SEPARABLE PCA FOR IMAGE CLASSIFICATION

Yongxin Taylor Xi and Peter J. Ramadge

Dept. Electrical Engineering, Princeton University, Princeton NJ

ABSTRACT

As an alternative to standard PCA, matrix-based image dimensionality reduction methods have recently been proposed and have gained attention due to reported computational efficiency and robust performance in classification. We unify all of these methods through one concept: Separable Principle Component Analysis (SPCA). We show that the proposed matrix methods are either equivalent to, special cases of, or approximations to SPCA. We include performance comparisons of the methods on two face data sets and a handwritten digit data set. The empirical results indicate that two existing methods, BD-PCA and its variant NGLRAM, are very good, efficiently computable, approximate solutions to practical SPCA problems.

Index Terms— Image classification, eigenvalues and eigenfunctions, discrete transforms, image representations, face recognition.

1. INTRODUCTION

Principal component analysis (PCA) is an important feature selection method used in many image detection/classification schemes. One prominent example is its successful application in face detection and classification, e.g. [1, 2]. However, estimation of the PCA projection from data has some limitations. First, its computational complexity makes it difficult to deal directly with high dimensional data, e.g. images. Second, the number of examples available for the estimation of the PCA projection is typically much smaller that the ambient dimension of the data and this can lead to over fitting of the projection. In an effort to alleviate these problems in image classification applications, several variations on standard PCA have recently been proposed [3, 4, 5, 6, 7]. These schemes are reported to have reduced computational burden and, when coupled with appropriate classifiers, to yield improved and robust classification rates [3, 4, 8, 5]. We seek to better understand the relationship of these algorithms with standard methods.

Our main contribution is the unification of these methods through *Separable PCA* (SPCA). SPCA seeks a separable basis of images that maximizes the variance of the coordinates over the ensemble of data images. We show that each of the above schemes is either equivalent to, a special case of, or an approximation to SPCA. Specifically, 2DPCA [3] is an easily solvable special case of SPCA. BD-PCA [4] and NGLRAM [7] project the image data onto a separable basis. We give precise conditions under which BD-PCA is a solution of SPCA and when these conditions are not satisfied, show that BD-PCA and NGLRAM give very good approximate solutions to SPCA. Finally, GLRAM [5], a method for obtaining low rank approximations, is equivalent to SPCA. Thus SPCA unifies a variety of prior proposals in the literature.

2. BACKGROUND

Let $\mathcal{X}$ denote a linear space and Y denote a finite set of labels. Given a set $\{(x_k, y_k) \in \mathcal{X} \times Y, k = 1, \ldots, N\}$ of training examples (x_i are instances, y_i are labels), we want to design a classifier $h\colon \mathcal{X} \to Y$ that 'best' predicts the label of a new test instance $x \in \mathcal{X}$. For example, each training instance might be an $m \times n$ grey scale face image with the associated label being the identifier of the corresponding individual.

The PCA approach to this problem uses the training data $\{x_k\}_{k=1}^N$ to determine a linear projection $Q\colon \mathcal{X} \to \mathbb{R}^d$ into a lower dimensional space. Then the label information is used to design a classifier $h\colon \mathbb{R}^d \to Y$. For example, this might be a nearest neighbor classifier in the projected space.

It will be helpful to review PCA when $\mathcal{X} = \mathbb{R}^s$, some integer $s > 0$. Without loss of generality, assume that the data is centered, i.e., $\sum_{k=1}^N x_k = 0$. To select the PCA projection, form the data matrix $D = [x_1, x_2, \ldots, x_N]$. The scatter matrix (empirical covariance) is then $DD^T = \sum_{k=1}^N x_k x_k^T \in \mathbb{R}^{s\times s}$. DD^T has at most $N-1$ nonzero eigenvalues. Let w_j, $j = 1, \ldots, d$, denote the first d eigenvectors ordered by eigenvalue, largest to smallest. The PCA projection into $\mathbb{R}^d$ results by setting $P = [w_1, w_2, \ldots, w_d]$ and $\hat{x}_k = P^T x_k$. In practice, one computes P from an SVD $D = U\Sigma V^T$, yielding $DD^T = U\Sigma^2 U^T$ and $P = [u_1, \ldots, u_d]$ where the u_j are the first d left singular vectors of D. For $N \ll s$, the complexity of computing P is O(sN^2) in time and O(sN) in space.

