AN INTERPOLATION METHOD FOR THE RECONSTRUCTION AND RECOGNITION OF FACE IMAGES

N. C. Nguyen and J. Peraire
Massachusetts Institute of Technology, 77 Massachusetts Ave, Cambridge, MA 20139, USA
cuongng@mit.edu and peraire@mit.edu

Keywords: Face reconstruction, face recognition, best points interpolation method, principal component analysis.

Abstract: An interpolation method is presented for the reconstruction and recognition of human face images. Basic ingredients include an optimal basis set defining a low-dimensional face space and a set of “best interpolation pixels” capturing the most relevant characteristics of known faces. The best interpolation pixels are chosen as points of the pixel grid so as to best interpolate the set of known face images. These pixels are then used in a least-squares interpolation procedure to determine interpolant components of a face image very inexpensively, thereby providing efficient reconstruction of faces. In addition, the method allows a fully automatic computer system to be developed for the real-time recognition of faces. Two significant advantages of this method are: (1) the computational cost of recognizing a new face is independent of the size of the pixel grid; and (2) it allows for the reconstruction and recognition of incomplete images.

1 INTRODUCTION

Image processing and recognition of human faces constitutes a very active area of research. The field has evolved rapidly and become one of the most successful applications of image analysis and computer vision partly because of availability of many powerful methods and partly because of its significant practical importance in many areas such as authenticity in security and defense systems, banking, human–machine interaction, image and multimedia processing, psychology, and neurology. Principal component analysis (PCA) or the Karhunen-Loève (KL) expansion is a well-established method for the representation (Sirovich and Kirby, 1987; Kirby and Sirovich, 1990) and recognition (Turk and Pentland, 1991) of human faces.

PCA approach (Kirby and Sirovich, 1990) for face representation consists of computing the “eigenfaces” of a set of known face images and approximating any particular face by a linear combination of the leading eigenfaces. For face recognition (Turk and Pentland, 1991), a new face is first projected onto the eigenface space and then classified according to the distances between its PCA coefficient vector and those representing the known faces. There are two drawbacks with this approach. First, PCA may not handle corrupted data well, that is, situations in which only partial information of an input image is available. Secondly, the computational cost per image classification depends on the size of the pixel grid. Despite this, PCA is still one of the most used techniques for face recognition due to its simplicity and efficiency over other methods.

This paper describes an interpolation method that aims to address these deficiencies of PCA. The method was first introduced in (Nguyen et al., 2006) for the approximation of parametrized fields. Here, we investigate the method for face reconstruction and recognition. The basic ingredient is a set of “best interpolation pixels” capturing the most relevant features of known face images. The essential component is a least-squares interpolation procedure for the very rapid computation of the interpolant coefficient vector of any given input face. The interpolant coefficient vector is then used to determine which face in the face set, if any, best matches the input face. A significant advantage of our approach is that the computational cost of recognizing a new face is independent of the size of the pixel grid, while achieving a recog-
An Interpolation Method for the Reconstruction and Recognition of Face Images

An interpolation method is presented for the reconstruction and recognition of human face images. Basic ingredients include an optimal basis set defining a low-dimensional face space and a set of best interpolation pixels capturing the most relevant characteristics of known faces. The best interpolation pixels are chosen as points of the pixel grid so as to best interpolate the set of known face images. These pixels are then used in a least-squares interpolation procedure to determine interpolant components of a face image very inexpensively, thereby providing efficient reconstruction of faces. In addition, the method allows a fully automatic computer system to be developed for the real-time recognition of faces. Two significant advantages of this method are: (1) the computational cost of recognizing a new face is independent of the size of the pixel grid; and (2) it allows for the reconstruction and recognition of incomplete images.
nition rate comparable to PCA approach. Moreover, the method allows the reconstruction and recognition of corrupted images.

In the past years, there have been a large number of contributions to face recognition and analysis. The reader is referred to a number of papers (Chellappa et al., 1995; Jain and Li, 2005; Delac et al., 2005; Draper et al., 2003; Phillips et al., 1998) for perspectives and recent advances in face recognition. Face analysis and representation have also been extensively studied by many authors (Sirovich and Kirby, 1987; Kirby and Sirovich, 1990; O’Toole et al., 1993; Ever-son and Sirovich, 1995; Kanade, 2005).

