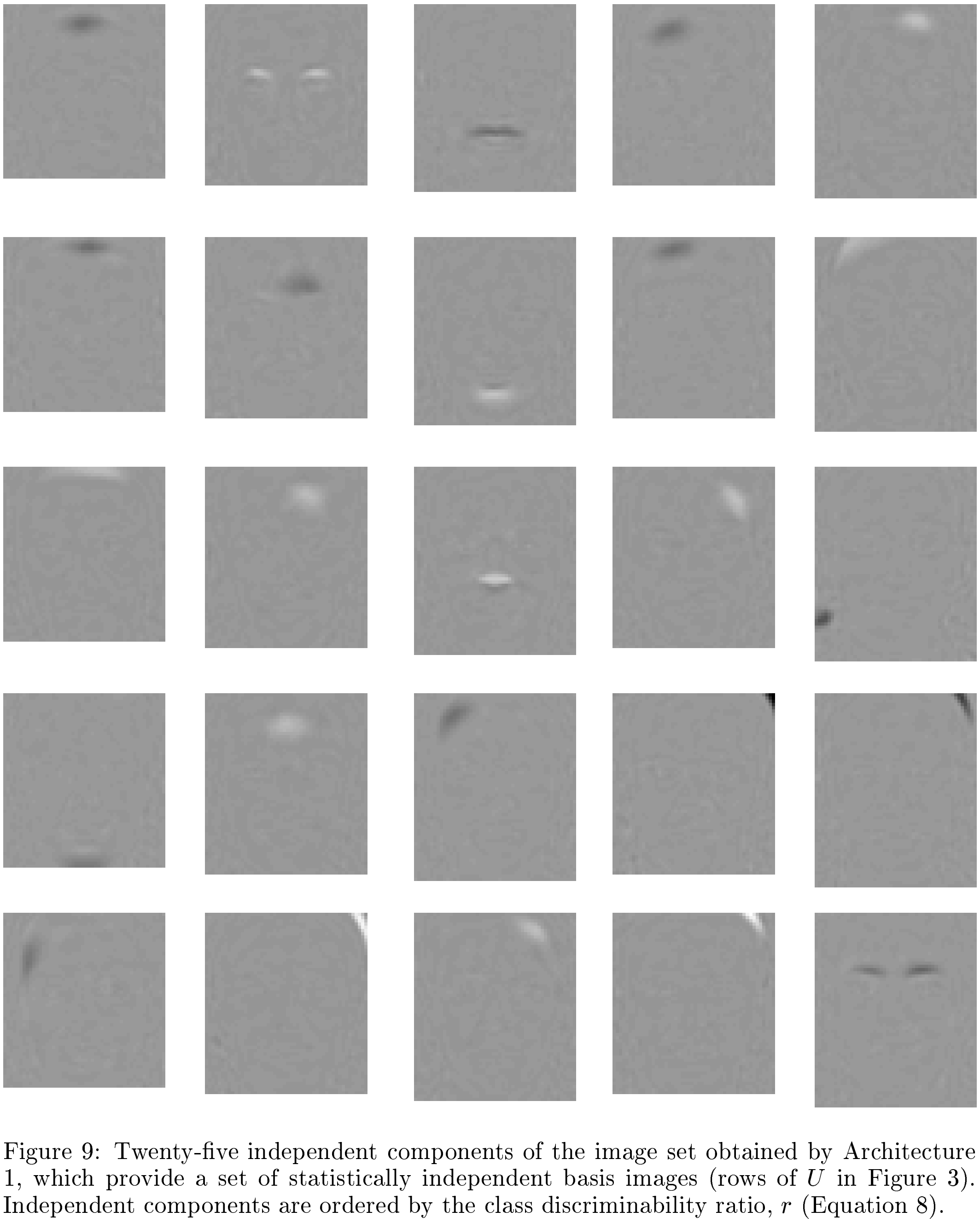

Figure 9: Twenty-five independent components of the image set obtained by Architecture 1, which provide a set of statistically independent basis images (rows of U in Figure 3). Independent components are ordered by the class discriminability ratio, r (Equation 8).

Figure 10: First 25 principal component axes of the image set (columns of P), ordered left to right, top to bottom, by the magnitude of the corresponding eigenvalue.