When each data point is an $m \times n$ grey scale image A_k, PCA finds an ON set $\{W_j\}_{j=1}^d$ of d principal eigenimages of the empirical covariance function of the image data [9]. Image A_k is then projected to its coordinates with respect to this ON basis, i.e., $\hat{a}_{kj} = \langle A_k, W_j\rangle$, $j = 1, \ldots, d$, where $\langle\cdot,\cdot\rangle$ is the standard inner product. It is convenient to compute these eigenimages by exploiting an isometry between $\mathbb{R}^{m\times n}$ and

$\mathbb{R}^{mn}$. Let r_i^T denote the ith row of $A \in \mathbb{R}^{m\times n}$. The bijection $\rho\colon \mathbb{R}^{m\times n} \to \mathbb{R}^{mn}$ that vectorizes $A \in \mathbb{R}^{m\times n}$ rowwise: $\rho(A) = \begin{pmatrix} r_1^T & r_2^T & \dots & r_m^T \end{pmatrix}^T$, preserves inner products and hence angles and Euclidean distances. Thus the PCA decomposition can be computed using the vectorized data and then converted back, by inverting ρ, to image space. With $N \ll mn$, the time cost of this computation is O(mnN^2) and the space required is O(mnN).

The time and space complexity of computing the truncated SVD limits the number and size of training images. Moreover, when the number of examples N is much smaller than the data dimension, the computed PCA projection may suffer from over-fitting (each basis W_j has mn unknowns and is estimated from N examples). We are particularly concerned here with the following proposals for addressing these two issues for $m \times n$ grey scale image data.

In [3] the authors propose a PCA-based scheme, called 2DPCA, which computes an SVD $R = Q_r \Omega Q_r^T$ of

$$R = \sum_{k=1}^{N} A_k^T A_k = \sum_{k=1}^{N}\sum_{i=1}^{m} r_i^k {r_i^k}^T \tag{1}$$

where ${r_i^k}^T$ denotes the i-th row of A_k. R is the scatter matrix of all rows over all training images. The first q columns of Q_r are used to form $V_q \in \mathbb{R}^{n\times q}$ and the data are then projected by right matrix multiplication:

$$\hat{A}_k = A_k V_q \tag{2}$$

Clearly, the same procedure can be applied to A_k^T, $k = 1, \dots, N$. This computes the eigenvectors Q_c of the scatter matrix of all columns of the training images:

$$C = \sum_{k=1}^{N} A_k A_k^T = \sum_{k=1}^{N}\sum_{j=1}^{m} c_j^k {c_j^k}^T \tag{3}$$

where c_j^k denotes the j-th column of A_k. The p principal eigenvectors are then used to form the projection matrix U_p. Combining both projections yields Bidirectional-PCA (BD-PCA) [4, 6]. BD-PCA computes the $m \times q$ projection matrix V_q as above and does the same for the transposed matrices A_i^T to yield an $n \times p$ projection matrix U_p. Then the BD-PCA projection of A_k is the $p \times q$ matrix:

$$\hat{A}_k = U_p^T A_k V_q \tag{4}$$

Comparison of (2) and (4) indicates that 2DPCA is a special case of BD-PCA with $p = m$ and $U_p = I_m$.

BD-PCA is closely related to a third method, GLRAM [5], for generalized low rank approximation of matrices. In GLRAM one directly seeks the best low rank approximations B_k to A_k using a common set of generators: $\min_{B_k,U,V} \sum_{k=1}^{N} \|A_k - U^T B_k V\|_F^2$. The matrices B_k can be thought of as low-rank projections of the data.

2DPCA, BD-PCA and GLRAM are of interest because of reported computational efficiency and robust performance in image classification [3, 4, 8, 5].

3. SEPARABLE PCA

The BD-PCA projection is obtained via a separable orthonormal image transform [9], followed by selecting a subset of the transform coefficients. To see this, let $V = [V_q\ \tilde{V}]$ where the columns of V form an ON basis for $\mathbb{R}^n$. Similarly, let $U = [U_p\ \tilde{U}]$, where the columns of U form an ON basis for $\mathbb{R}^m$. Then the set $W_{i,j} = u_i v_j^T$, $i = 1, \dots, m$, $j = 1, \dots, n$, is a separable ON basis for $\mathbb{R}^{m\times n}$. The matrix of coefficients of the expansion of $A \in \mathbb{R}^{m\times n}$ in this basis is $U^T A V$ [9]. Clearly, the BD-PCA projection is an upper left hand subblock of this matrix, verifying our observation. There are many possible separable transform projections of the above form. Hence it is natural to ask what is the optimal (in the PCA sense) projection onto a separable basis?

Recall that the image PCA projection to d dimensions finds the d principal eigenimages of the empirical covariance function and projects the example images onto these eigenimages. This maximizes the total variance of the projected coefficients. However, in general, the eigenimages are not separable. Adding the requirement of separability places an additional constraint on the projection and helps alleviate the over fitting problem. In PCA one must estimate d, $m \times n$ eigenimages. The extra requirement of separability means we need only estimate p ON vectors in $\mathbb{R}^m$ and q ON vectors in $\mathbb{R}^n$ with $d = pq$. The number of variables to estimate is thus reduced from O($pqmn$) to O($mp + nq$).

