Appears in the *Fifth European Conference on Computer Vision*, **ECCV'98**, University of Freiburg, Germany, June 2-6, 1998.

Face Recognition Using Evolutionary Pursuit

Chengjun Liu and Harry Wechsler

Department of Computer Science, George Mason University,
4400 University Drive, Fairfax, VA 22030-4444, USA
{cliu, wechsler}@cs.gmu.edu

Abstract. This paper describes a novel and adaptive dictionary method for face recognition using genetic algorithms (GAs) in determining the optimal basis for encoding human faces. In analogy to pursuit methods, our novel method is called Evolutionary Pursuit (EP), and it allows for different types of (non-orthogonal) bases. EP processes face images in a lower dimensional whitened PCA subspace. Directed but random rotations of the basis vectors in this subspace are searched by GAs where evolution is driven by a fitness function defined in terms of performance accuracy and class separation (scatter index). Accuracy indicates the extent to which learning has been successful so far, while the scatter index gives an indication of the expected fitness on future trials. As a result, our approach improves the face recognition performance compared to PCA, and shows better generalization abilities than the Fisher Linear Discriminant (FLD) based methods.

1 Introduction

A successful face recognition methodology depends heavily on the particular choice of the features used by the (pattern) classifier [6], [31], [4]. The search for the best feature set corresponds to finding an optimal neural code, biologically characterized as a lattice of receptive fields (RFs) ('kernels') and computationally developed as an optimal basis [26], [2], [30]. Optimization of the visual system then requires searching for such an optimal basis according to the design criteria such as (A) redundancy minimization and decorrelation, (B) minimization of the reconstruction error, (C) maximization of information transmission (infomax) [24], and (D) sparseness of the neural code [26]. Furthermore, to the design criteria listed above one should add as an important functionality the one related to successful pattern classification, referred to by Edelman [13] as neural Darwinism. The rationale behind feature extraction using an optimal basis representation is that most practical methods for both regression and classification use parameterization in the form of a linear combination of basis functions. This leads to a taxonomy based on the type of the basis functions used by a particular method and the corresponding optimization procedure used for parameter estimation. According to this taxonomy, most practical methods use basis function representation — those are called dictionary or kernel methods, where the particular type of chosen basis functions constitutes a kernel. Further distinction is

made between non-adaptive methods using fixed (predetermined) basis functions and adaptive dictionary methods where basis functions depend (nonlinearly) on some (tunable) parameters, such that the basis functions themselves (or their parameters) are fit to available data [9].

Representative classes of adaptive dictionary methods include two approaches sharing similar dictionary representations : Projection Pursuit (statistical method) and Multilayer Perceptron (neural network method) [20]. Since most practical methods use nonlinear models, the determination of optimal kernels becomes a nonlinear optimization problem. When the objective function lacks an analytical form suitable for gradient descent or the computation involved is prohibitively expensive one should use (directed) random search techniques for nonlinear optimization and variable selection as those methods characteristic of evolutionary computation and genetic algorithms [17].

Most neural network methods use the same type of basis function, defined as hidden units of a feed forward net and having the same form of activation function (sigmoid or radial basis). In contrast, many statistical adaptive methods do not require the form of all basis functions to be the same. In terms of optimization, statistical methods estimate the basis functions one at a time, hence there is no need for all basis functions to be the same. On the other hand, neural network methods based on gradient descent optimization are more suitable for handling representations with identical basis functions which are updated simultaneously.

Projection Pursuit (PP) regression is an example of an additive model with univariate basis functions [15] [19]. A greedy optimization approach, called backfitting, is often used to estimate additive approximating functions. The backfitting algorithm provides a local minimum of the empirical risk encountered during functional approximation by sequentially estimating the individual basis functions of the additive approximating function. Similar to PP in spirit and characteristic of the non-orthogonal and over complete methods is the Matching Pursuit (MP) algorithm [25]. MP decomposes any signal into a linear expansion of waveforms that are selected from a redundant dictionary of functions. These waveforms are chosen in order to best match the signal structure. Using a dictionary of Gabor functions a matching pursuit defines an adaptive time-frequency transform. More recently, Chen and Donoho [7] have described Basis Pursuit as a technique for decomposing a signal into an optimal superposition of dictionary elements using as the optimization criterion the l^1 norm of coefficients.

The search for optimal basis amounts to identifying relevant feature subsets as a result of exploiting non-linear interactions in high dimensional feature spaces. The identification of optimal basis can be approached through the use of Genetic Algorithms (GAs) [17]. GAs work by maintaining a constant-sized population of candidate solutions known as individuals ('chromosomes'). The power of a genetic algorithm lies in its ability to exploit, in a highly efficient manner, information about a large number of individuals. The search underlying GAs is such that breadth and depth — exploration and exploitation — are balanced according to the observed performance of the individuals evolved so

far. By allocating more reproductive occurrences to above average individual solutions, the overall effect is to increase the population's average fitness. We advance in this paper an adaptive dictionary method for face recognition using GAs in determining the optimal basis for encoding human faces. In analogy to the pursuit methods referred to earlier our novel method is called Evolutionary Pursuit (EP). The EP method, takes advantage of both statistical and neural methods, and it is described in Sect. 4. EP allows for different types of bases, as some statistical methods do, but it would update the dictionary of choices simultaneously as neural networks do.

