Representational Oriented Component Analysis (ROCA) for Face Recognition with One Sample Image per Training Class

Fernando De la Torre† Ralph Gross† Simon Baker† B.V.K. Vijaya Kumar‡

†, Robotics Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania 15213.
ftorre@cs.cmu.edu ralph@cs.cmu.edu simonb@cs.cmu.edu kumar@ece.cmu.edu

‡, Department of Electrical and Computer Engineering. Carnegie Mellon University.
Pittsburgh, Pennsylvania 15213. kumar@ece.cmu.edu

Abstract

Subspace methods such as PCA, LDA, ICA have become a standard tool to perform visual learning and recognition. In this paper we propose Representational Oriented Component Analysis (ROCA), an extension of OCA, to perform face recognition when just one sample per training class is available. Several novelties are introduced in order to improve generalization and efficiency:

- *Combining several OCA classifiers based on different image representations of the unique training sample is shown to greatly improve the recognition performance.*
- *To improve generalization and to account for small misregistration effect, a learned subspace is added to constrain the OCA solution,*
- *A stable/efficient generalized eigenvector algorithm that solves the small size sample problem and avoids overfitting.*

Preliminary experiments in the FRGC Ver 1.0 dataset (http://www.bee-biometrics.org/) show that ROCA outperforms existing linear techniques (PCA,OCA) and some commercial systems.

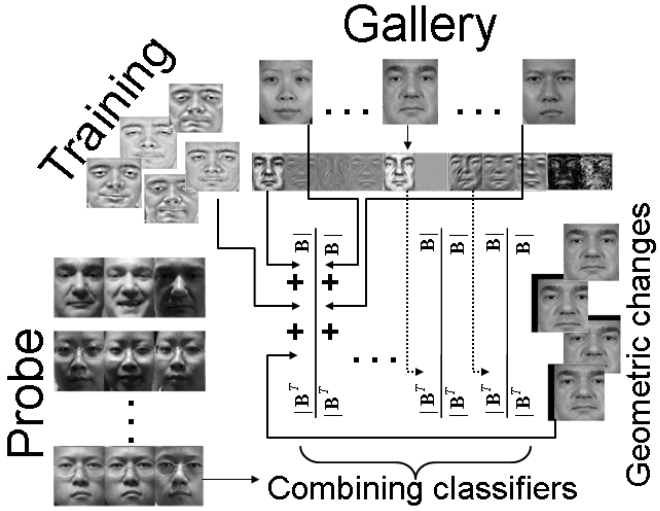

Figure 1: For each of the unique image per person in the gallery set several representations are constructed. For each representation of class i an OCA classifier, which maximizes the response to shifted versions of the training image while minimizes it for the rest of the classes, is built. Later, OCA classifiers are combined to match each of the probe images.

1 Introduction

Subspace methods (SM) such as Principal Component Analysis (PCA), Independent Component Analysis (ICA), Linear Discriminant Analysis (LDA), Canonical Correlation Analysis (CCA), Oriented Component Analysis (OCA), etc have been extensively used for classification, dimensionality reduction and data modeling. The modeling power of SM is especially useful when available data increase in features/samples, since there is a need for dimensionality reduction while preserving relevant attributes of the data [1]. SM have been very successful in computer vision to solve problems such as structure from motion, detection/recognition, information retrieval and face tracking. In particular, among several classification methods (e.g. SVM, decision trees), SM remain a powerful preliminary tool for dimensionality reduction to preserve discriminative features that can avoid the "curse of dimensionality" and filter undesirable noise. For these reasons, SM

[1] Also many times it is helpful to find a new coordinate system (e.g. Fourier transform in the context of correlation filters [10])

have been extensively used in the context of face recognition. See [19, 22] for a review.

This paper introduces ROCA, a new classification scheme that performs face recognition by combining several OCA classifiers that are built on different representations of a unique training sample. Each of the representations provides useful discriminatory information, which if fused properly greatly improves the recognition performance. One of the major problems for each individual OCA classifier is the lack of generalization and overfitting issues due to the small training size (1 sample). To avoid such phenomena, several strategies are proposed. Firstly, using a weighted factorization of the covariance matrices allows more stable OCA classifiers. Secondly, a subspace that accounts for small misregistrations is proposed, it avoids specializing OCA to noisy directions. Finally, a generalized eigensolver that is able to deal with high dimensional data and to avoid the small sample size problem is also suggested. Preliminary experiments in the Face Recognition Grand Challenge (FRGC) Ver 1.0 dataset (http://www.bee-biometrics.org/) show that ROCA outperforms existing linear techniques (PCA,OCA) and some commercial systems. For instance, in experiment 4, ROCA achieves 75.5% recognition rate, whereas traditional techniques such as PCA (33%), Nearest Neighbour (27%), traditional OCA (23%) and a commercial system (41%) perform poorly. Figure 1 illustrates the main points of the paper.

2 Oriented Component Analysis

Matched filter theory [16, 18] and Oriented Component Analysis (OCA) [4] are similar linear statistical techniques whose main aim is to maximize the response of a wanted signal while minimizing it with respect to an unwanted signal (e.g. noise). For instance, the classical matched filter [18] seeks to maximize the square signal to noise ratio $(\frac{S}{N}(\mathbf{m}))^2 = \frac{(\mathbf{m}^T\mathbf{f})^2}{\mathbf{m}^T\boldsymbol{\Sigma}_n\mathbf{m}}$, where[2] $\mathbf{m}$ is the matched filter to be designed, $\boldsymbol{\Sigma}_n$ is an estimate of the noise covariance, and $\mathbf{f}$ is the mean of the signal to be detected. Observe that if the noise is decorrelated and has unit variance, i.e. $(\boldsymbol{\Sigma}_n = \mathbf{I})$, the best match is the signal itself (or the complex conjugate, if it is imaginary). Similarly, if a second order statistical descriptor of the signal of interest is provided $\boldsymbol{\Sigma}_x$, OCA maximizes the signal to signal ratio between two random vectors $\mathbf{x}, \mathbf{n}$