3.1. The SPCA Problem

We hence pose the *Separable PCA (SPCA) Problem:*

$$\max_{\substack{U\in\mathbb{R}^{m\times p}\\ V\in\mathbb{R}^{n\times q}}} T(U,V) \quad = \quad \sum_{i=1}^{p}\sum_{j=1}^{q}\sum_{k=1}^{N} \langle A_k, u_i v_j^T\rangle^2 \tag{5}$$

$$\text{subj. to}: U^T U \quad = \quad I_p \quad \& \quad V^T V = I_q \tag{6}$$

Simple algebraic rearrangement of (5) yields two equivalent expressions for the objective:

$$T(U,V) \quad = \quad \text{trace}\left(V^T \left(\sum_{k=1}^{N} A_k^T U U^T A_k \right) V \right) \tag{7}$$

$$\quad = \quad \text{trace}\left(U^T \left(\sum_{k=1}^{N} A_k V V^T A_k^T \right) U \right) \tag{8}$$

(U, V) lies on the product of two Stiefel manifolds. The continuity of T and compactness of the product manifold ensure a maximizing solution (U^*, V^*) exists. (U^*, V^*) is not unique since (7) is also maximized by (U^*Q_1, V^*Q_2) for orthogonal $Q_1 \in \mathbb{R}^{p\times p}$, $Q_2 \in \mathbb{R}^{q\times q}$. At best, the range subspaces $\mathcal{U}^* = \mathcal{R}(U^*)$ and $\mathcal{V}^* = \mathcal{R}(V^*)$ may be unique.

We now use the above formulation to examine 2DPCA, BD-PCA and GLRAM in greater detail.

Theorem 1. *Let $U \in \mathbb{R}^{m\times p}$ and $V \in \mathbb{R}^{n\times q}$ have ON columns and $\mathcal{U} = \mathcal{R}(U)$ and $\mathcal{V} = \mathcal{R}(V)$. Then*

$$T(U,V) \le \min\left\{\sum_{j=1}^{p}\sigma_j(C), \sum_{j=1}^{q}\sigma_j(R)\right\} \qquad (9)$$

with equality if either: the columns of U are the first p eigenvectors of C and for each k, $A_k^T\mathcal{U} \subseteq \mathcal{V}$; or the columns of V are the first q eigenvectors of R and for each k, $A_k\mathcal{V} \subseteq \mathcal{U}$.

Proof. This follows by applying $\mathrm{trace}(AB) = \mathrm{trace}(BA)$, and standard upper bounds to the RHS of (7) and (8). □

Note that the LHS of (9) depends on the data and p, q. Hence if (U, V) achieves equality in (9), then (U, V) is a solution of SPCA for the given p, q. Moreover, (9) suggests that a good rule-of-thumb for adjusting p versus q, is to ensure the two terms on the RHS are as close as possible.

3.1.1. Optimality of BD-PCA, 2DPCA and GLRAM

Corollary 1.1. *Let $\mathcal{U}_p = \mathcal{R}(U_p)$ and $\mathcal{V}_q = \mathcal{R}(V_q)$. If for each k, $A_k\mathcal{V}_q \subseteq \mathcal{U}_p$ or for each k, $A_k^T\mathcal{U}_p \subseteq \mathcal{V}_q$, then BD-PCA solves SPCA.*

Proof. The first requirement for equality in (9) is clearly satisfied. The second is the assumption of the corollary. □

The conditions of the Corollary are readily checkable: compute U_p and V_q; then for every example A_k and column v of V_q, check if $A_k v$ is in the range U_p, etc. This is trivial to check if $p = m$ (resp. $q = n$), since $\mathcal{U}_m = \mathbb{R}^m$ (resp. $\mathcal{V}_n = \mathbb{R}^n$). Hence the following result.

Corollary 1.2. *If $p = m$ or $q = n$, BD-PCA solves SPCA.*

2DPCA is obtained from BD-PCA by the restriction $p = m$ and $U_m = I_m$. Note that (I_m, V_q) satisfies the conditions of Theorem 1 for equality in (9). Thus 2DPCA solves a special case of SPCA.