As systems that employ several strategies have been shown to offer significant advantages over single-strategy systems, we have developed a hybrid methodology seeking the basis representation for human faces that leads to optimal performance on face recognition tasks. The optimal basis for face recognition is usually defined in terms of 2nd order statistics. PCA related 2nd order methods and their use for face recognition are reviewed in Sect. 2 as they provide the benchmark for comparing our new hybrid and evolutionary methodology for face recognition. Sect. 3 describes the overall strategy for face recognition and the modules involved, while Sect. 4 details the evolutionary pursuit method for deriving the optimal basis and its use for face recognition. Experimental results are given in Sect. 5, while conclusions are presented in Sect. 6.

2 2nd Order Methods and Face Recognition

Principal Component Analysis (PCA), also known as the Karhunen-Loeve expansion, is a classical technique for signal representation [21], [16]. Sirovich and Kirby [32], [22] applied PCA for representing face images. They showed that any particular face can be economically represented along the eigenpictures coordinate space, and that any face can be approximately reconstructed by using just a small collection of eigenpictures and the corresponding projections ('coefficients') along each eigenpicture.

PCA generates a set of orthonormal basis vectors, known as principal components, that maximize the scatter of all projected samples. Let $X = [X_1, X_2, \ldots, X_n]$ be the sample set of the original images. After normalizing the images to unity norm and subtracting the grand mean a new image set $Y = [Y_1, Y_2, \ldots, Y_n]$ is obtained. Each Y_i represents a normalized image with dimensionality N, $Y_i = (y_{i_1}, y_{i_2}, \ldots, y_{i_N})^t$, $(i = 1, 2, \ldots, n)$. The covariance matrix of the normalized image set is defined as

$$\Sigma_Y = \frac{1}{n} \sum_{i=1}^{n} Y_i Y_i^t = \frac{1}{n} Y Y^t \tag{1}$$

and the eigenvector and eigenvalue matrices Φ, Λ are computed as

$$\Sigma_Y \Phi = \Phi \Lambda \tag{2}$$

Note that YY^t is an $N \times N$ matrix while Y^tY is an $n \times n$ matrix. If the sample size n is much smaller than the dimensionality N, then the following method saves some computation [35]

$$(Y^tY)\Psi = \Psi\Lambda_1 \tag{3}$$

$$\Im = Y\Psi \tag{4}$$

where $\Lambda_1 = diag\{\lambda_1, \lambda_2, \ldots, \lambda_n\}$, and $\Im = [\Phi_1, \Phi_2, \ldots, \Phi_n]$. If one assumes that the eigenvalues are sorted in decreasing order, $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$, then the first m leading eigenvectors define matrix P

$$P = [\Phi_1, \Phi_2, \ldots, \Phi_m] \tag{5}$$

The new feature set Z with lower dimensionality m ($m \ll N$) is derived

$$Z = P^tY \tag{6}$$

For pattern recognition, the PCA technique is exploited both directly and indirectly. The direct approaches use the principal components (PCs) as the projection basis, hence preserve the orthogonality of the basis vectors. The indirect methods use PCA primarily as a dimensionality reduction technique for subsequent transformations, and the overall projection basis vectors are usually no longer orthogonal. Unlike signal representation, orthogonality is not a requirement for pattern recognition, and one can expect better performance from non-orthogonal bases over orthogonal ones as they lead to an over complete and robust representational space [12].

Since eigenpictures are fairly good at representing face images, one can also consider using the projections along them as classification features to recognize faces. As a result, Turk and Pentland developed a well known face recognition method, known as eigenfaces, where the eigenfaces correspond to the eigenvectors associated with the dominant eigenvalues of the face covariance matrix. The eigenfaces define a feature space, or "face space", which drastically reduces the dimensionality of the original space, and face detection and identification are carried out in the reduced space [35].

The advantage of direct approaches (PCA only) is their generalization ability [27]. PCA yields projection axes based on the variations from all the training samples, hence these axes are fairly robust for representing both training and testing images (not seen during training). This is the merit of PCA as an optimal technique for signal representation. As a result, the performance during testing will not be very different from that encountered during training. In other words, direct approaches display good generalization ability. The disadvantage of the direct approaches is that they can not distinguish the variations between within and between class scatters, since PCA treats all the training samples equally. As a consequence, by maximizing the scatter measurement, the unwanted within class scatters are also maximized along with the between class scatter maximization. This will lead to poor performance when the within class scatter is big due to lighting, facial expression, pose, and duplicate images.

While PCA is a classical technique for signal representation, Fisher's Linear Discriminant (FLD) is a classical technique for pattern recognition [14], [3]. Several authors have applied this technique for face recognition, gesture recognition, and pattern rejection [8], [11], [1]. Recently Swets and Weng have pointed out that the eigenfaces derived using PCA are only the most expressive features (MEF), which are unrelated to actual face recognition. To derive most discriminating features (MDF), one needs a subsequent FLD projection [33]. Their procedure involves the simultaneous diagonalization of the two within and between class scatter matrices [16]. The MDF space is superior to the MEF space for face recognition only when the training images are representative of the range of face (class) variations; otherwise, the performance difference between the MEF and MDF is not significant. Belhumire, Hespanha, and Kriegman developed a similar approach called fisherfaces by applying first PCA for dimensionality reduction and then FLD for discriminant analysis [3].