$$\max_{\mathbf{B}} \frac{|\mathbf{B}^T\boldsymbol{\Sigma}_x\mathbf{B}|}{|\mathbf{B}^T\boldsymbol{\Sigma}_n\mathbf{B}|} \tag{1}$$

where $\boldsymbol{\Sigma}_x$ and $\boldsymbol{\Sigma}_n$ are covariance or correlation matrices. The optimal $\mathbf{B}$ will preserve the directions of maximum variation of $\mathbf{x}$, which do not have high projection in the $\mathbf{n}$ directions. A closed form solution of eq. 1 is given by the following generalized eigenvalue problem, $\boldsymbol{\Sigma}_x\mathbf{B} = \boldsymbol{\Sigma}_n\mathbf{B}\Lambda$. The generalized eigenvalue problem is equivalent to a joint diagonalization, that is, finding a common basis $\mathbf{B}$ that simultaneous diagonalizes both matrices $\boldsymbol{\Sigma}_x$ and $\boldsymbol{\Sigma}_n$ (i.e. $\mathbf{B}^T\boldsymbol{\Sigma}_n\mathbf{B} = \mathbf{I}$ and $\mathbf{B}^T\boldsymbol{\Sigma}_x\mathbf{B} = \boldsymbol{\Lambda}$).

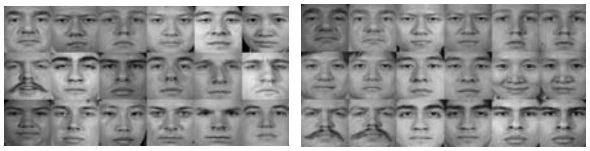

Figure 2: a) Gallery b) Probe

2.1 OCA for face recognition

The classical recognition system can be divided into two main blocks: feature extraction and classification (usually trained separately). The goal of feature extraction is to reduce the complexity of the original signal, subspace methods (ICA,PCA, etc) have commonly performed this task [17]; however, the classification may suffer if the most discriminative features are not extracted. In this section, we explore the use of OCA as a dimensionality reduction step and as a classifier for face recognition. Although not previously used in the context of computer vision, OCA has been applied to the speaker verification task [11]. Similar in spirit to OCA, Moghaddam et *al.* [13] have posed the face recognition problem (a c-ary classification problem) as a binary pattern classification with two classes.

Let $\mathbf{D} \in \Re^{d\times n}$ be a data matrix, such that each column $\mathbf{d}_i$ is a vectorized image. Let $\mathbf{G} \in \Re^{n\times c}$ be a dummy indicator matrix such that $\sum_j g_{ij} = 1$, $g_{ij} \in \{0,1\}$ and g_{ij} is 1 if $\mathbf{d}_i$ belongs to class j. c denotes the number of classes and n the number of images in the gallery set. For each class i, OCA will design a basis $\mathbf{B}_i$, which will maximize eq.1. Where $\boldsymbol{\Sigma}_x^i = \mathbf{D}\mathbf{g}_i\mathbf{g}_i^T\mathbf{D}^T$ will contain the autocorrelation matrix of the sample in class i and $\boldsymbol{\Sigma}_n^i = \frac{1}{n-1}\mathbf{D}(\mathbf{G}\mathbf{G}^T - \mathbf{g}_i\mathbf{g}_i^T)\mathbf{D}^T$ the extra class variation.

In our recognition challenge, we have just one sample per training class (gallery images) and several images per class in the testing set (probe). Fig 2.a shows some examples of the gallery images and Fig. 2.b some of the

[2] Bold capital letters denote a matrix $\mathbf{D}$, bold lower-case letters a column vector $\mathbf{d}$. $\mathbf{d}_j$ represents the j column of the matrix $\mathbf{D}$. All non-bold letters will represent variables of scalar nature. *diag* is an operator which transforms a vector to a diagonal matrix. $\mathbf{1}_k \in \Re^{k\times 1}$ is a vector of ones. $\mathbf{I}_k \in \Re^{k\times k}$ is the identity matrix and $\mathbf{e}_i$ is the i column. $tr(\mathbf{A}) = \sum_i a_{ii}$ is the trace of the matrix $\mathbf{A}$. $||\mathbf{A}||_F = tr(\mathbf{A}^T\mathbf{A}) = tr(\mathbf{A}\mathbf{A}^T)$ designates the Frobenious norm of a matrix.

probe set. Because just one sample $\mathbf{d}_i$ per each training class is given, $\boldsymbol{\Sigma}_x^i = \mathbf{d}_i\mathbf{d}_i^T$ has rank 1, whereas $\boldsymbol{\Sigma}_n^i$ has rank $n-1$. In this particular case, OCA for class i will maximize: $\frac{((\mathbf{b}^i)^T\mathbf{d}_i)^2}{(\mathbf{b}^i)^T\boldsymbol{\Sigma}_n^i\mathbf{b}^i}$. Making derivatives w.r.t. $\mathbf{b}^i$, it can be shown that the optimal OCA basis for class i will be given by $\mathbf{b}^i = (\boldsymbol{\Sigma}_n^i)^{-1}\mathbf{d}_i$. In the testing phase, we will assign a new test image $\mathbf{d}_t$ to the class that maximizes the Raleigh quotient ($\frac{((\mathbf{b}^i)^T\mathbf{d}_t)^2}{(\mathbf{b}^i)^T\boldsymbol{\Sigma}_n^i\mathbf{b}^i}$).