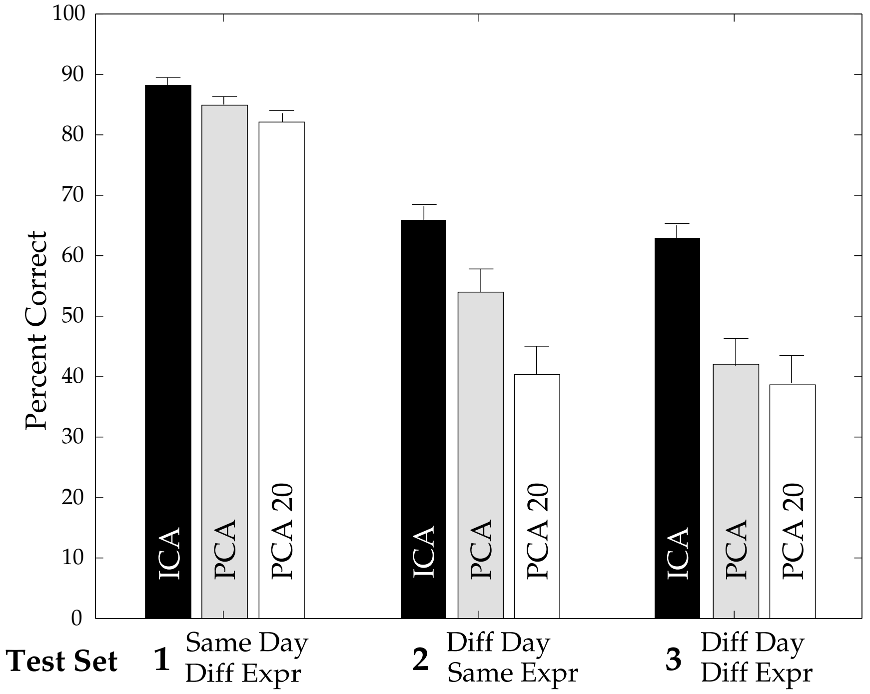

Figure 11: Percent correct face recognition for the ICA representation using 200 independent components, the PCA representation using 200 principal components, and the PCA representation using 20 principal components. Groups are performances for Test Set 1, Test Set 2, and Test Set 3. Error bars are one standard deviation of the estimate of the success rate for a Bernoulli distribution.

Face recognition performances for the PCA and ICA representations were next compared by selecting subsets of the 200 components by class discriminability. Let $\overline{x}$ be the overall mean of a coefficient b_k across all faces, and $\overline{x}_j$ be the mean for person j. For both the PCA and ICA representations, we calculated the ratio of between-class to within-class variability, r, for each coefficient:

$$r = \frac{\sigma_{between}}{\sigma_{within}} \tag{8}$$

where $\sigma_{between} = \sum_j (\overline{x}_j - \overline{x})^2$ is the variance of the j class means, and $\sigma_{within} = \sum_j \sum_i (x_{ij} - \overline{x}_j)^2$ is the sum of the variances within each class.

The class discriminability analysis was carried out using the 43 subjects for which four frontal view images were available. The ratios r were calculated separately for each test set, excluding the images from the corresponding test set from the analysis. Both the PCA and ICA coefficients were then ordered by the magnitude of r. Figure 12 (Top) compares the discriminability of the ICA coefficients to the PCA coefficients. The ICA coefficients consistently had greater class discriminability than the PCA coefficients.

Face classification performance was compared using the k most discriminable components of each representation. Figure 12 (Bottom) shows the best classification performance obtained for the PCA and ICA representations, which was with the 60 most discriminable components for the ICA representation, and the 140 most discriminable components for the PCA representation. Selecting subsets of coefficients by class discriminability improved