This paper is organized as follows. In Section 2, we present an overview of PCA. In Section 3, we extend the best points interpolation method (BPIM) introduced in (Nguyen et al., 2006) and apply it to de-
velop an automatic real-time face recognition system. In Section 4, we test and compare our approach with PCA. Finally, in Section 5, we close the paper with some concluding remarks.

2 PRINCIPAL COMPONENT ANALYSIS

2.1 Eigenfaces

An ensemble of face images is denoted by \( \mathcal{U}_K = \{ u_i \} \), \( 1 \leq i \leq K \), where \( u_i \) represents an \( i \)-th mean-subtracted face and \( K \) represents the number of faces in the ensemble. It is assumed that after proper normalization and resizing to a fixed pixel grid \( \Xi \) of dimension \( N_1 \times N_2 \), \( u_i \) can be considered as a vector in an \( N \)-dimensional image space, where \( N = N_1 N_2 \) is the number of pixels. PCA (Sirovich and Kirby, 1987; Kirby and Sirovich, 1990) derives an optimal representation of the face ensemble in the sense that the average reconstruction error

\[
\mathcal{E}^* = \frac{1}{K} \sum_{i=1}^{K} \left( u_i - \sum_{j=1}^{K} (\Phi_j^T u_i) \Phi_j \right)^2 ,
\]

is minimal for all \( k \leq K \). In the literature (Turk and Pentland, 1991), the basis vectors \( \Phi_j \) are referred as eigenfaces and the space spanned by them is known as the face space. The construction of the eigenfaces is described as follows.

Let \( U \) be the \( N \times K \) matrix whose columns are \([u_1, \ldots, u_K]\). It can be shown that the \( \Phi_i \) satisfy

\[
A \Phi_i = \lambda_i \Phi_i ,
\]

where the covariance matrix \( A \) is given by

\[
A = \frac{1}{K} UU^T .
\]

We present in Figure 1 the mean face and a few of the top eigenfaces for a training ensemble of 400 face images extracted from the AT&T database (see Section 4.1 for details). Figure 2 shows \( \mathcal{E}_{rel}^* \) as a function of \( k \). Here \( \mathcal{E}_{rel}^* \) is the average of the relative error

Here the eigenvalues are arranged such that \( \lambda_1 \geq \ldots \geq \lambda_K \). Since the matrix \( A \) of size \( N \times N \) is large, solving the above eigenvalue problem can be very expensive.

However, if \( K < N \), there will be only \( K \) meaningful eigenvectors and we may express any \( \Phi_i \) as

\[
\Phi_i = \sum_{j=1}^{K} \varphi_{ij} u_j ,
\]

Inserting (3) and (4) into (2), we immediately obtain

\[
G \varphi_i = \lambda_i \varphi_i ,
\]

where \( G = \frac{1}{K} U^T U \) is a symmetric positive-definite matrix of size \( K \) by \( K \). The eigenvalue problem (5) can be solved for \( \varphi_{ij}, 1 \leq i, j \leq K \), from which the eigenfaces \( \Phi_i \) are obtained.

2.2 Face Reconstruction

We briefly describe the reconstruction of face images using PCA and later compare the results with those obtained using our method. To this end, we seek to project an input face \( u \) onto the face space \( \Phi_k = \text{span}\{\phi_1, \ldots, \phi_k\} \) to obtain

\[
u^* = \sum_{i=1}^{k} a_i \phi_i ,
\]

where for \( i = 1, \ldots, k \),

\[
a_i = \phi_i^T u .
\]

We also define the associated error as

\[
e^* = ||u - u^*|| .
\]

Note that the mean face of the ensemble \( \mathcal{U}_K \) should be added to \( u^* \) to obtain the reconstructed image; and that if \( k \) is set equal to \( K \), the reconstruction is exact for all members of the ensemble.

We present in Figure 1 the mean face and a few of the top eigenfaces for a training ensemble of 400 face images extracted from the AT&T database (see Section 4.1 for details). Figure 2 shows \( \mathcal{E}_{rel}^* \) as a function of \( k \). Here \( \mathcal{E}_{rel}^* \) is the average of the relative error.
dependent component analysis (ICA) (Draper et al., 2003; Bartlett et al., 2002) and linear discriminant analysis (LDA) (Etemad and Chellappa, 1997; Lu et al., 2003) suffer from similar drawbacks.

3 BEST POINTS
INTERPOLATION METHOD

In this section, we extend the best points interpolation method (Nguyen et al., 2006) to deal with face images and apply it for face recognition. We shall use the eigenfaces as basis functions in the interpolation process, as they possess optimal $L_2$ representation of face images. The key idea, however, is to find a set of interpolation points which provides a good uniform approximation.