3 Representational OCA

When working with visual data, it is often the case that the training sample size is small compared to the number of pixels (”dimensionality”). Our scenario is an extreme case, since just one sample per training class is available. A classifier like OCA, which is built on such a small sample set, is likely to be biased/unstable, to have a large probability of misclassification, and it is unlikely that it can capture the huge variability due to expression, appearance and illumination changes. There exist several techniques to stabilize weak classifiers (poor performance), such as noise injection [1] or combinationing several classifiers (bagging, boosting, etc) [20]. However, these techniques are not specially useful in our scenario because just one sample per training class is available.

A key observation is the fact that the choice of representation is crucial for the success of a recognition system. For instance, Fig. 3 shows an example of how the same classifier performs very differently, depending on the representation. In the FRGC dataset many factors such as expression, illumination, geometric transformations and out-of-focus images are combined together. In this case, it is certainly difficult to choose the optimal representation that maximizes recognition because there is no explicit model of changes between probe and gallery images. In this section we develop the theory for Representational OCA, which combines OCA classifiers built on different representations to improve the recognition performance.

Given the unique gallery image per class, different representations are built by applying several linear and non-linear filters. In order to mitigate the effects of illumination changes, two algorithms were implemented. The first one uses a factorization method that separates albedo from illuminant and has reported very good performance for face recognition [6]. The second method uses constrast-limited adaptive histogram equalization (CLAHE) [23]. This method enhances the contrast of the image by equalizing small regions (we divide the image into 16 patches). Fig. 4 shows examples of the illumination normalized images.

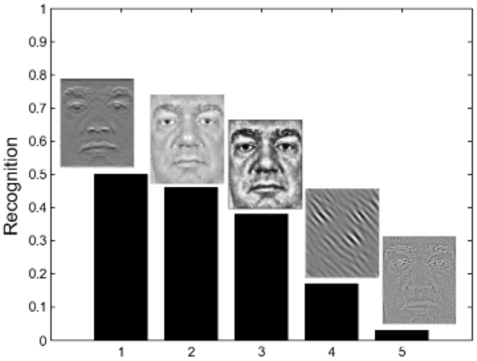

Figure 3: Recognition using OCA classifiers for different representations (Gradient, illumination normalization , local histogram equalization, Gabor, Laplacian).

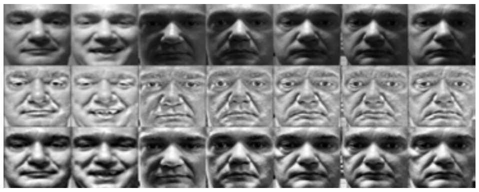

Figure 4: a) Original images b) Albedo estimation [6] c) CLAHE [23].

Several linear and non-linear filters are applied to both of the illumination normalized images. The linear filters are prewitt, sobel, laplacian, Gaussian, box filter, Gabor at 5 scales and 4 orientations, and oriented filter pairs [12] at 3 scales and 6 orientations. The non-linear filters are morphological operators such as erode, dilate, opening and close, anisotropic diffusion and phase congruency [9]. Figure 5 shows several representations of the same image for the CLAHE normalized image. We can observe that many representations are redundant, but they introduce different types of robustness against different types of noises and appearance changes. For each illumination-normalized image, 75 different (but redundant) representations are constructed, so that in total, 150 representations are used to perform face recognition. In Fig. 6 the recognition performance for each of the 150 representations in experiment 4 is shown.

After normalizing each image with respect to the mean and variance, an OCA classifier is constructed for each representation and each person in the gallery. It is important to notice that all the classifiers discard some information; however, the information retained for one usually complements another. The algorithm to combine all the representations is as follows:

- For each representation and each person in the gallery build an OCA classifier using eq. 3 (see below).

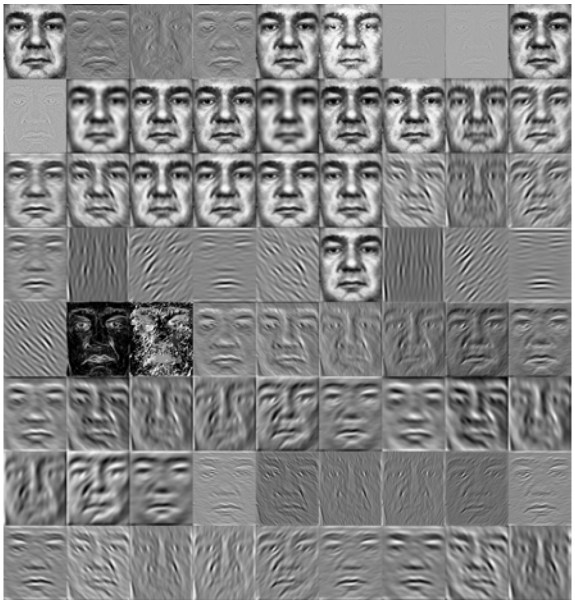

Figure 5: Representations for one training sample.

- For all samples in the testing set, compute the projection into all the classifiers.
- Normalize rows and columns of the response matrix (number of testing samples × number of classes). For each classifier compute the maximum response in the testing set and normalize with it all the responses. The maximum response of any classifier for any of the testing samples should be 1. Later compute the maximum response by rows, that is the maximum for each filter should be 1.
- Order the normalized responses of all the testing samples for each filter. Weight the contribution of the classifier as the inverse of the response of the second biggest score. Total all the classifiers weighted this way and select the maximum.

Observe that the classification error for each classifier in the training set is always 0% for any representation (just 1 sample). However, in order to take into account the discriminability of the classifier, more weight is given to the classifiers that the response to the second biggest score is lower. The straight line (66%) in Fig. 6 indicates the performance of combining all the classifiers (experiment 4). It has improved 13% over the best weak classifier. There are other possible ways of combining classifiers [8], but this one has reported better results in our experiments.

4 Improving Generalization

This section introduces three modifications of the original OCA in order to improve generalization.