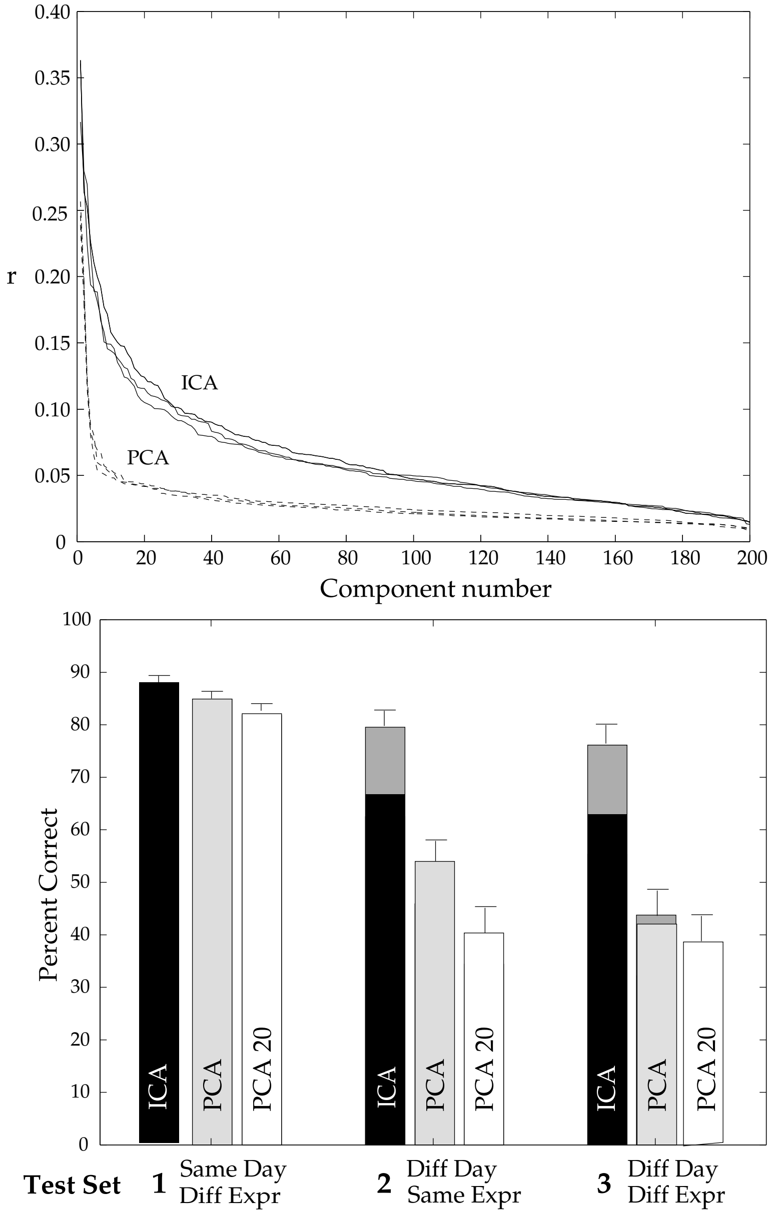

Figure 12: Selection of components by class discriminability. Top: Discriminability of the ICA coefficients (solid lines) and discriminability of the PCA components (dotted lines) for the three test cases. Components were sorted by the magnitude of r. Bottom: Improvement in face recognition performance for the ICA and PCA representations using subsets of components selected by the class discriminability r. The improvement is indicated by the gray segments at the top of the bars.

the performance of the ICA representation, but had little effect on the performance of the PCA representation. The ICA representation again outperformed the PCA representation. The difference in recognition performance between the ICA and PCA representations was significant for Test Set 2 and Test Set 3, the two conditions that required recognition of images collected on a different day from the training set ($Z = 2.9, p < .05; Z = 3.4, p < .01$), respectively.

4.2 Factorial code architecture

ICA was next performed on the face images using Architecture 2 to find independent coding variables across images. Instead of performing ICA directly on the 3000 image pixels, ICA was performed on the first 200 PCA coefficients of the face images in order to reduce the dimensionality. The first 200 principal components accounted for over 98% of the variance in the images. These coefficients comprised the columns of the input data matrix, $X = R_{200}^T$.

The ICA algorithm found a 200×200 weight matrix W_I that produced a set of independent coefficients in the output. The basis functions for this representation consisted of the columns of $A = W_I^{-1}$. A sample of the basis set is shown in Figure 13, where the principal component reconstruction $P_{200}A$ was used to visualize the bases as images. The basis images in A have more global properties than the basis images in the ICA output of Architecture 1 (Figure 9). Unlike the ICA output, U, the algorithm does not force the columns of A to be either sparse or independent.

The columns of U contained the representational codes for the training images. The representational code for the test images was found by $W_I X_{test} = U_{test}$, where X_{test} was the zero-mean matrix of the test images. This produced 200 coefficients for each face image, consisting of the outputs of the 200 ICA filters.

Face recognition performance was again evaluated by the nearest neighbor procedure. Figure 14 compares the face recognition performance using the ICA factorial code representation to the independent basis representation of Section 3.1 and to the PCA representation, each with 200 coefficients. Again, there was a trend for the ICA factorial representation (ICA2) to outperform the PCA representation for recognizing faces across changes in session. The difference in performance for Test Set 2 is significant ($Z = 2.7, p < 0.01$). There was no significant difference in the performances of the two ICA representations.

Class discriminability of the 200 ICA factorial coefficients was calculated according to Equation 8. Unlike the coefficients in the independent basis representation, the ICA factorial coefficients did not differ substantially from each other according to discriminability r. Selection of subsets of components for the representation by class discriminability had little effect on the recognition performance using the ICA-factorial representation (see Figure 15). The difference in performance between ICA1 and ICA2 for Test Set 3 following the discriminability analysis just misses significance ($Z = 1.88, p = 0.06$).

Recognition performances were also tested after separating 85 rather than 200 components, and hence estimating fewer weight parameters. The same overall pattern of results was obtained when 85 components were separated. Both ICA representations significantly outperformed the PCA representation on Test Sets 2 and 3. With 85 independent components, ICA1 obtained 87%, 62%, 58% correct performance, respectively on Test Sets 1, 2, and 3, ICA2 obtained 85%, 76%, and 56% correct performance, whereas PCA obtained 85%, 56% and 44% correct, respectively. Again, as found for 200 separated components, selection of subsets of components by class discriminability improved the performance of ICA1 to 86%, 78%, and 65%, respectively, and had little effect on the performances with the PCA and ICA2 representations. This suggests that the results were not simply an artifact due to small sample size.

Figure 13: Basis images for the ICA factorial representation (columns of $A = W_I^{-1}$) obtained with Architecture 2. (See Figure 6).