The advantage of the indirect methods (combining PCA and FLD) is that they distinguish the different roles of within and between class scatter by applying discriminant analysis, e.g. FLD, and they usually produce non-orthogonal projection axes. But the indirect methods have their disadvantage too, namely poor generalization to new data, because those methods overfit to the training data. As the FLD procedure involves the simultaneous diagonalization of the two within and between class scatter matrices, it is equivalent to two-step operations: first 'whitening' the within class scatter matrix — applying an appropriate transformation that will make the within class scatter matrix equal to unity, and second applying PCA on the new between class scatter matrix [16]. Note that whitening as used here lacks generalization ability when compared to global whitening methods (see Sect. 3.1) which are applied across both within and between scatter matrices defined together as the covariance matrix Σ_Y (see Eq. 1). The purpose of the 'whitening' step here is to normalize the within class scatter to unity, while the second step would then maximize the between class scatter. The robustness of the FLD procedure thus depends on whether or not the within class scatter can capture enough variations for a specific class. When the training samples do not include most of the variations due to lighting, facial expression, pose, and/or duplicate images as those encountered during testing, the 'whitening' step is likely to fit misleading variations, i.e. the normalized within class scatter would best fit the training samples but it would generalize poorly when exposed to new data. As a consequence the performances during testing for such an indirect method will deteriorate. In addition, when the training sample size for each class is small, the within class scatter would usually not capture enough variations. The FLD procedure thus leads to overfitting.

The Evolutionary Pursuit (EP) approach detailed in the following sections would take into consideration of both performance accuracy and generalization capability and evolve balanced results displaying good performance during testing. As a result, EP improves the face recognition performance compared to (direct) PCA (Eigenfaces), and shows better generalization abilities than the FLD based (indirect) methods (MDF/Fisherfaces).

3 Optimal Basis and Face Recognition

Our architecture for face recognition is shown in Fig. 1. The main thrust is to find out an optimal basis along which faces can be projected leading to a compact and efficient face encoding in terms of recognition ability. As discussed in the previous section, PCA first projects the face images into a lower dimensional space. The next step is the whitening transformation and it counteracts the fact that the Mean-Square-Error (MSE) principle underlying PCA preferentially weights low frequencies. Directed but random rotations of the lower dimensional (whitened PCA) space are now driven by evolution and use domain specific knowledge ('fitness'). The fitness behind evolution, the one used to find the optimal basis, considers both recognition rates and the scatter index which are derived using the projections of the face images onto the rotated axes. Evolution is implemented using Evolutionary Pursuit (EP) as a special form of Genetic Algorithms (GAs). Note that the reachable space of EP is increased as a result of using a non-orthonormal (whitening) transformation. One can expect better performance from non-orthogonal bases over orthogonal ones as they lead to an over complete and robust representational space [12]. Note that under the whitening transformation the norms (distances) are not preserved.

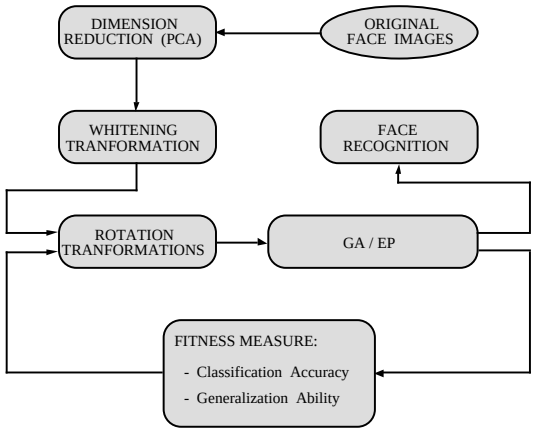

Fig. 1. System Architecture for Face Recognition using Evolutionary Pursuit

3.1 Whitening Transformation

After dimensionality reduction using PCA, the lower dimensional feature set Z (from Eq. 6) is now subjected to the whitening transformation and leads to another feature set V

$$V = \Gamma Z \tag{7}$$

where $\Gamma = diag\{\lambda_1^{-1/2}, \lambda_2^{-1/2}, \ldots, \lambda_m^{-1/2}\}$.

The reason why the whitening procedure can lead to non-orthogonal bases of the overall transformation is as follows. Let Q be a $m \times m$ rotation matrix ($Q^tQ = QQ^t = I$) and apply Q to the feature set V. Combined with Eqs. 6 and 7 one obtains the overall transformation matrix Ξ

$$\Xi = P\Gamma Q \tag{8}$$

Now assume the basis vectors in Ξ are orthogonal (using *proof by contradiction*),

$$\Xi^t\Xi = \Delta \tag{9}$$

where Δ is a diagonal matrix. From Eqs. 8 and 9 it follows that

$$\Gamma^2 = \Delta = cI \tag{10}$$

where c is a constant. Eq. 10 holds only when all the eigenvalues are equal, and when this is not the case the basis vectors in Ξ are not orthogonal. (see Fig. 6).

3.2 Rotation Transformations

The rotation transformations are carried out in the whitened m dimensional space, in which the feature set V lies (see Eq. 7). Let $\Omega = [\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_m]$ be the basis of this space where $\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_m$ are the unit vectors. Our evolutionary pursuit approach would later on search for that (reduced) subset of some basis vectors rotated from $\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_m$ in terms of best discrimination performance. The rotation procedure is carried out by pairwise axes rotation. In particular, let us suppose the basis vectors ε_i and ε_j need to be rotated by α_k, then a new basis $\xi_1, \xi_2, \ldots, \xi_m$ is derived by

$$[\xi_1, \xi_2, \ldots, \xi_m] = [\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_m]Q_k \tag{11}$$

where Q_k is a rotation matrix. There are $M = m(m-1)/2$ rotation angles in total corresponding to the M pairs of basis vectors to be rotated. For the purpose of evolving optimal basis for recognition, it makes no difference if the angles are confined to $(0, \pi/2)$, since the positive directions and the order of axes are not important. The overall rotation matrix Q is defined by

$$Q = Q_1Q_2 \cdots Q_{m(m-1)/2} \tag{12}$$

3.3 Face Recognition

Let $T = [\Theta_{i_1}, \Theta_{i_2}, \ldots, \Theta_{i_l}]$ be the optimal basis derived by EP (evolutionary pursuit) (see Sect. 4.2). The new feature set U is derived as

$$U = [U_1, U_2, \ldots, U_n] = T^tV \tag{13}$$

where V is the whitened feature set (Eq. 7), and $[U_1, U_2, \ldots, U_n]$ are the feature vectors corresponding to different face images.