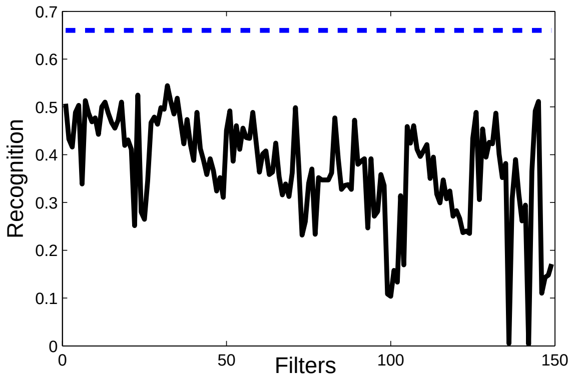

Figure 6: Weak classifiers and its combination.

4.1 Understanding over-fitting

In section 2, it has been shown that the optimal OCA basis for class i will be given by $\mathbf{b}^i = (\mathbf{\Sigma}_n^i)^{-1}\mathbf{d}_i$. However, in general $d >> n$, that is, the number of pixels will be much less than the number of samples. In such cases, the matrix $\mathbf{\Sigma}_n^i$ that should be full rank and strictly positive definite will not be, and undesired directions will be amplified by inverting the matrix (the eigenvectors whose eigenvalues are close to zero are extremely unstable). Observe that in order to be full rank we need at least $d-1$ independent samples, and even in this case it will be a poor estimate of the covariance.

In this scenario, working with huge covariance matrices presents two major problems: the first is computational tractability (storage, efficiency and rank deficiency), and the second has to do with generalization. The most important problem is the lack of generalization when we have few samples. As noticed by Hugues [7], increasing the dimensionality would have to enhance performance for recognition (more information is added), but due to lack of training data this will rarely occur. In these cases, OCA over-fits the data and does not generalize well to new samples.

In order to be able to achieve better generalization and not suffer from storage/computational requirements, we approximate the covariance matrices as the sum of outer products plus a scaled identity matrix $\mathbf{\Sigma}_n \approx \mathbf{U}_n\mathbf{\Lambda}_n\mathbf{U}_n^T + \sigma_n^2\mathbf{I}_d$. $\mathbf{U}_n \in \Re^{d\times k}$, $\mathbf{\Lambda}_n \in \Re^{k\times k}$ is a diagonal matrix. In order to estimate the parameters σ_n^2, $\mathbf{U}_n$, $\mathbf{\Lambda}_n$, a fitting approach is followed by minimizing $E_c(\mathbf{U}_n, \mathbf{\Lambda}_n, \sigma_n^2) = ||\mathbf{\Sigma}_n - \mathbf{U}_n\mathbf{\Lambda}_n\mathbf{U}_n^T - \sigma_n^2\mathbf{I}_d||_F$. The optimal parameters are given by: $\sigma_n^2 = tr(\mathbf{\Sigma}_n - \mathbf{U}_n\hat{\mathbf{\Lambda}}_n\mathbf{U}_n^T)/d - k$, $\mathbf{\Lambda}_i = \hat{\mathbf{\Lambda}}_n - \sigma_n^2\mathbf{I}_d$, where $\hat{\mathbf{\Lambda}}_n$ are the eigenvalues of the covariance matrix $\mathbf{\Sigma}_n$ and $\mathbf{U}_n$ the eigenvectors. See [3] for a detailed derivation. It is worthwhile to point out an important aspect of the previous factorization. The original covariance matrix has $d(d+1)/2$ free parameters, and after the factorization the number of parameters is reduced to $k(2d-k+1)/2$ (assuming orthonormality of $\mathbf{U}_n$), so much less data is needed to estimate these parameters and hence it is not so prone to over-fitting. The same expression could be derived from probabilistic assumptions [14, 21].

Once the covariance has been factorized, it is much easier to understand the behavior of OCA and over-fitting effects. Applying the matrix inversion lemma $((\mathbf{A}^{-1} + \mathbf{V}\mathbf{C}^{-1}\mathbf{V}^T)^{-1} = \mathbf{A} - (\mathbf{A}\mathbf{V}(\mathbf{C} + \mathbf{V}^T\mathbf{A}\mathbf{V})^{-1})\mathbf{V}^T\mathbf{A})$ to $(\mathbf{U}\mathbf{\Lambda}\mathbf{U}^T + \sigma^2\mathbf{I}_d)$ results in: $(\mathbf{U}\mathbf{\Lambda}\mathbf{U}^T + \sigma^2\mathbf{I}_d)^{-1} = \frac{1}{\sigma^2}(\mathbf{I}_d - \frac{1}{\sigma^2}\mathbf{U}(\mathbf{\Lambda}^{-1} + \frac{\mathbf{I}_d}{\sigma^2})^{-1}\mathbf{U}^T)$. Using the previous equality, it is easy to show that the OCA filter is proportional to:

$$\mathbf{b}^i \propto (\mathbf{I}_d - \mathbf{U}_n \begin{pmatrix} \frac{\lambda_1-\sigma_n^2}{\lambda_1} & 0 & 0 \\ 0 & \frac{\lambda_2-\sigma_n^2}{\lambda_2} & 0 \\ \cdots & & \\ 0 & 0 & \frac{\lambda_k-\sigma_n^2}{\lambda_k} \end{pmatrix} \mathbf{U}_n^T)\mathbf{d}_i \quad (2)$$

where recall that λ_i are the eigenvalues of $\mathbf{\Sigma}_n$. Several interesting things are worth pointing out from eq. 2. If $\sigma = 0$, OCA chooses as a projection/filter for class i, $\mathbf{b}^i$, the training sample $\mathbf{d}_i$ projected into the null space of $\mathbf{\Sigma}_n$, that is $(\mathbf{I}_d - \mathbf{U}_n\mathbf{U}_n^T)$. In the testing phase, the filter/projection $\mathbf{b}^i$ will match the part of the signal $\mathbf{d}_i$ which is not in the space generated by $\mathbf{U}_n$. Observe that this type of discriminative model although highly selective can be extremely sensitive to noise. By having few samples in the training set, we take the risk of learning noise features as discriminative ones. If $\sigma_n^2 >> \lambda_i$ (which never will be by construction), $\mathbf{b}^i \approx \mathbf{d}_i$, which makes sense because the noise will not have a particular orientation. Usually, $\lambda_i > \sigma_n^2$, the bigger σ_n^2 the less importance the last eigenvectors will have in rejecting directions ($\mathbf{\Sigma}_n$).