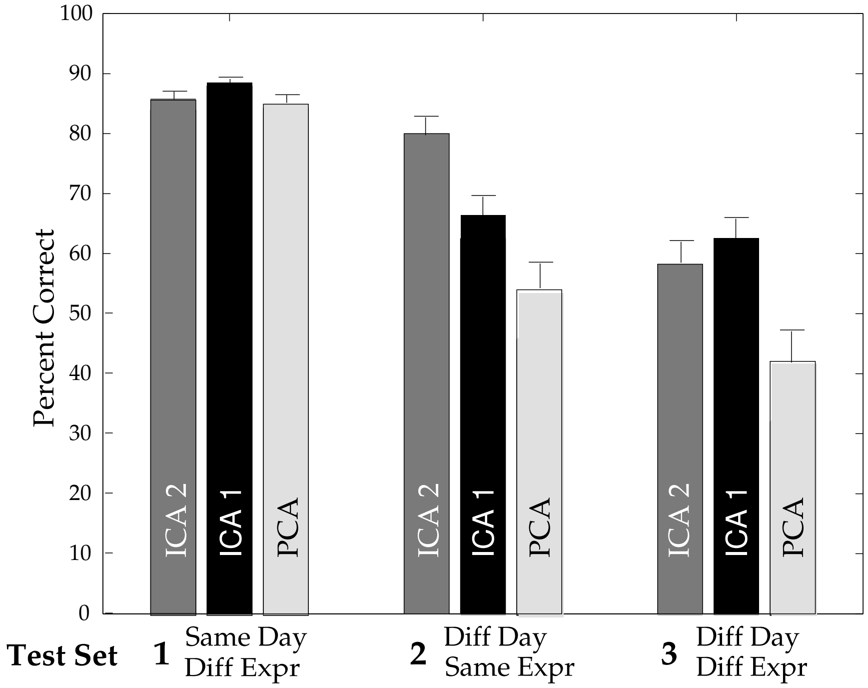

Figure 14: Recognition performance of the factorial code ICA representation (ICA2) using all 200 coefficients, compared to the ICA independent basis representation (ICA1), and the PCA representation, also with 200 coefficients.

5 Examination of the ICA Representations

5.1 Mutual information

A measure of the statistical dependencies of the face representations was obtained by calculating the mean mutual information between pairs of 50 basis images. Mutual information was calculated as

$$I(u_1, u_2) = \frac{H(u_1) + H(u_2) - H(u_1, u_2)}{H(u_1)} \tag{9}$$

$$\text{where } H(u_1) = -E\left[log(P_{u_1})\right].$$

Figure 16 (Top) compares the mutual information between *basis images* for the original graylevel images, the principal component basis images, and the ICA basis images obtained in Architecture 1. Principal component images are uncorrelated, but there are remaining high order dependencies. The information maximization algorithm decreased these residual dependencies by more than 50%. The remaining dependence may be due to a mismatch between the logistic transfer function employed in the learning rule and the cumulative density function of the independent sources, the presence of sub-Gaussian sources, or the large number of free parameters to be estimated relative to the number of training images.

Figure 16 (Bottom) compares the mutual information between the *coding variables* in the ICA factorial representation obtained with Architecture 2, the PCA representation, and graylevel images. For graylevel images, mutual information was calculated between pairs of pixel locations, for the PCA representation, mutual information was calculated between

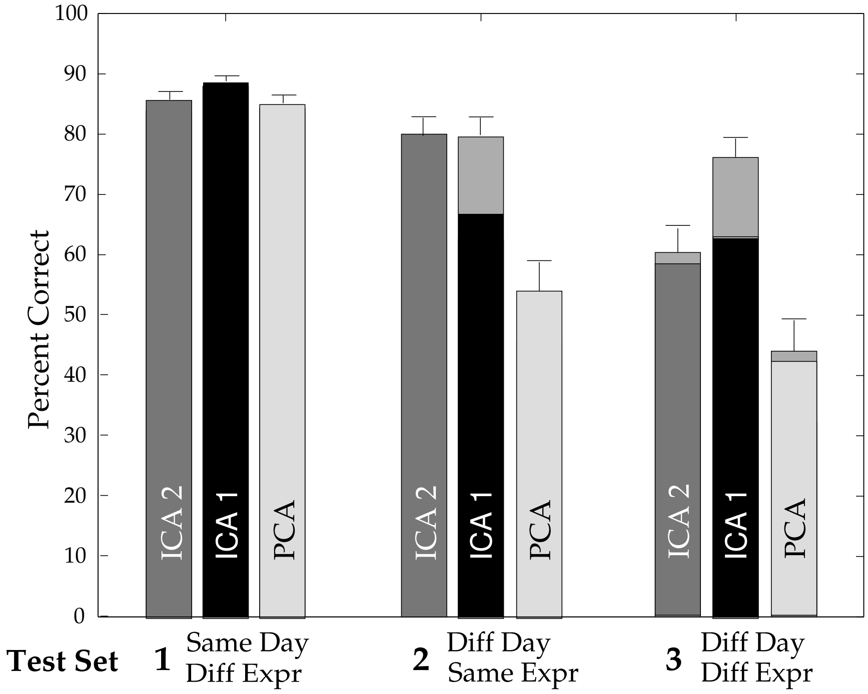

Figure 15: Improvement in recognition performance of the two ICA representations and the PCA representation by selecting subsets of components by class discriminability. Gray extensions show improvement over recognition performance using all 200 coefficients.

pairs of principal component coefficients, and for the ICA factorial representation, mutual information was calculated between pairs of coefficients, b. Again, there were considerable high-order dependencies remaining in the PCA representation that were reduced by more than 50% by the information maximization algorithm. The ICA representations obtained in these simulations are most accurately described not as "independent," but as "redundancy reduced," where the redundancy is less than half that in the principal component representation.