3.1 Interpolation Procedure

Let us recall the pixel grid $\Xi$ and the face space $\Phi_k = \text{span}\{\phi_1, \ldots, \phi_k\}$. In this space, we shall seek an approximation of any input image $u$. However, rather than performing the projection of $u$ onto $\Phi_k$ for the best approximation $u^*$, we choose to interpolate $u$ as follows.

In particular, we aim to find a good approximation $\tilde{u} \in \Phi_k$ of $u$ via $m \geq k$ interpolation pixels $\{z_j \in \Xi\}, 1 \leq j \leq m$, such that
\[
\tilde{u} = \sum_{i=1}^{k} \tilde{a}_i \phi_i
\]
where the coefficients $\tilde{a}_i$ are the solution of
\[
\sum_{i=1}^{k} \phi_i(z_j) \tilde{a}_i = u(z_j), \quad j = 1, \ldots, m.
\]
We define the associated error as
\[
\tilde{e} = \|u - \tilde{u}\|.
\]
In general, the linear system (10) is over-determined because there are more equations than unknowns. However, the interpolant coefficient vector $\tilde{a} = [\tilde{a}_1, \ldots, \tilde{a}_k]^T$ can be determined from
\[
C^T \tilde{a} = C^T c,
\]
where $C \in \mathbb{R}^{m \times k}$ with $C_{ji} = \phi_i(z_j), 1 \leq i \leq k, 1 \leq j \leq m$ and $c = [u(z_1), \ldots, u(z_m)]^T$. It thus follows that
\[
\tilde{a} = Bc.
\]
Here the matrix $B = (C^T C)^{-1} C^T$ is precomputed and stored. Therefore, for any new face $u$, the cost of evaluating the interpolant coefficient vector $\tilde{a}$ is only $O(mk)$ and becomes $O(k^2)$ when $m = O(k)$. 

Figure 2: Average relative error $\epsilon_{rel}$ versus $k$ for the training ensemble.
We proceed to describe our approach for determining the interpolation pixels. The crucial observation is that much of the surface of a face is smooth with regular texture and that faces are similar in appearance and highly constrained; for example, the frontal view of a face is symmetric. Moreover, the value of a pixel is typically highly correlated with the values of the surrounding pixels. Therefore, a large number of pixels in the image space does not represent physically possible faces. Only a small number of pixels may suffice to represent facial characteristics. The question we aim to address is to find such pixels and proceed with our interpolation.

We choose the interpolation pixels by exploiting the training ensemble $\mathcal{U}_k$. Specifically, we might consider to choose $\{z_j\}$ by formulating a minimization problem that minimizes the sum of squared errors between $u_\ell$, $1 \leq \ell \leq K$, and their approximations. More precisely, we might wish to find $\{z_j\}$ as a minimizer of

$$\min_{x_i \in \Xi,...,x_m \in \Xi} \sum_{\ell=1}^{K} \left\| u_\ell - \sum_{i=1}^{k} a_{i\ell}(x_1,\ldots,x_m) \phi_i \right\|^2 \tag{14}$$

$$\sum_{i=1}^{k} \phi_i(x_j)a_{i\ell} = u_\ell(x_j), \quad 1 \leq j \leq m, 1 \leq \ell \leq K.$$  

Clearly, the minimizer of the above error minimization problem is optimal for the interpolation of the face images belonging to $\mathcal{U}_k$. However, since the problem is nonconvex with multiple local minima and the Hessian is not easily computed, solving it is particularly difficult. In practice, we find $\{z_j\}$ by solving a simpler minimization problem introduced below.

To begin, we introduce a set of images, $\mathcal{U}_k^* = \{u_\ell^*\}, 1 \leq \ell \leq K^*$, where $u_\ell^*$ is the best approximation to $u_\ell$. It thus follows that

$$u_\ell^* = \sum_{i=1}^{k} a_{i\ell}^* \phi_i \tag{15}$$

where for $1 \leq i \leq k$, $1 \leq \ell \leq K^*$,

$$a_{i\ell}^* = \hat{a}_{i\ell}^* u_\ell^* \tag{16}$$

By replacing $u_\ell$ in the objective of the problem (14) with $u_\ell^*$ and expanding the resulting objective, we arrive at the nonlinear least squares minimization

$$\min_{x_i \in \Xi,...,x_m \in \Xi} \sum_{\ell=1}^{K_k} \sum_{i=1}^{k} \left( a_{i\ell} - \hat{a}_{i\ell}(x_1,\ldots,x_m) \right)^2 \tag{17}$$

$$\sum_{i=1}^{k} \phi_i(x_j)a_{i\ell} = u_\ell(x_j), \quad 1 \leq j \leq m, 1 \leq \ell \leq K.$$