In table 4.1, we compare the recognition performance for experiment 1 (Figure 2) using the full $\mathbf{\Sigma}_n$ and the factorized one. The Opinverse approach computes OCA by inverting $(\mathbf{\Sigma}_n)^{-1}$. Because $\mathbf{\Sigma}_n$ is not full rank Opinverse just inverts $\mathbf{U}_n\mathbf{\Lambda}_n\mathbf{U}_n^T$ with $\mathbf{U}_n$ preserving % of the energy. In the case of preserving all the directions (100%), this will be equivalent to performing the Pseudo-inverse of $\mathbf{\Sigma}_n$ (eliminating the directions with 0 eigenvalues). OCA uses the same approach but takes σ_n^2 into account, and as it can be observed, it makes a difference.

Energy	70%	80 %	85%	90%	100%
Opinverse	0.65	0.75	0.79	0.81	0.84
OCA	0.91	0.90	0.91	0.91	0.91

Table 1: Recognition performance.

4.2 Weighting Subspaces

The basis $\mathbf{U}_n$ in Eq.2 are constructed by weighting all the training samples except $\mathbf{d}_i$ equally. However, it would be beneficial to weight more the samples that are closer to the training sample, i, and to weight less the ones that are far away. In this section, we explore the construction of weighted subspaces for $\mathbf{U}_n$.

A weighted subspace calculates the eigenvectors of a weighted covariance matrix $\mathbf{D}\mathbf{W}_s\mathbf{D}^T \in \Re^{d\times d}$, where $\mathbf{W}_s \in \Re^{n\times n}$ is a diagonal matrix containing the weights for each sample. Computing the eigenvectors of $\mathbf{D}\mathbf{W}_s\mathbf{D}^T\mathbf{B} = \mathbf{B}\mathbf{\Lambda}$ is not efficient in either space or time. Using the fact that the solution of $\mathbf{B}$ can be expressed as $\mathbf{B} = \mathbf{D}\boldsymbol{\alpha}$, it can be shown that the matrix $\mathbf{W}_s\mathbf{D}^T\mathbf{D} \in \Re^{n\times n}$ will have the same eigenvalues as $\mathbf{D}\mathbf{W}_s\mathbf{D}^T$ and the eigenvectors are related by $\mathbf{D}$.

Energy	70%	80 %	85%	90%	100%
OCAw	0.92	0.92	0.93	0.94	0.93

Table 2: OCA by weighted subspaces.

Table 2 shows the recognition results in experiment 1 (see table 4.1) by using weighted subspaces. In order to decide which samples to weight more, the Euclidian distance between the samples is computed in the PCA space. Once the distance between $\mathbf{d}_i$ and the rest of the samples in the gallery is computed, we order the samples, and the ones that are closer to $\mathbf{d}_i$ are weighted more. Each 20 samples we decrease the weighting factor by 0.05 (starting from 1).

4.3 Modeling intra-person variation.

OCA can be very sensitive to the lack of training data, especially when many classes are available. In these situations, OCA selects noisy directions because most of the generative aspect of the sample can be represented by a linear combination of some of the images that not belong to the class. In order to make OCA less sensitive to variations in the training set and to be able to model small misregistrations, a learned subspace is incorporated to constrain the solution.

Two main subspaces are constructed in order to better constrain the directions of the intra-person variation. The first subspace $\mathbf{B}^g$ will model changes in the intra-personal variation owing to such factors as appearance changes, illumination, expression, out-of focus, etc. This subspace will be learned from a different training set than the gallery images. A data matrix $\mathbf{D}^g$ will be constructed by stacking the difference between all the samples in the same class, and $\mathbf{B}^g$ is obtained by SVD $\mathbf{D}^g$.

As registration is one of the most important steps towards improve recognition performance, the second subspace, $\mathbf{B}^m$, will compensate for small misregistrations. In order to achieve that, we will construct $\mathbf{B}^m$ by SVD shifted versions of the sample $\mathbf{d}_i$. Once these subspaces are computed, the aim will be to find $\mathbf{B}$ such that it maximizes:

$$\frac{|\mathbf{B}^T(\mathbf{d}_i\mathbf{d}_i^T + \lambda_1\mathbf{B}^m(\mathbf{B}^m)^T + \lambda_2\mathbf{B}^g(\mathbf{B}^g)^T)\mathbf{B}|}{|\mathbf{B}^T\mathbf{\Sigma}_n^i\mathbf{B}|} \quad (3)$$

where recall $\mathbf{\Sigma}_n^i = \frac{1}{n-1}\mathbf{D}\sqrt{\mathbf{W}}(\mathbf{G}\mathbf{G}^T - \mathbf{g}_i\mathbf{g}_i^T)\sqrt{\mathbf{W}}\mathbf{D}^T$ represents the extra class variation. λ_1 and λ_2 will weight the importance of each subspace.

Solution of eq.(3) involves solving the following $d \times d$ generalized eigenvalue problem $\big(\mathbf{d}_i\mathbf{d}_i^T + \lambda_1\mathbf{B}^m(\mathbf{B}^m)^T + \lambda_2\mathbf{B}^g(\mathbf{B}^g)^T\big)\mathbf{B} = \mathbf{\Sigma}_n^i\mathbf{B}\mathbf{\Lambda}$. In the next section we will propose a stable and efficient (in space and time) generalized eigensolver to deal with this large scale generalized eigenvalue problem.