5.2 Sparseness

Field [10] has argued that sparse distributed representations are advantageous for coding visual stimuli. Sparse representations are characterized by highly kurtotic response distributions, in which a large concentration of values are near zero, with rare occurrences of large positive or negative values in the tails. In such a code, the redundancy of the input is transformed into the redundancy of the response patterns of the the individual outputs. This is equivalent to the minimum entropy codes discussed by Barlow [3]. A transformation that minimizes the entropy of the individual outputs encourages statistical independence between the outputs.[2]

Given the relationship between sparse codes and minimum entropy, the advantages for sparse codes as outlined by Field in [10] mirror the arguments for independence presented by Barlow in [3]. Codes that minimize the number of active neurons can be useful in the detection of suspicious coincidences. Because a nonzero response of each unit is relatively rare, high-order relations become increasingly rare, and therefore more informative when

[2]Information maximization is consistent with minimum entropy coding. By maximizing the *joint* entropy of the output, the entropies of the *individual* outputs tend to be minimized.

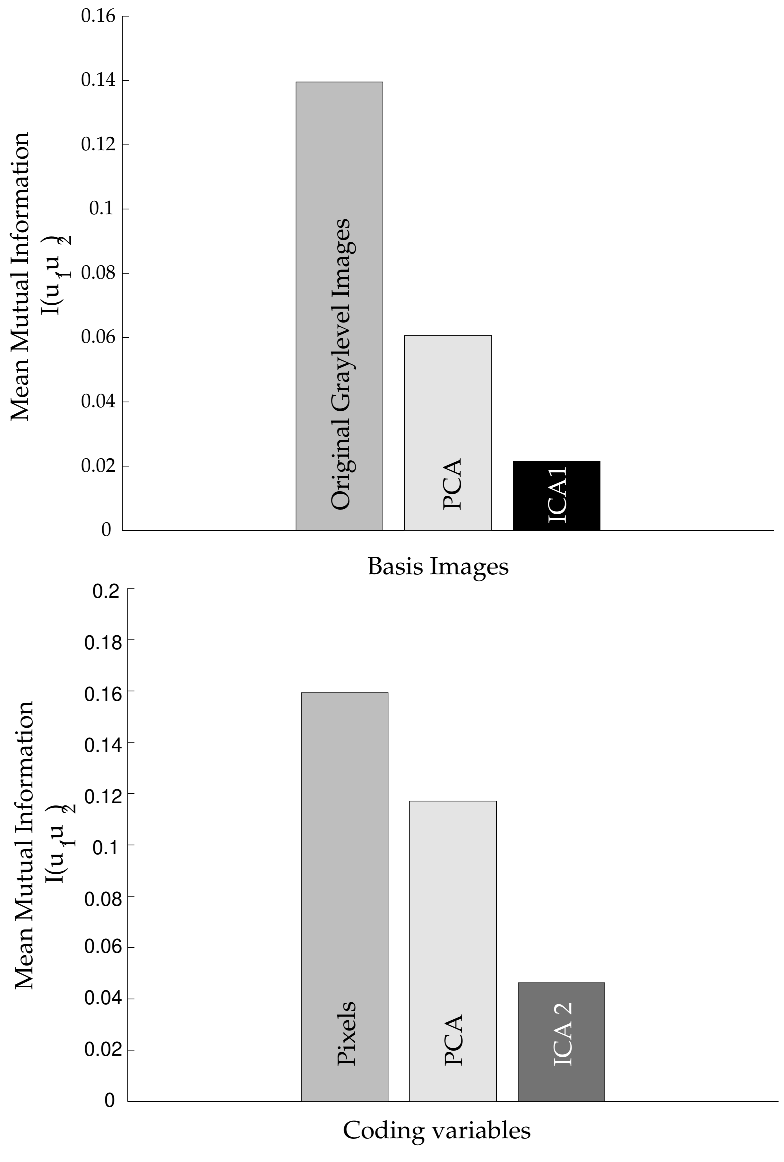

Figure 16: Pairwise mutual information. Top: Mean mutual information between basis images. Mutual information was measured between pairs of graylevel images, principal component images, and independent basis images obtained by Architecture 1. Bottom: Mean mutual Information between coding variables. Mutual information was measured between pairs of image pixels in graylevel images, PCA coefficients, and ICA coefficients obtained by Architecture 2.

they are present in the stimulus. Field contrasts this with a compact code such as principal components, in which a few units have a relatively high probability of response, and therefore high-order combinations among this group are relatively common. In a sparse distributed code, different objects are represented by which units are active, rather than by how much they are active. These representations have an added advantage in signal-to-noise, since one need only determine which units are active without regard to the precise level of activity. An additional advantage of sparse coding for face representations is storage in associative memory systems. Networks with sparse inputs can store more memories and provide more effective retrieval with partial information [20, 5].