Let $U_k^0, (k = 1, 2, \ldots, n)$, be the prototype of class k, the decision rule can be expressed as

$$\|U_i - U_k^0\|_2 = \min_j \|U_i - U_j^0\|_2, \qquad U_i \in \omega_k \tag{14}$$

The face image U_i is classified to the class ω_k to which it has the minimum Euclidean distance.

4 Genetic Algorithms (GAs) and Evolutionary Pursuit (EP)

The task for EP is to search through all the rotation axes defined over properly whitened PCA subspaces. Evolution is driven by a fitness function defined in terms of performance accuracy and class separation (scatter index). Accuracy indicates the extent to which learning has been successful so far, while the scatter index gives an indication of the expected fitness on future trials. A large scatter index calls for additional learning so present performance becomes a good indicator on future ('predicted') performance. Predictors on future performance can thus modulate the amount of learning and stop learning, when constant but well behaved performance can be expected from the individual chromosomes. EP defined as above is thus a hybrid between the "filter" and "wrapper" approaches [23] and takes advantage of their comparative merits.

The EP method is implemented using GAs and it has the following advantages. First, directed search in the whitened PCA subspaces which are more reliable than the whitened subspaces of the within class scatter matrix (see MDF for comparison); the reason is that PCA exploits the variations from all the training samples while the within class scatter uses only within class variations. When the sample size of each class is small and the variations are not representative, the whitened subspaces of the within class scatter matrix do not represent the actual unit within class scatter any more. Second, the fitness function consists of two terms: performance accuracy and class separation. These two terms put opposite pressures on the fitness function: the performance accuracy term is similar to the criterion of choosing projection axes with smaller scatter, while the class separation term favors axes with larger scatter. By combining these two terms together (with proper weights), GA can evolve balanced results with good testing performances and generalization abilities.

One should also point out that just using more PCs (principal components) does not necessarily lead to better performance, since some PCs might capture the within class scatter which is unwanted for the purpose of recognition. In our experiments we searched the 20 and 30 dimensional whitened PCA subspaces corresponding to the leading eigenvalues, since it is in those subspaces that most of the variations characteristic of human faces occur.

4.1 Chromosome Representation and Genetic Operators

As is discussed in Sect. 3.2, corresponding to different sets of rotation angles different basis vectors are derived. GAs are used to search among the differ-

ent rotation transformations and different combinations of basis vectors in order to pick up the best subset of vectors with the most discriminant power. The optimal basis is evolved from a larger vector set $\{\xi_1, \xi_2, \ldots, \xi_m\}$ rotated from a basis $\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_m$ in m dimensional space by a set of rotation angles $\alpha_1, \alpha_2, \ldots, \alpha_{m(m-1)/2}$ with each angle in the range of $(0, \pi/2)$. If the angles are discretized with small enough steps, then we can use GA to search this discretized space. GA requires the solutions to be represented in the form of bit strings or chromosomes. If we use 10 bits (resolution) to represent each angle, then each discretized (angle) interval is less than 0.09 degree, and we need $10 * [m(m-1)/2]$ bits to represent all the angles. As we also have m basis vectors (projection axes) to choose from, another m bits should be added to the chromosome to facilitate that choice. Fig. 2 shows the chromosome representation, where $a_i, (i = 1, 2, \ldots, m)$, has the value 0 or 1 and indicates whether the i-th basis vector is chosen or not.

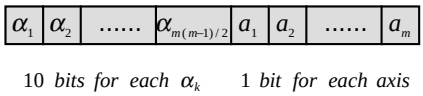

Fig. 2. Chromosome Representation of Rotation Angles and Projection Axes

Let N_s be the number of different choices of basis vectors in the search space. The size of genospace, too large to search it exhaustively, is

$$N_s = 2^{5m(m-1)+m} \tag{15}$$

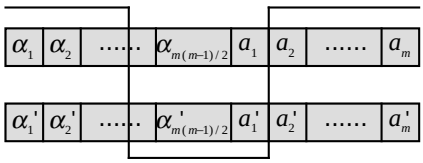

Fig. 3. Two Points Crossover

As it searches the genospace, the GA makes its choices via genetic operators as a function of a probability distribution driven by the fitness function. The genetic operators are selection, crossover (or recombination), and mutation [17]. In our experiments, we use (i) proportionate selection: preselection of parents in proportion to their relative fitness; (ii) two points crossover: exchange the sections between the crossover points as shown in Fig. 3; and (iii) fixed probability mutation: each position of a chromosome is given a fixed probability of undergoing mutation (flipping the corresponding bit).