5 Solving Generalized Eigenvalue Problems for High Dimensional Data

This section gives the details of an efficient and stable subspace iteration algorithm to solve eq. 3. The proposed method achieves filters/basis that have better generalization properties. Let $\mathbf{D}_1 = [\mathbf{d}_i \;\; \sqrt{\lambda_1}\mathbf{D}^m \;\; \sqrt{\lambda_2}\mathbf{D}^g] \in \Re^{d \times n_1}$, where $\mathbf{D}^m \in \Re^{d \times 48}$ is a matrix such that each column contains a vectorized shifted version of $\mathbf{d}_i$ (by at most 3 pixels in x and y) and $\mathbf{D}^g \in \Re^{d \times n_3}$ incorporates training images with the intra-class difference of a different training set (not the gallery). Finally, let $\mathbf{D}_2 = \frac{1}{\sqrt{n-1}}\mathbf{D}\sqrt{\mathbf{W}}(\mathbf{G} - \mathbf{g}_i\mathbf{1}_c^T) \in \Re^{d \times n_2}$, a matrix retaining all the gallery images but i. $\mathbf{A} = \mathbf{D}_1\mathbf{D}_1^T \in \Re^{d \times d}$, $\mathbf{C} = \mathbf{D}_2\mathbf{D}_2^T \in \Re^{d \times d}$ and the maximum rank of the eigensystem is $max(n_1, n_2)$ $(d >> n_1, d >> n_2)$. Recall that $\mathbf{A}$ and $\mathbf{C}$ are $d \times d$ positive definite symmetric matrices (by construction), that are very large and have no general pattern of zeros or specific structure. In this case, traditional efficient methods based on factorization of either matrix (e.g. QZ, QR, etc [5]) would become impractical. Other methods that employ iterative schemes for minimizing the Raleigh quotient $\frac{\mathbf{x}^T\mathbf{A}\mathbf{x}}{\mathbf{x}^T\mathbf{C}\mathbf{x}}$ [15] to obtain the biggest/smallest eigenvalue, rely on deflation procedures in order to obtain several eigenvectors. Such a deflation process often breaks down numerically (especially when increasing number of eigenvectors).

In the subspace iteration method [2], an initial random vector $\mathbf{V}_k \in \Re^{d \times q}$ is first generated. Then for $k = 1, \;\cdots\;$, the following iterations are performed:

$$\mathbf{C}\hat{\mathbf{V}}_{k+1} = \mathbf{A}\mathbf{V}_k \quad \hat{\mathbf{V}}_{k+1} = \hat{\mathbf{V}}_{k+1}/max(\hat{\mathbf{V}}_{k+1}) \quad (4)$$
$$\mathbf{S} = \hat{\mathbf{V}}_{k+1}^T\mathbf{A}\hat{\mathbf{V}}_{k+1} \quad \mathbf{T} = \hat{\mathbf{V}}_{k+1}^T\mathbf{C}\hat{\mathbf{V}}_{k+1}$$
$$\mathbf{S}\mathbf{W} = \mathbf{C}\mathbf{W}\mathbf{\Delta}$$
$$\mathbf{V}_{k+1} = \hat{\mathbf{V}}_{k+1}\mathbf{W}$$

The first step of the algorithm solves a linear system of equations to find $\hat{\mathbf{V}}_{k+1}$. Later, in order to impose the constraints that $\mathbf{V}_{k+1}^T\mathbf{C}\mathbf{V}_{k+1} = \mathbf{\Lambda}$ and $\mathbf{V}_{k+1}^T\mathbf{A}\mathbf{V}_{k+1} = \mathbf{I}_d$, a normalization is done by solving the following $q \times q$ generalized eigenvalue problem, $\mathbf{S}\mathbf{W} = \mathbf{C}\mathbf{W}\mathbf{\Delta}$. It can be shown [2] that as k increases $\mathbf{V}_{k+1}$ will converge to the eigenvectors of $\mathbf{C}\mathbf{V} = \mathbf{A}\mathbf{V}\mathbf{\Lambda}$ and $\mathbf{\Delta}$ to the eigenvalues $\mathbf{\Lambda}$, where $\mathbf{\Delta} = diag(\delta_1, \;\cdots\;\delta_q)$. The convergence is achieved when $\frac{|\delta_i^{k+1} - \delta_i^k|}{\delta_i^{k+1}} < \epsilon \;\forall i$.

At each iteration the computationally expensive part is to solve the linear systems of equations $\mathbf{C}\hat{\mathbf{V}}_{k+1} = \mathbf{A}\mathbf{V}_k$. In our particular application the matrices $\mathbf{C}$ and $\mathbf{A}$ are rank deficient and hence $\mathbf{C}$ is not invertible. In order to achieve numerically stable results, avoid overfitting and improve efficiency, the matrices $\mathbf{A}$ and $\mathbf{C}$ are factorized. That is, $\mathbf{A} = \mathbf{D}_1\mathbf{D}_1^T \approx \mathbf{U}_1\mathbf{\Lambda}_1\mathbf{U}_1^T + \sigma_1^2\mathbf{I}_d$ and $\mathbf{C} = \mathbf{D}_2\mathbf{D}_2^T \approx \mathbf{U}_2\mathbf{\Lambda}_2\mathbf{U}_2^T + \sigma_2^2\mathbf{I}_d$. Once such factorizations are obtained (see [3]), we apply the matrix inversion lemma (2) and $\hat{\mathbf{V}}_{k+1}$ will be:

$$\frac{1}{\sigma_2^2}(\mathbf{I}_d - \frac{1}{\sigma_2^2}\mathbf{U}_2(\mathbf{\Lambda}_2^{-1} + \frac{\mathbf{I}_d}{\sigma_2^2})^{-1}\mathbf{U}_2^T)(\mathbf{U}_1\mathbf{\Lambda}_1(\mathbf{U}_1^T\mathbf{V}_k) + \sigma_1^2\mathbf{V}_k)$$

The reminder of the steps are equivalent to the subspace iteration algorithm. The computational cost (once the factorization is done) is greatly improved.