Let us denote a minimizer of this problem by $\{z_j\}, 1 \leq j \leq m$. We shall call the $z_j$ as best interpolation pixels, because these pixels are optimal for the interpolation of the best approximations $u_\ell^*$. It remains to describe the solution procedure for (17).
3.3 Solution Procedure

We first write the linear system in (17) for \( \tilde{a}_i = [a_{i1}, \ldots, a_{ik}]^T \) into the matrix form as

\[
\begin{align*}
D^T \tilde{D} \tilde{a}_\ell &= D^T d_\ell, & 1 \leq \ell \leq K,
\end{align*}
\]

where \( d_\ell = [u_1(x_1), \ldots, u_L(x_m)]^T \) and \( D \in \mathbb{R}^{m \times k} \) with \( D_{ji} = \phi_i(x_j) \). Next let \( s = [x_1, \ldots, x_m]^T \), for \( 1 \leq i \leq k, 1 \leq \ell \leq K, 1 \leq q \leq Q = kk \), we set

\[
\begin{align*}
&f_q(s) = a_{\ell i} - \tilde{a}_{\ell i}(s); \\
&F(s) = \frac{1}{2} \sum_{q=1}^Q f_q^2(s).
\end{align*}
\]

The gradient and Hessian of the objective function \( F(s) \) can thus be computed as

\[
\begin{align*}
\nabla F(s) &= \sum_{q=1}^Q f_q(s) \nabla f_q(s) = J(s)^T f(s), \\
\nabla^2 F(s) &= J(s)^T J(s) + \sum_{q=1}^Q f_q(s) \nabla^2 f_q(s),
\end{align*}
\]

where \( J(s) \in \mathbb{R}^{Q \times 2m} \), for \( 1 \leq q \leq Q, 1 \leq p \leq 2m, \)

\[
J_{qp}(s) = \frac{\partial f_q(s)}{\partial x_p^q}, & 1 \leq j \leq m, d = 1, 2.
\]

Hence, when the residuals \( f_q(s) \) are small, we may approximately compute the Hessian in terms of only the Jacobian matrix \( J(s) \) as

\[
\nabla^2 F(s) = J(s)^T J(s).
\]

To compute the Jacobian \( J(s) \), we differentiate both sides of (18) with respect to \( x_j^d \) to obtain

\[
\frac{\partial \tilde{a}_\ell}{\partial x_j^d} = E^{-1} \left( \frac{\partial D^T}{\partial x_j^d} d_\ell + D^T \frac{\partial D}{\partial x_j^d} \tilde{a}_\ell \right),
\]

where \( E = D^T D \). The partial derivatives \( \partial d_\ell / \partial x_m^q \), \( \partial D^T / \partial x_m^q \), and \( \partial E / \partial x_m^q \) are computed by finite differences. Note also that \( s = (x_1^1, x_2^2) \).

Having determined the gradient and the Hessian, we may now use the Levenberg-Marquardt (LM) algorithm (Marquardt, 1963) to solve (17). The LM algorithm is very efficient, but it is sensitive to an initial guess. Hence it is important to start the algorithm with a good initial guess. In our implementation, we use the empirical interpolation method (Barrault et al., 2004; Grepl et al., 2006) to obtain an initial set of interpolation points \( \{ z_j^{ig} \} \) as follows. We first set

\[
zeq = \arg \sup_{x \in \Xi} |\phi_1(x)|.
\]

Then for \( \ell = 2, \ldots, m \), we solve the linear system

\[
\sum_{j=1}^{\ell-1} \phi_j(z_j^{ig}) \sigma_j = \phi_\ell(z_\ell^{ig}), & 1 \leq i \leq \ell - 1,
\]

for \( \sigma_j, 1 \leq j \leq \ell - 1 \), and set

\[
z_\ell^{ig} = \arg \sup_{x \in \Xi} \left| \phi_1(x) - \sum_{j=1}^{\ell-1} \sigma_j \phi_j(x) \right|.
\]

This set of points, when used as an initial guess, yields very satisfactory results. For further details of the empirical interpolation method, we refer the reader to (Barrault et al., 2004; Grepl et al., 2006).