4.2 The Fitness Function

Fitness values guide GA on how to choose offsprings for the next generation from the current parent generation. Let $F \equiv \alpha_1, \alpha_2, \ldots, \alpha_{m(m-1)/2}; a_1, a_2, \ldots, a_m$ represent the parameters to be evolved by GA, then the fitness function $\zeta(F)$ is defined as

$$\zeta(F) = \zeta_a(F) + \lambda\zeta_s(F) \tag{16}$$

where $\zeta_a(F)$ is the performance accuracy term, $\zeta_s(F)$ is the class separation term, and λ is a positive constant. In our experiments, we set $\zeta_a(F)$ to be the number of faces correctly recognized as the top choice after the rotation and selection of a subset of axes, and $\zeta_s(F)$ the scatter measurement among different classes. λ is empirically chosen such that $\zeta_a(F)$ contributes more to the fitness than $\zeta_s(F)$ does. Note that the fitness function defined here has a similar form compared to the cost functional derived from the principle of regularization theory, which is very useful for solving ill-posed problems in computer vision and improving the generalization ability of RBF networks in neural network [34], [29], [18]. Actually, those two terms, $\zeta_a(F)$ and $\zeta_s(F)$, put opposite pressures on the fitness function: the performance accuracy term $\zeta_a(F)$ is similar to the criterion of choosing projection axes with smaller scatter, while the class separation term $\zeta_s(F)$ favors axes with lager scatter. By combining those two terms together (with proper weight λ), GA can evolve balanced results displaying good performance during testing.

Let the rotation angle set be $\alpha_1^{(k)}, \alpha_2^{(k)}, \ldots, \alpha_{m(m-1)/2}^{(k)}$, and the basis vectors after the transformation be $\xi_1^{(k)}, \xi_2^{(k)}, \ldots, \xi_m^{(k)}$ according to Eqs. 11 and 12. If GA chooses l vectors $\eta_1, \eta_2, \ldots, \eta_l$ from $\xi_1^{(k)}, \xi_2^{(k)}, \ldots, \xi_m^{(k)}$, then the new feature set is specified as

$$W = [\eta_1, \eta_2, \ldots, \eta_l]^t V \tag{17}$$

where V is the whitened feature set (see Eq. 7).

Let $\omega_1, \omega_2, \ldots, \omega_L$ and $N_1, N_2, \ldots, N_L$ denote the classes and number of images within each class, respectively. Let $M_1, M_2, \ldots, M_L$ and M_0 be the means of corresponding classes and the grand mean in the new feature space $span[\eta_1, \eta_2, \ldots, \eta_l]$, we then have

$$M_i = \frac{1}{N_i}\sum_{j=1}^{N_i} W_j^{(i)}, \qquad i = 1, 2, \ldots, L \tag{18}$$

where $W_j^{(i)}$, $j = 1, 2, \ldots, N_i$, represent the sample images from class ω_i, and

$$M_0 = \frac{1}{n}\sum_{i=1}^{L} N_i M_i \tag{19}$$

where n is the total number of images for all the classes. Thus, $\zeta_s(F)$ is computed as

$$\zeta_s(F) = \sqrt{\sum_{i=1}^{L}(M_i - M_0)^2} \tag{20}$$

Driven by this fitness function, GA would evolve the optimal solution $F^o \equiv \alpha_1^o, \alpha_2^o, \ldots, \alpha_{m(m-1)/2}^o; a_1^o, a_2^o, \ldots, a_m^o$. Let Q in Eq. 12 represent this particular basis set corresponding to the rotation angles $\alpha_1^o, \alpha_2^o, \ldots, \alpha_{m(m-1)/2}^o$ (remember $\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_m$ are unit vectors), and let the column vectors in Q be $\Theta_1, \Theta_2, \ldots, \Theta_m$

$$Q = [\Theta_1, \Theta_2, \ldots, \Theta_m] \tag{21}$$

Let $\Theta_{i_1}, \Theta_{i_2}, \ldots, \Theta_{i_l}$ be the basis vectors corresponding to $a_1^o, a_2^o, \ldots, a_m^o$, then the optimal basis T can be expressed as

$$T = [\Theta_{i_1}, \Theta_{i_2}, \ldots, \Theta_{i_l}] \tag{22}$$

where $i_j \in \{1, 2, \ldots, m\}$, $i_j \neq i_k$ for $j \neq k$, and $l < m$.

4.3 The Evolutionary Pursuit (EP) Algorithm

The evolutionary pursuit (EP) algorithm works as follows:

1. Compute the eigenvector and eigenvalue matrices of Y^tY using singular value decomposition (SVD) or Jacobi's method, and derive $\Lambda_1 = diag\{\lambda_1, \lambda_2, \ldots, \lambda_n\}$ and $\Im = [\Phi_1, \Phi_2, \ldots, \Phi_n]$ according to Eqs. 3 and 4. Choose then the first m leading eigenvectors from $\Im$ as basis vectors (Eq. 5) and project the original image set Y onto those vectors to form the feature set Z (Eq. 6) in this reduced PCA subspace.

2. Whiten the feature set Z and derive the new feature set V in the whitened PCA subspace (Eq. 7).

3. Set $[\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_m]$ to be a $m \times m$ unit matrix: $[\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_m] = I_m$.

4. Begin the evolution loop until the stopping criteria (e.g., the maximum number of trials) are reached:

   (a) Sweep the $m(m-1)/2$ pairs of axes according to a fixed order to get the rotation angle set $\alpha_1^{(k)}, \alpha_2^{(k)}, \ldots, \alpha_{m(m-1)/2}^{(k)}$ from the individual chromosome representation (Fig. 2), and rotate the unit basis vectors, $[\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_m]$, in this m dimensional space to derive the new projection axes: $\xi_1^{(k)}, \xi_2^{(k)}, \ldots, \xi_m^{(k)}$ using Eqs. 11 and 12.