It is worthwhile to point out that there exist other ways to solve the small sample case. For instance, assuming $n_1 >> n_2$, one could compute the subspace $\mathbf{B}_1 \in \Re^{d \times n_1}$ which diagonalizes $\mathbf{D}_1\mathbf{D}_1^T$ if $n_1 >> n_2$. $\mathbf{B}_1$ will be given by the eigenvectors of $\mathbf{D}_1\mathbf{D}_1^T$ (efficiently computed as the eigenvectos of $\mathbf{D}_1^T\mathbf{D}_1$) with non-zero eigenvalues. Then we project $\mathbf{D}_2$ into the subspace spanned by $\mathbf{B}_1$ and compute the eigenvectors of $\mathbf{B}_1^T\mathbf{D}_2\mathbf{D}_2^T\mathbf{B}_1 \in \Re^{n_1 \times n_1}$ which will be stored in $\mathbf{B}_2$. It is easy to show that the transformation $\mathbf{B}_2\mathbf{B}_1^T$ simultaneously diagonalizes $\mathbf{D}_1\mathbf{D}_1^T$ and $\mathbf{D}_2\mathbf{D}_2^T$. Although a *closed* form solution, this method is likely to overfit the data and if some truncation is done in the first step some discriminatory power could be lost.

6 Experiments

In this section we show results from experiments 1 and 4 of the FRGC v1.0 dataset (http://www.beebiometrics.org/).

In experiment 1, the gallery is composed of 152 images (resize to 150×130 pixels) and the probe has 608 images. All the images are recorded in a controlled indoors environment. Fig. 2 shows some images of the gallery 2.a and probe 2.b. In the probe set, there are small changes due to expression and mild illumination. In this case, we have set up $\lambda_1 = 1$ and $\lambda_2 = 0$, so no extra training images are used. Table 3 shows the recognition performance for several techniques. PCm refers to performing PCA matching using the Mahalanobis distance, the PCA basis is computed using the gallery images and preserving all the eigenvectors, 100% energy (maximum recognition rate). Similarly PCe holds for PCA matching using the Euclidian distance. NN refers to the Nearest Neighbour using the Euclidean distance and NN_2 is a weighted (by the inverse of the variance of each pixel) NN. The first row indicates the type of illumination normalization, No indicates normalizing just by the energy, Il-1 uses [6] and Il-2 [23]. In the first experiment there are no big changes in illumination, but the illumination normalization algorithms improve performance. As it can be observed, ROCA outperforms PCA or NN type of techniques and performs similarly to the commercial system. In this dataset, combining several representations is not particularly useful or interesting, since a single classifier with Il-2 type of normalization can achieve 95.2% of recognition rate. However, it is still interesting to observe that by combining several representations, we are able to improve 1.4% of recognition rate. Most of classification errors are due to changes in expression or out of focus images.

Ilu	PCm	PCe	NN	NN_2	ROCA	FaceIt
No	0.75	0.50	0.51	0.50	0.966	0.965
Il-1	0.88	0.75	0.68	0.75	0.966	0.965
Il-2	0.88	0.75	0.71	0.75	0.966	0.965

Table 3: Recognition comparison for experiment 1.

Experiment 4 is a much more challenging one. The gallery images (152) are taken in a controlled indoor environment, but the probe images (608) are taken in an uncontrolled indoor environment. Figure 7 shows some gallery and probe images. The appearance, expression and illumination changes are strong. Also, several images are out of focus, which makes the recognition problem much harder. For this experiment, $\lambda_1 = 1$ and $\lambda_2 = \frac{10}{n_3}$, where n_3 is the number of images in the training set. In the training set (not the gallery), we used 212 images from the original 366 (the ones with more than 2 samples with the same person) to create the matrix $\mathbf{D}^g$. Table 4 shows the results.

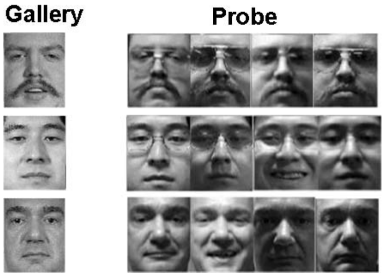

Figure 7: Some images of the gallery and probe.

Ilu	PCm	PCe	NN	NN_2	ROCA	FaceIt
No	0.12	0.15	0.15	0.16	0.755	0.409
Il-1	0.33	0.20	0.27	0.20	0.755	0.409
Il-2	0.23	0.18	0.20	0.20	0.755	0.409

Table 4: Recognition comparison for experiment 4.

In this case PCA/NN perform very poorly because the appearance and expression changes between the gallery and the probe are significant. The commercial system also performs modestly [3]. In this experiment, the power of using several representations that are robust for different type of noises becomes evident. By combining 150 representations ROCA, achieves a 66% recognition rate (14% better than the best representation), Fig. 6. However, some of these representations perform very poorly and they are more a source of confusion than a help. A subset of these filters has been selected with the purpose of trying to combine very different representations. The best set of Gabor filters at 4 orientations (1 scale), the best scale for the oriented filter pairs [12] at 6 orientations, and the phase of the phase congruency [9] are selected (17 filters). In order to weight similarly the contribution of each set of filters, we weighted twice the Gabor responses and 8 times the phase congruency (28 filters). Combining these 28 representations for the two illumination invariant representations (56 classifiers), ROCA achieves 75.5% recognition rate. Fig. 8 illustrates the combination of this set of filters. This representation has achieved 96.4% recognition in experiment 1.