3.4 Application to Face Recognition

We now apply the BPIM to develop a fully automatic real-time face recognition system involving the generation stage and the recognition stage. The detailed implementation of the system is given below:

1. Determine the dimension of the face space \( k \) and choose some \( m \) (say \( m = 2k \)), then calculate \( \phi_1, \ldots, \phi_m \). Note \( m \) eigenfaces are required to obtain the initial guess \( z_1^{ig} \), \( 1 \leq j \leq m \).

2. Compute and store \( \{ z_j \} \), \( B = (C^T C)^{-1} C^T \). Recall that \( C_{ji} = \phi_i(z_j) \), \( 1 \leq i \leq k, 1 \leq j \leq m \).

3. For a “gallery” of images \( \Psi_{k'} = \{ v_i \}, 1 \leq i \leq K' \), compute \( \tilde{a}_i = B [v_i(z_1), \ldots, v_i(z_m)]^T \), \( 1 \leq i \leq K' \). (Note \( \Psi_{k'} \) can be the same or different from \( \Xi_k \)).

4. For each new face to be classified \( u \), calculate its interpolant coefficient vector \( \tilde{a} \) from (13) and find

\[
i_{\min} = \arg \min_{1 \leq i \leq K'} ||\tilde{a} - \tilde{a}_i||.
\]

5. If \( ||\tilde{a} - \tilde{a}_{\text{in}}|| \) is less than a chosen threshold, the input image \( u \) is identified as the individual associated with the coefficient vector \( \tilde{a}_{\text{in}} \). Otherwise, the image is classified as a new individual.

The generation stage (steps 1–2) is computationally expensive, but performed only when the training set changes. Furthermore, even if it is necessary to perform the generation stage due to an update of the training set, we may compute only the eigenfaces and reuse the best pixels. This can save us some computational time.

However, the recognition stage (steps 4–5) is very inexpensive. The calculation of \( \tilde{a} \) takes \( O(mk) \). Note further that the problem (28) is the nearest-neighbor search which can be solved in \( O(kK^{0.25}) \) time with the storage of \( O(kK' + K' \log K') \) (Andoni and Indyk, 2006). Hence, if \( K' \) is in order of \( O(k^2) \) or less, the computational cost is only \( O(k^2) \). This is usually the
case even for large-scale applications; for example, for a training database of $10^4$ images, one would need more (or many more) than 10 eigenfaces to achieve acceptable recognition rates.

In summary, the operation count of the recognition stage is about $O(mk)$. The computational complexity of our system is thus independent of $N$. As mentioned earlier, the complexity of PCA-based algorithms is at least $O(Nk)$. Our approach leads to a computational reduction of $N/m$ relative to PCA. Since $m$ is typically much smaller than $N$, significant savings are expected. The savings per image classification certainly translate to real-time performance especially when many face images need to be classified simultaneously.

However, some applications of face recognition may regard the recognition quality more importantly than the computational performance. Therefore, in order to be useful and gain acceptance, our approach must be tested and compared with existing approaches, particularly here with the PCA.

4 EXPERIMENTS

4.1 Face Database

The AT&T face database (Samaria and Harter, 1994) consists of 400 images of 40 individuals (10 images per individual). The images were taken at different times with variation in lighting, poses, and facial expressions, with and without glasses. The images were cropped and resized by us to a resolution of $74 \times 90$. We formed a training ensemble of 400 images by using 200 images of the database, 10 each of 20 different individuals, and including 200 mirror images of these images (Kirby and Sirovich, 1990).

The testing set contains the (200) remaining images of 20 individuals not belonging to the training ensemble. We further divide the testing set into the gallery of 20 individual faces and 180 probe images containing 9 views of every individual in the gallery. The recognition task is to match the probe images to the 20 gallery faces. The fact that the training and testing sets have no common individual serves to assess the performance of a face recognition system more critically — the ability to recognize new faces which are not part of the face space constructed from the training set.

4.2 Results for Face Reconstruction

We first present in Figure 4 the reconstruction results for a face in the training ensemble. The BPIM produces reconstructions almost as well as PCA: most facial features captured by the PCA reconstructed images also appear in the BPIM reconstructed images. We underline the fact that the interpolation method requires less than 5% of the total number of pixels $N=6660$, but delivers quite satisfactory results.