   (b) Compute the fitness value (Eq. 16) in the feature space defined by the l projection axes, $\eta_1, \eta_2, \ldots, \eta_l$, chosen from the rotated axes set $\left\{\xi_1^{(k)}, \xi_2^{(k)}, \ldots, \xi_m^{(k)}\right\}$ according to the $a_i's$ from the individual chromosome representation (Fig. 2).

   (c) Find the sets of angles and the subsets of projection axes that maximize the fitness value, and keep these chromosomes as the best solutions so far.

(d) Change the values of rotation angles and the subsets of the projection axes according to GA's genetic operators, and repeat the evolution loop.

5. Carry out recognition using Eqs. 22, 13, and 14, after GA evolves the optimal basis, $\Theta_{i_1}, \Theta_{i_2}, \ldots, \Theta_{i_l}$.

The computational complexity of the algorithm falls mainly into two parts: the PCA computation of step 1 and the evolution loop of step 4. In step 1, the SVD of matrix of size $n \times n$ has the complexity of $O(n^3)$ according to [5], the computation of the eigenvector matrix $\Im$ (Eq. 4) is $O(n^2 N)$, and the derivation of the feature set Z (Eq. 6) is $O(mnN)$. In step 4, the rotation transformations of (a) and the fitness value computations of (b) account for most of the computation. In step 4 (a), each rotation transformation changes two column vectors (pairwise axes rotation), and there are $m(m-1)/2$ rotations in total, hence the complexity is $O(m^3)$. In step 4 (b), if we only count the number of multiplications, then Eq. 17 accounts for the major part of the computation with the computational complexity $O(lmn)$. The overall complexity of the evolution procedure also depends on the maximum number of trials.

5 Experimental Results

The experimental data consists of 1107 facial images corresponding to 369 subjects and it comes from the US Army FERET database [28]. 600 out of the 1107 images correspond to 200 subjects with each subject having three images — two of them are the first and the second shot, and the third shot is taken under low illumination (see Fig. 4). For the remaining 169 subjects there are also three images for each subject, but two out of the three images are duplicates taken at a different time (see Fig. 4). Two images of each subject are used for training with the remaining image for testing. The images are cropped to the size of 64×96, and the eye coordinates are manually detected.

We implemented the evolutionary pursuit (EP) algorithm with $m = 20$ and $m = 30$, respectively (PCA reduces the dimensionality of the original image space from $N = (64 \times 96)$ to m). The Eigenface and MDF methods were implemented and experimented with as well. Note that once EP found a reduced subset of basis vectors, the same number of projection axes was used by both the eigenface and MDF methods for comparison purposes (see Tables 1, 2 and 3). Table 1 shows comparative training performance, while Tables 2 and 3 give comparative testing performance. In Table 2 and 3, top 1 recognition rate means the accuracy rate for the top response being correct, while top 3 recognition rate represents the accuracy rate for the correct response being included among the first three ranked choices.

When $m = 20$, the evolutionary pursuit approach derived 18 vectors as the optimal basis. Fig. 5 plots the 18 basis vectors, and Fig. 6 shows the non-orthogonality of these vectors. For each row (or column) the unit bar (along the diagonal position) represents the norm of a basis vector, and the other bars

represent the dot products of this vector and the other 17 basis vectors. Since the dot products are non-zero, these basis vectors are not orthogonal. When $m = 30$, the EP approach derived 26 vectors as the optimal basis.

Table 1. Comparative **Training Performances** for the Eigenface, MDF, and Evolutionary Pursuit Methods Using **18** and **26** Basis Vectors, respectively

method \features	18	26
Eigenface Method	78.05%	81.30%
MDF Method	100%	100%
Evolutionary Pursuit	83.47%	82.66%

Table 2. Comparative **Testing Performances** for the Eigenface, MDF, and Evolutionary Pursuit Methods when the **20** dimensional whitened PCA subspace is searched by EP ($m = \mathbf{20}$)

method	# features	top 1 recognition rate	top 3 recognition rate
Eigenface Method	18	81.57%	94.58%
MDF Method	18	79.95%	87.80%
Evolutionary Pursuit	18	87.80%	95.93%

Table 3. Comparative **Testing Performances** for the Eigenface, MDF, and Evolutionary Pursuit Methods when the **30** dimensional whitened PCA subspace is searched by EP ($m = \mathbf{30}$)

method	# features	top 1 recognition rate	top 3 recognition rate
Eigenface Method	26	87.26%	95.66%
MDF Method	26	86.45%	93.77%
Evolutionary Pursuit	26	92.14%	97.02%

Table 1 gives the comparative training performances of Eigenface, MDF, and Evolutionary Pursuit methods with 18 and 26 basis vectors, respectively, and one can see that the training performances for MDF method is perfect (100% correct recognition rate). During testing (see Tables 2 and 3) and using 369 test images (not used during training), the performance displayed by the MDF method, however, deteriorates as it lacks a good generalization ability. Both the Eigenface and EP approach display better generalization abilities when compared against MDF. In particular, Table 2 shows that when the 20 dimensional whitened PCA