The probability densities for the values of the coefficients of the two ICA representations and the PCA representation are shown in Figure 17. The sparseness of the face representations were examined by measuring the kurtosis of the distributions. Kurtosis is defined as the ratio of the fourth moment of the distribution to the square of the second moment, normalized to zero for the Gaussian distribution by subtracting 3:

$$kurtosis = \frac{\sum_i (b_i - \overline{b})^4}{\left(\sum_i (b_i - \overline{b})^2\right)^2} - 3 \tag{10}$$

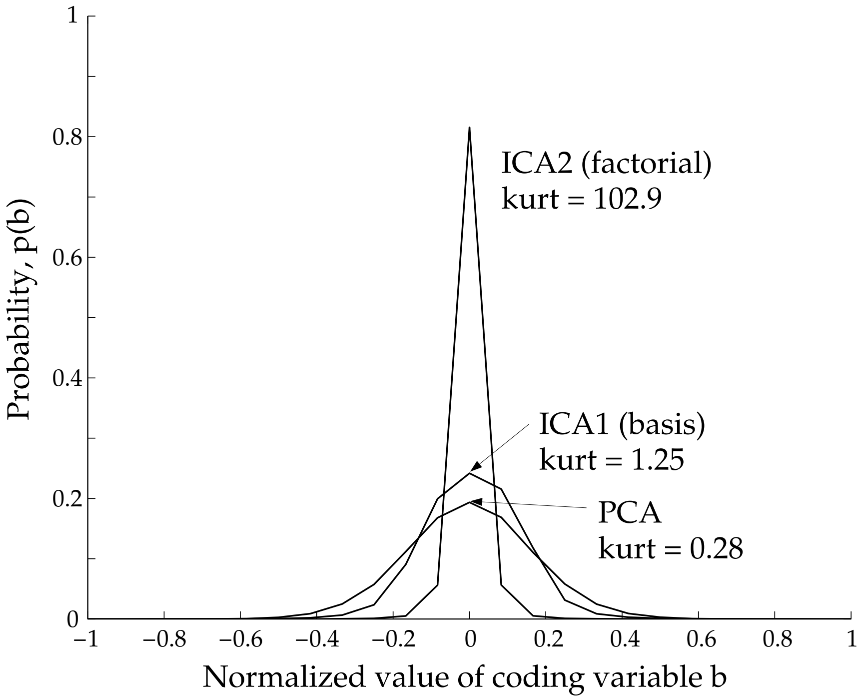

Figure 17: Kurtosis (sparseness) of ICA and PCA representations.

The kurtosis of the PCA representation was measured for the principal component coefficients. The principal components of the face images had a kurtosis of 0.28. The coefficients, b, of the independent basis representation from Architecture 1 had a kurtosis of 1.25. In contrast, the coefficients, b, of the ICA factorial code representation from Architecture 2 was highly kurtotic, at 102.9.

5.3 Combined ICA recognition system

Given that the two ICA representations gave similar recognition performances, we examined whether the two representations gave similar patterns of errors on the face images. There was a significant tendency for the two algorithms to misclassify the same images. The probability that the ICA-factorial representation (ICA2) made an error given that the ICA-basis representation (ICA1) made an error was .72, .88, and .89 respectively for the three test sets. These conditional error rates were significantly higher than the marginal error rates ($Z = 7.4, p < .001; Z = 3.4, p < .001; Z = 2.8, p < .01$), respectively. Examples of successes and failures of the two algorithms are shown in Figure 18.

Figure 18: Recognition successes and failures. Left: Two face image pairs which both ICA algorithms correctly recognized. Right: Two face image pairs that were misidentified by both ICA algorithms.

When the two algorithms made errors, however, they did not confuse the same *pairs* of images. Out of a total of 62 common errors between the two systems, only once did both algorithms assign the same incorrect identity. The two representations were therefore used in conjunction to provide a reliability measure, where classifications were accepted only if both algorithms gave the same answer. This combined ICA recognition system gave an overall classification performance of 99.8% for the 400 images that met this simple criterion, out of the total of 509 test images (100%, 100%, and 97% for the three test sets, respectively).

Because the confusions made by the two algorithms differed, a combined classifier was employed in which the similarity between a test image and a gallery image was defined as $d_1 + d_2$, where d_1 and d_2 correspond to the similarity measure d in Equation 7 for ICA1 and ICA2. Class discriminability analysis was carried out on ICA1 and ICA2 before calculating d_1 and d_2. Performance of the combined classifier is shown in Figure 19. The combined classifier improved performance to 91.0% ,88.9%, and 81.0% for the three test cases, respectively. The difference in performance between the combined ICA classifier and PCA was significant for all three test sets ($Z = 2.7, p < 0.01; Z = 3.7, p < .001; Z = 3.7; p < .001$).

6 Discussion

In a task such as face recognition, much of the important information may be contained in the high-order relationships among the image pixels. Face representations such as "Eigenfaces" and "Holons" are based on principal component analysis, which separates the second-order statistics of the image set, but does not address the high-order relationships in the images. We derived two representations for face recognition based on the statistically independent components of face images. One representation used independent component analysis (ICA)