The code is implemented in non optimized Matlab code, ROCA takes 30 hours (Pentium IV- 2Mhz) to compute the results for experiment 4. Several improvements could be done (e.g. rather than computing PCA of $\mathbf{\Sigma}_n^i$ for each filter, it could be computed recursively by adding and subtracting one sample) and we feel ROCA is suitable to process huge datasets as we expect to do in FRGC ver 2.0 dataset.

7 Conclusions

In this paper we have proposed Representational Oriented Component Analysis (ROCA), an extension of

[3] FaceIt classified correctly 200 images and 288 incorrectly. In the remaining 120 images FaceIt was not able to locate the face.

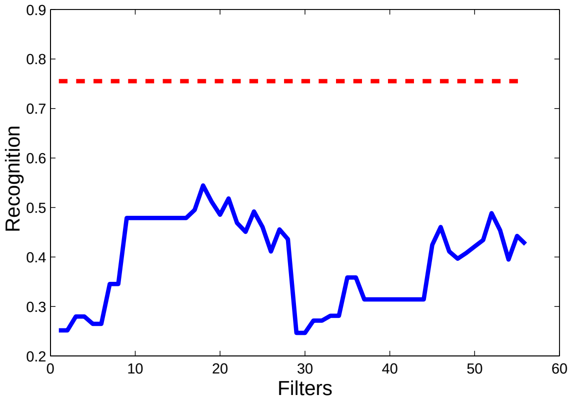

Figure 8: OCA classifiers and its combination.

OCA to perform face recognition when just one sample per training class is available. By combining classifiers with different representations, ROCA is able to improve by 20% the recognition performance over the best individual classifier. Although combining many representations is a promising approach, several questions remain unsolved, such as how to automatically select the best representations or how to optimally combine these classifiers. In particular, we plan to use a cross-validation procedure to choose the set of representations, and analyze the trade-off performance versus the number of representations. On the other hand, several numerical novelties have been introduced to improve generalization, avoid overfitting, and deal with high dimensional data. Finally, it is worthwhile to mention that other vision problems (e.g. appearance tracking or eigen-X problems) can greatly benefit from multiple representations.

8 Acknowledgements

The authors have been partially supported by U.S. Department of Defense contract N41756-03-C4024 and NIMH Grant R01 MH51435. Thanks to the anonymous reviewers for sharp comments that have improved the quality of the paper.

References

[1] G. An. The effects of adding noise during backpropagation training on a generalization performance. *Neural Computation*, 8(3):643–674, 1996.

[2] K. J. Bathe and E. Wilson. *Numerical Methods in Finite Element.* Prentice-Hall. 1976.

[3] F. de la Torre and T. Kanade. Multimodal oriented discriminant analysis. In *tech. report CMU-RI-TR-05-03, Robotics Institute, CMU, January 2005.*

[4] K. I. Diamantaras. *Principal Component Neural Networks (Therory and Applications).* John Wiley & Sons, 1996.

[5] G. Golub and C. F. V. Loan. *Matrix Computations.* 2nd ed. The Johns Hopkins University Press, 1989.

[6] R. Gross and V. Brajovic. An image pre-processing algorithm for illumination invariant face recognition. In *4th International Conference on Audio-and Video Based Biometric Person Authentication (AVBPA)*, pages 10–18, June 2003.

[7] G. Hughes. On the mean accuracy of statistical pattern recognition,. *IEEE Transactions on Information Theory*, 14:55–63, 1968.

[8] J. Kittler, M. Hatef, R. Duin, and J. Matas. On combining classifiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(3), 1998.

[9] P. Kovesi. Image features from phase congruency. *Videre: A Journal of Computer Vision Research.*, 1(3), 1999.

[10] B. V. Kumar. Tutorial survey of composite filter designs for optical correlators. *Applied Optics*, 31:4773–4801, 1992.

[11] N. Malayath, H. Hermansky, A. Kain, and R. Carlson. Speaker-independent feature extraction by oriented principal component analysis. In *EUROSPEECH*, 1997.

[12] J. Malik, S. Belongie, T. Leung, and J. Shi. Contour and texture analysis for image segmentation. *International Journal of Computer Vision*, 2001.

[13] B. Moghaddam, T. Jebara, and A. Pentland. Bayesian face recogntion. *Pattern Recogntion*, 11(33):1771–1782, Nov. 2000.

[14] B. Moghaddam and A. Pentland. Probabilistic visual learning for object representation. *Pattern Analysis and Machine Intelligence*, 19(7):137–143, July 1997.

[15] H. Murase and S. K. Nayar. Visual learning and recognition of 3D objects from appearance. *International Journal of Computer vision*, 1(14):5–24, 1995.

[16] D. North. Analysis of the factors which determine signal/noise discrimination in radar. *Proc. IEEE. (Report PPR-6C. RCA Laboratories, Princeton, N. J. 1943)*, (51):1016–1027, 1963.

[17] E. Oja. *Subspace methods of Pattern Recognition.* Research Studies Press, Hertfordshire, 1983.

[18] J. Proakis. *Digital Communications.* McGraw Hill, 2003.

[19] G. Shakhnarovich and B. Moghaddam. Face recognition in subspaces. In *Handbook of Face Recognition. Eds. Stan Z. Li and Anil K. Jain, Springer-Verlag. Also MERL TR2004-041*, 2004.

[20] M. Skurichina and R. P. W. Duin. Bagging, boosting and the randon subspace method for linear classifiers. *Pattern Analysis and Applications*, (5):121–135, 2002.

[21] M. Tipping and C. M. Bishop. Probabilistic principal component analysis. *Journal of the Royal Statistical Society B*, 61:611–622, 1999.

[22] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld. Face recognition: A literature survey. *ACM computing surveys*, 35(4):399–458, 2003.

[23] J. Zuiderveld. *Contrast Limited Adaptive Histogram Equalization. Graphics Gems IV.* Cambridge, MA, Academic Press, 1994.