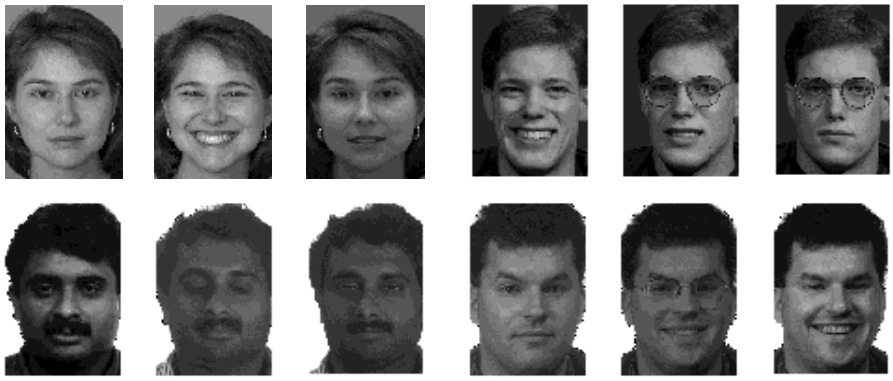

Fig. 4. Examples of Face Images from FERET Database

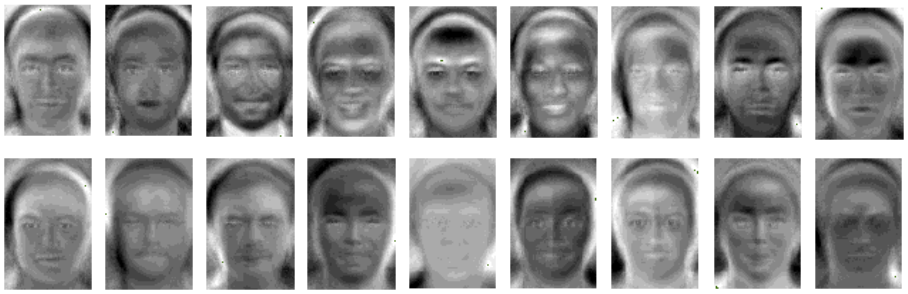

Fig. 5. Optimal Basis (18 Vectors) Derived by the Evolutionary Pursuit (EP) Method

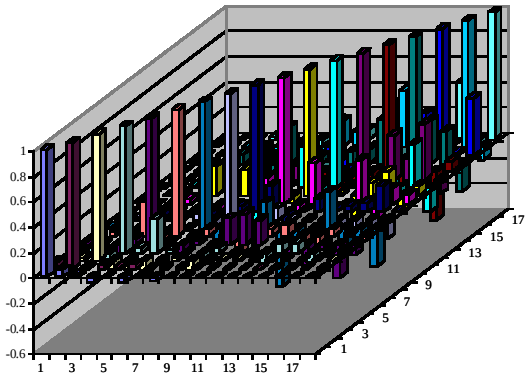

Fig. 6. Non-orthogonality of the Basis Vectors Derived by Evolutionary Pursuit

subspace is searched, the EP approach derives 18 vectors as the optimal basis with top 1 recognition rate 87.80% compared to 81.57% for the Eigenface method and 79.95% for the MDF method. For top 3 recognition rate, the EP approach again comes first and yields 95.93%, compared to 94.58% for Eigenface and 87.80% for MDF method. When the EP approach evolves the optimal basis in the 30 dimensional whitened PCA subspace (see Table 3), it requires only 26 vectors for its optimal basis and achieves 92.14% top 1 recognition rate, compared to 87.26% for Eigenface and 86.45% for the MDF methods. For top 3 recognition rate, the EP approach yields 97.02%, compared to 95.66% for Eigenface and 93.77% for the MDF method.

From Tables 1, 2 and 3 it becomes apparent that MDF does not display good generalization abilities, while PCA and the evolutionary pursuit approach do. The range of training data is quite large as it consists of both original and duplicate images acquired at a later time. As a consequence, during training, MDF performs better than both the Eigenface and evolutionary pursuit (EP) methods because it overfits to a larger extent its classifier to the data. Evolutionary pursuit yields, however, improved performances over the other two methods, during testing.

6 Conclusions

This paper describes an adaptive dictionary method for face recognition using GAs in determining the optimal basis for encoding human faces. In analogy to pursuit methods, our novel method is called Evolutionary Pursuit (EP), and it allows for different types of bases, as some statistical methods do, but it updates the dictionary of choices ('kernels') simultaneously as neural networks do. The main thrust of the EP method is to find out an optimal basis along which faces can be projected leading to a compact and efficient face encoding in terms of recognition ability. EP processes face images in a lower dimensional space defined as PCA projections. The projections are then whitened to counteract the fact that the Mean-Square-Error (MSE) principle underlying PCA preferentially weights low frequencies. The reachable space of EP is increased as a result of using a non-orthonormal (whitening) transformation. One can expect better performance from non-orthogonal bases over orthogonal ones as they lead to an over complete and robust representational space. Directed but random rotations of the lower dimensional (whitened PCA) space are then searched by GAs and use domain specific knowledge ('fitness'). Experimental results show that the EP approach compares favorably against the two methods for face recognition — the Eigenfaces and MDF methods.

The fitness driving evolution considers both recognition rates ('performance accuracy') — empirical risk — and the scatter index — predicted risk — corresponding to the projections of the face images onto the rotated axes. The fitness function is similar to cost functionals implementing regularization methods for ill-posed problems in computer vision. The prediction risk, included as a penalty, is a measure of generalization ability and is driven by the scatter index ('class

separation'). The relative contribution of performance accuracy and the scatter index to the fitness function is given through a positive weight parameter λ. The weight parameter indicates the degree of generalization expected from the EP method. In one of the limiting cases, $\lambda \to 0$ implies that only performance accuracy defines fitness and the derived optimal basis will display poor generalization abilities. The other limiting case, $\lambda \to \infty$ implies that now it is the scatter index which fully defines fitness and the derived optimal basis will display poor recognition rates. The weight parameter used for the experimental data presented earlier gives more weight to the empirical risk than to the predicted risk.