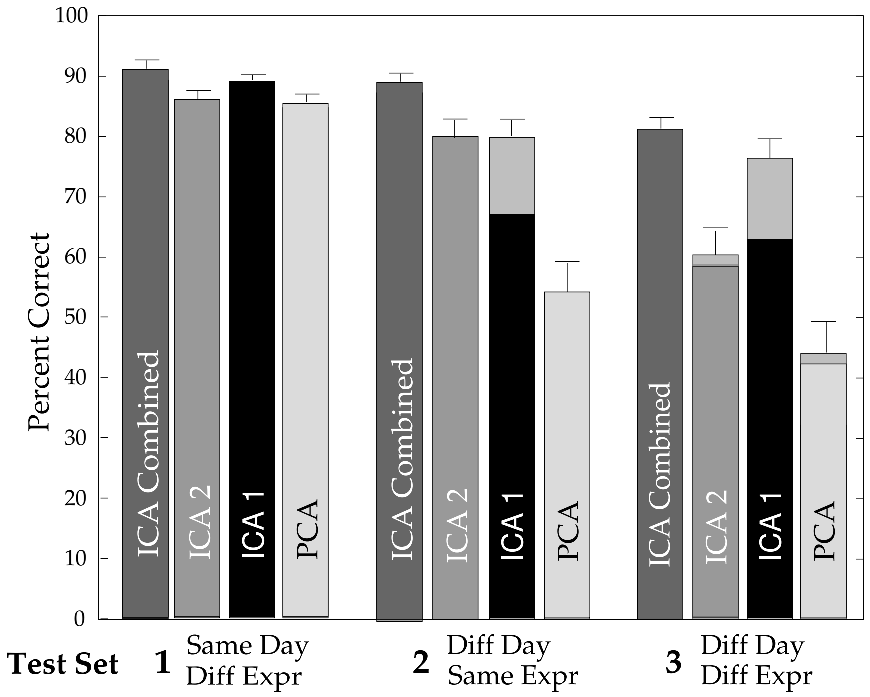

Figure 19: Face recognition performance of the combined ICA classifier, compared to the individual classifiers for ICA1 and ICA2, and PCA.

to find a set of independent basis images that can be considered a set of independent facial feature images. This representation was obtained by employing an architecture that found a set of independent images across spatial location. The representation defined faces as a linear combination of a set of independent feature images. The face code consisted of the coefficients for the linear combination of basis images that comprised each face image. The second representation used ICA to find a factorial face code, in which the coding variables were independent. This representation was obtained by employing an architecture that separated a set of independent coding variables across images. The ICA representations embodied a prior that these image features were independent across individuals, so that when there were statistical dependencies, they more reliably signaled a feature combination that occurred within an individual.

Principal component analysis defines face space in terms of directions of covariance in the data. ICA, on the other hand defines face space in terms of directions of statistical dependence. ICA encodes the statistical dependencies that are expected in the input and removes them from the output. Each output unit learns a set of weights that encodes a portion of the statistical dependencies in the input, so that the dependencies are removed from between the output units.

Both ICA representations outperformed the "Eigenface" representation [26], which was based on principal components, for recognizing images of faces sampled on a different day from the training images. This result is particularly encouraging, since most applications of automated face recognition require identification of images collected on a different day from the sample images. These images can differ in the precise lighting conditions and facial pose, in addition to possible gross differences due to changes in hair, make-up, and facial expression. A classifier that combined the two ICA representations outperformed Eigenfaces across all three test sets. Methods have been presented for optimizing recognition performance with "Eigenfaces," such as building a modular representation consisting of

"Eigenfaces" plus "Eigenfeatures," which are principal components of subimages containing the eyes, nose, or mouth [22]. These optimization procedures are applicable to the ICA representations as well. The purpose of the comparison in this paper was to compare the ICA and PCA-based representations under identical conditions.

The ICA representation of faces presented in Section 3.1 is related to the method of local feature analysis (LFA), which is a topographic representation based on principal components analysis. The LFA kernels are found by performing zero phase whitening (Equation 3) on the principal component axes, followed by a rotation to topographic correspondence with pixel location. Convolving an image with these kernels is equivalent to reconstructing the image with whitened principal component coefficients. In LFA, the kernels are matched to the second-order statistics of the input ensemble, whereas in the ICA representation, the kernels are matched to the high-order statistics of the ensemble as well as the second-order statistics. Both methods produce local filters from an objective of redundancy reduction.

In Section 3.3, independent component analysis provided a set of statistically independent coefficients for coding the images. It has been argued that such a factorial code is advantageous for encoding complex objects that are characterized by high-order combinations of features, since the prior probability of any combination of features can be obtained from their individual probabilities [3, 1]. According to the arguments of both Field [10] and Barlow [3], the ICA-factorial representation is a more optimal object representation than the ICA-basis representation given its sparse, factorial properties. Due to the difference in architecture, the ICA-factorial representation always had fewer training samples to estimate the same number of free parameters as the ICA-basis representation. Figure 16 shows that the residual dependencies in the ICA factorial representation were higher than in the ICA basis representation. The ICA-factorial representation may prove to have a greater advantage given a much larger training set of images. It also is possible that the factorial code representation may prove advantageous with a more powerful recognition engine than nearest neighbors on cosines, such as a Bayesian classifier. An image set containing many more frontal view images of each subject will be needed to test that hypothesis.

The information maximization learning algorithm was developed from the principle of optimal information transfer in sigmoidal neurons. It contains a Hebbian correlational term between the nonlinearly transformed outputs and weighted feedback from the linear outputs and [7]. The biological plausibility of the learning algorithm, however, is limited by fact that the learning rule is nonlocal. Local learning rules for independent component analysis are presently under development [13].