To illustrate the use of the interpolation approach for reconstructing a full image from a partial image, we consider a face (in the training set) shown at the bottom right and a mask shown at the top right in Figure 5. This is a relatively extreme mask that obscures 90% of the pixels in a random manner. Because the masked face may not have intensity values at all the best interpolation pixels, we need to define a new set of interpolation pixels. To this end, we keep the best interpolation pixels which coincide with some of the white pixels of the masked face and replace the remaining best pixels with the “nearest” white pixels. In Figure 5, the reconstructed images using those interpolation pixels are compared with the PCA reconstructed images utilizing all the pixels. Although the
interpolation procedure does not recover the original face exactly, the construction is visually close to the “best” reconstruction.

4.3 Results for Face Recognition

We apply the face recognition system developed in Section 3.4 to classify the probe images. We illustrate in Figure 6 the recognition accuracy as a function of \( k \) for the BPIM and PCA. As it may be expected, the BPIM yields smaller recognition rates than PCA. However, as \( k \) increases, the BPIM gives recognition rates which are quite comparable to those of PCA for large enough \( k \): PCA achieves a recognition rate of 74.98%, while PBIM results in a recognition rate of 73.66% for \( k = 80 \). In many applications, the small accuracy loss of only 1.32% is paid off very well by the significant reduction of \( 6660/160(>40) \) in complexity. This is confirmed in Table 2 which shows the computational times for the BPIM and PCA. The values are normalized with respect to the time to recognize a face for \( k = 10 \) and \( m = 20 \) with the BPIM. Clearly, the BPIM is significantly faster than PCA. This important advantage is very useful to applications that require a real-time recognition capability.

Finally, in order to appreciate the power of the interpolation approach for classifying corrupted images, we consider a random chosen mask of 10% pixels shown in Figure 7. Next to the mask, we show a few faces which are correctly recognized with using the interpolation procedure when their intensity values are available only at the white pixels of the mask. Note here that the interpolation pixels are chosen in the same way as before.

5 CONCLUSION

We have presented a best points interpolation method for the reconstruction and recognition of face images. The method gives very good reconstruction of face images. Therefore, the method has significant potential to be used as an alternative to PCA for data reconstruction. It is important to note that PCA uses full knowledge of the data in the reconstruction process. In contrast, the BPIM uses only partial knowledge of the data. Therefore, the BPIM is very useful to the restoration of a full image from a partial image.

We have also developed a fully automatic real-time face recognition system based on the BPIM. The system is shown to be able to recognize corrupted images. Moreover, the computational cost of recognizing a new face is only \( O(mk) \), translating to a saving of \( N/m \) relative to PCA approach. Typically, since \( N \) is \( O(10^4) \) and \( m \) is \( O(10^2) \), this implies two orders of magnitude less expensive computationally than PCA. As confirmed in Figure 6 and Table 2, the system is significantly faster, while yielding a comparable recognition rate for large enough \( k \), than a standard PCA-based system. The significant reduction in time should enable us to tackle very large problems. Hence, it is imperative to test our system on a larger database such as the FERET database. We plan to pursue this direction in future research.

The present work may also present an opportunity combining the BPIM with some subspace techniques.
For example, in the context of ICA, instead of using intensity values at all pixels one may use intensity values at only the best interpolation pixels to build an ICA-based recognition system, thereby effecting significant computational savings. It should be mentioned that most presented ICA-based algorithms do not perform ICA directly on the training ensemble of face images, but on either eigenfaces or PCA coefficient vectors, to reduce the heavy computational cost. Although we have not put effort to investigate this direction, we believe that using partial image pixels will not reduce the recognition capability of ICA-based algorithms provided that a sufficient (small) number of the interpolation pixels is used. Furthermore, in ICA Architecture II (Draper et al., 2003; Bartlett et al., 2002), one may want to input the interpolant coefficient vectors (instead of PCA coefficient vectors) to ICA. Similarly, one may choose to perform LDA on the interpolant coefficient vectors to reduce the computational burden considerably.

Although we do not claim that our findings necessarily have a wide range of applications, we believe that our work could open a new direction of research in face recognition and image analysis in general.

ACKNOWLEDGEMENTS

We would like to thank Professor A. T. Patera of MIT for his long-standing collaboration and many invaluable contributions to this work. The authors also thank AT&T Laboratories Cambridge for providing the ORL face database. This work was supported by DARPA and AFOSR under Grant FA9550-03-1-0439 and F49620-03-1-0439, and by the Singapore-MIT Alliance.

REFERENCES