As 2nd order statistics provide only partial information on the statistics of both natural images and human faces it becomes necessary to consider higher order statistics as well. Towards that end and in analogy to recent methods such as Independent Component Analysis (ICA) [10] we plan to expand the EP method so it can also consider higher order statistics when deriving the optimal basis — neural code.

Acknowledgments: This work was partially supported by the DoD Counterdrug Technology Development Program, with the U.S. Army Research Laboratory as Technical Agent, under contract DAAL01-97-K-0118.

References

1. S. Baker and S.K. Nayar: Pattern Rejection. Proc. IEEE Conf. Computer Vision and Pattern Recognition (1996) 544–549
2. H.B. Barlow: Unsupervised Learning. Neural Computation **1** (1989) 295–311
3. P.N. Belhumeur, J.P. Hespanha, and D.J. Kriegman: Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection. IEEE Trans. Pattern Analysis and Machine Intelligence **19** (1997) 711–720
4. R. Brunelli and T. Poggio: Face Recognition: Features vs. Templates. IEEE Trans. Pattern Analysis and Machine Intelligence **15** (1993) 1042–1053
5. T.F. Chan: An Improved Algorithm for Computing the Singular Value Decomposition. ACM Trans. Math. Software **8** (1982) 72–83
6. R. Chellappa, C.L. Wilson, and S. Sirohey: Human and Machine Recognition of Faces: A Survey. Proc. IEEE **83** (1995) 705–740
7. S. Chen and D. Donoho: Basis Pursuit. Technical Report, Department of Statistics, Stanford University (1996)
8. Y. Cheng, K. Liu, J. Yang, Y. Zhuang, and N. Gu: Human Face Recognition Method Based on the Statistical Model of Small Sample Size. SPIE Proc. Intelligent Robots and Computer Vision X: Algorithms and Technology (1991) 85–95
9. Y. Cherkassky and F. Mulier: Learning from Data : Concepts, Theory and Methods. Wiley (1998) (to appear)
10. P. Comon: Independent Component Analysis — A New Concept?. Signal Processing **36** (1994) 11–20
11. Y. Cui, D. Swets, and J. Weng: Learning-Based Hand Sign Recognition Using SHOSLIF-M. Int'l Conf. on Computer Vision (1995) 45–58
12. J.G. Daugman: An information-theoretic view of analog representation in striate cortex. in Computational Neuroscience, E.L. Schwartz, eds. MIT Press (1990) 403–424

13. G.M. Edelman: Neural Darwinism. Basic Books (1987)
14. R.A. Fisher: The Use of Multiple Measures in Taxonomic Problems. Ann. Eugenics **7** (1936) 179–188
15. J.H. Friedman and W. Stuetzle: Projection Pursuit Regression. J. Amer. Statist. Asso. **76** (1981) 817–823
16. K. Fukunaga: Introduction to Statistical Pattern Recognition, 2nd Edition. Academic Press (1991)
17. D. Goldberg: Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley (1989)
18. S. Haykin: Neural Networks — A Comprehensive Foundation. Macmillan College Publishing Company, Inc. (1994)
19. P.J. Huber: Projection Pursuit. Ann. Stat. **13** (1985) 435–475
20. J. Hwang, S. Lay, M. Maechler, R.D. Martin, and J. Schimert: Regression Modeling in Back-Propagation and Projection Pursuit Learning. IEEE Trans. Neural Networks **5** (1994) 342–353
21. I.T. Jolliffe: Principal Component Analysis. Springer, New York (1986)
22. M. Kirby and L. Sirovich: Application of the Karhunen-Loeve Procedure for the Characterization of Human Faces. IEEE Trans. Pattern Analysis and Machine Intelligence **12** (1990) 103–108
23. R. Kohavi and G. John: Wrappers for Feature Subset selection. Technical Report, Computer Science Department, Stanford University (1995)
24. R. Linsker: Self-organization in a Perceptual Network. Computer **21** (1988) 105–117
25. S.G. Mallat and Z. Zhang: Matching Pursuits With Time-Frequency Dictionaries. IEEE Trans. Signal Processing **41** (1993) 3397–3415
26. B.A. Olshausen and D.J. Field: Emergence of Simple-cell Receptive Field Properties by Learning a Sparse Code for Natural Images. Nature **381** (1996) 607–609
27. P.S. Penev and J.J. Atick: Local Feature Analysis: A general statistical theory for object representation. Network: Computation in Neural Systems **7** (1996) 477–500
28. P.J. Phillips, H. Wechsler, J. Huang, and P. Rauss: The FERET Database and Evaluation Procedure for Face Recognition Algorithms. Image and Vision Computing (1998) (to appear)
29. T. Poggio, V. Torre, and C. Koch: Computational Vision and Regularization Theory. Nature **317** (1985) 314–319
30. D. Ruderman: The statistics of Natural Images. Network : Computation in Neural Systems **5** (1994) 598–605
31. A. Samal and P.A. Iyengar: Automatic Recognition and Analysis of Human Faces and Facial Expression: A Survey. Pattern Recognition **25** (1992) 65–77
32. L. Sirovich and M. Kirby: Low-dimensional Procedure for the Characterization of Human Faces. J. Optical. Soc. Am. A **4** (1987) 519–524
33. D.L. Swets and J. Weng: Using Discriminant Eigenfeatures for Image Retrieval. IEEE Trans. Pattern Analysis and Machine Intelligence **18** (1996) 831–836
34. A.N. Tikhonov and V.Y. Arsenin: Solutions of Ill-posed Problems. W.H. Winston, Washington, DC (1977)
35. M. Turk and A. Pentland: Eigenfaces for Recognition. Journal of Cognitive Neuroscience **3** (1991) 71–86