The principle of independence, if not the specific learning algorithm employed here [7], may have relevance to face and object representations in the brain. Horace Barlow [3] and Joseph Atick [1] have argued for redundancy reduction as a general coding strategy in the brain. This notion is supported by the findings of Bell and Sejnowski [7] that the filters that produce independent outputs from natural scenes are local, oriented, spatially opponent filters similar to the response properties of V1 simple cells. Olshausen and Field [18, 17] obtained a similar result with a sparseness objective, where there is a close information theoretic relationship between sparseness and independence [3, 7]. It has also been shown that Gabor filter outputs of natural images are pairwise independent in the presence of divisive normalization such as the contrast gain control mechanisms proposed for V1 simple cells [25, 11]. The finding in this paper that face representations derived from independent component analysis give superior face recognition performance to representations that remove only the second-order redundancies supports arguments that independence is a good strategy for high-level object recognition.

Acknowledgments

We are grateful to Javier Movellan, Martin McKeown, and Michael Gray for helpful discussions on this topic, and valuable comments earlier drafts of this paper. Support for this work was provided by Lawrence Livermore National Laboratories ISCR agreement B291528, the McDonnell-Pew Center for Cognitive Neuroscience at San Diego, and the Howard Hughes Medical Institute.

References

[1] J. Atick. Could information theory provide an ecological theory of sensory processing? *Network*, 3:213–251, 1992.

[2] J. Atick and A. Redlich. What does the retina know about natural scenes? *Neural Computation*, 4:196–210, 1992.

[3] H. Barlow. Unsupervised learning. *Neural Computation*, 1:295–311, 1989.

[4] H. Barlow. What is the computational goal of the neocortex? In C. Koch, editor, *Large scale neuronal theories of the brain*, pages 1–22. MIT Press, Cambridge, MA, 1994.

[5] E. Baum, J. Moody, and F. Wilczek. Internal representaions for associative memory. *Biological Cybernetics*, 59:217–228, 1988.

[6] A. Bell and T. Sejnowski. An information-maximization approach to blind separation and blind deconvolution. *Neural Computation*, 7(6):1129–1159, 1995.

[7] A. Bell and T. Sejnowski. The independent components of natural scenes are edge filters. *Vision Research*, 37(23):3327–3338, 1997.

[8] P. Comon. Independent component analysis - a new concept? *Signal Processing*, 36:287–314, 1994.

[9] G. Cottrell and J. . Metcalfe. Face, gender and emotion recognition using holons. In D. Touretzky, editor, *Advances in Neural Information Processing Systems*, volume 3, pages 564–571, San Mateo, CA, 1991. Morgan Kaufmann.

[10] D. Field. What is the goal of sensory coding? *Neural Computation*, 6:559–601, 1994.

[11] D. Heeger. Normalization of cell responses in cat striate cortex. *Visual Neuroscience*, 9:181–197, 1992.

[12] S. Laughlin. A simple coding procedure enhances a neuron's information capacity. *Z. Naturforsch*, 36:910–912, 1981.

[13] J. Lin, D. Grier, and J. Cowan. Source separation and density estimation by faithful equivariant som. In M. Mozer, M. Jordan, and T. Petsche, editors, *Advances in Neural Information Processing Systems*, volume 9, pages 536–541, Cambridge, MA, 1997. MIT Press.

[14] S. Makeig, A. Bell, T.-P. Jung, and T. Sejnowski. Independent component analysis of electroencephalographic data. In D. Touretzky, M. Mozer, and M. Hasselmo, editors, *Advances in Neural Information Processing Systems*, volume 8, pages 145–151, Cambridge, MA, 1996. MIT Press.

[15] M. McKeown, S. Makeig, G. Brown, T.-P. Jung, S. Kindermann, A. Bell, and T. Sejnowski. Analysis of fmri data by decomposition into independent components. *Proc. Nat. Acad. Sci.*, in press.

[16] J.-P. Nadal and N. Parga. Non-linear neurons in the low noise limit: a factorial code maximizes information trans fer. *Network*, 5:565–581, 1994.

[17] B. Olshausen and D. Field. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381:607–609, 1996.

[18] B. Olshausen and D. Field. Natural image statistics and efficient coding. *Network: Computation in Neural Systems*, 7(2):333–340, 1996.

[19] A. Oppenheim and J. Lim. The importance of phase in signals. *Proceedings of the IEEE*, 69:529–541, 1981.

[20] G. Palm. On associative memory. *Biological Cybernetics*, 36:19–31, 1980.

[21] P. Penev and J. Atick. Local feature analysis: a general statistical theory for object representation. *Network: Computation in Neural Systems*, 7(3):477–500, 1996.

[22] A. Pentland, B. Moghaddam, and T. Starner. View-based and modular eigenspaces for face recognition. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 1994.

[23] P. Phillips, H. Wechsler, J. Juang, and P. Rauss. The feret database and evaluation procedure for face-recognition algorithms. *Image and Vision Computing Journal*, in press.

[24] L. Piotrowski and F. Campbell. A demonstration of the visual importance and flexibility of spatial-frequency, amplitude, and phase. *Perception*, 11:337–346, 1982.

[25] E. P. Simoncelli. Statistical models for images: Compression, restoration and synthesis. In *31st Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, CA, November 2-5 1997.

[26] M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.