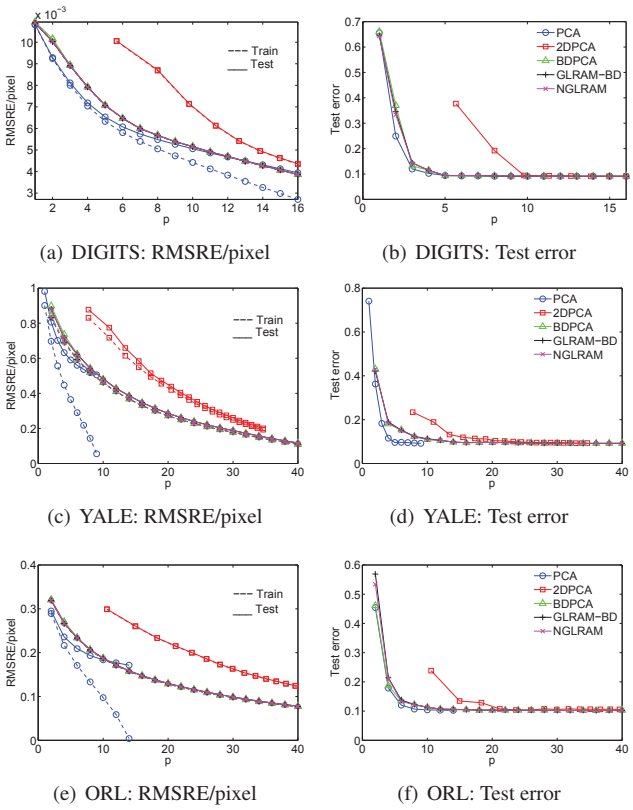
The objective of GLRAM [5] is to obtain $U \in \mathbb{R}^{m\times p}$, $V \in \mathbb{R}^{n\times q}$ and a set of low rank approximations $B_k \in \mathbb{R}^{p\times q}$ to minimize $\sum_k \|A_k - UB_kV^T\|_F^2$. This reduces to solving a problem of the SPCA form [5, Theorem 3.2]. Thus GLRAM is equivalent to separable PCA. This connection to PCA restricted to a separable basis is not explicitly identified in [5]. Moreover, [7] remarks on the desire to "build a close relationship between classical PCA and GLRAM." We believe SPCA gives a very concrete connection.

3.2. Solving SPCA

In general, the solution of SPCA is nontrivial. One approach, the GLRAM algorithm [5], iteratively uses (7) and (8) to update the values of V and U until a local maximum of the objective is attained. Starting from the BD-PCA solution (U_p, V_q) the algorithm can proceed via two paths: first update U, then V and so on; or first update V, then U and so on. A non iterative approximation method, NGLRAM [7], takes one step along each path from (U_p, V_q), and selects the result with the greatest value of T. Clearly BD-PCA is faster than NGLRAM which is faster than GLRAM. Correspondingly, starting from (U_p, V_q), $T_{\mathrm{BD-PCA}} \le T_{\mathrm{NGLRAM}} \le T_{GLRAM}$.

4. EXPERIMENTS

We use the Handwritten Digits database (UCI repository) and the YALE and ORL face databases to experimentally compare the performance of PCA, SPCA (GLRAM), 2DPCA, BD-PCA and NGLRAM. The Digit images (5,620 binary, 32×32 images of the digits 0 through 9) were used without preprocessing. The YALE images (15 subjects, 11 images per subject, 320×243 image size) were first centered, histogram normalized and resized to 60×50 pixels. The ORL images (40 subjects, 10 images per subject, 112×92 image size) were used without preprocessing.



(a) DIGITS: RMSRE/pixel (b) DIGITS: Test error

(c) YALE: RMSRE/pixel (d) YALE: Test error

(e) ORL: RMSRE/pixel (f) ORL: Test error

Fig. 1. Results.

We repeated each experiment 20 times with random selection of the training set (100 per digit for Digits, 6 per person for YALE, 5 per person for ORL). We tested on the remaining data and averaged the results of the 20 runs. For each

method we use a Euclidean nearest-neighbor classifier. In each experiment we record the training time, the root mean square reconstruction error (RMSRE) [5] and the test image misclassification rate. For SPCA and its special cases, except 2DPCA, we set $p = q$ so the projected dimension is $d = p^2$. The results are plotted versus p. SPCA is computed using the GLRAM algorithm (stoping criterion: $\Delta\text{RMSRE} < 10^{-6}$). We tested random (GLRAM-RAN) and the BD-PCA solution (GLRAM-BD) as the initial condition. PCA is computed using the same dimension as SPCA up to the maximum value $p_{\max}^{\text{PCA}} = \lfloor\sqrt{N-1}\rfloor$. For 2DPCA we use projection dimensions $m \times 1, m \times 2, \ldots$, where m is the image height. We plot 2DPCA performance versus $p = \sqrt{m}, \sqrt{2m}, \ldots$. The following table summarizes key parameters. The examples are listed in order of increasing value of mn/N.

Data Set	$m \times n$	N	mn/N	$p_{\max}^{\text{PCA}}$
Digits	32×32	1000	1.028	31
Yale	60×50	90	33.3	9
ORL	112×92	200	206	14

Fig. 1 summarizes the main results. We begin with some general observations. First, the GLRAM algorithm was robust to variations in initial conditions; the RMSRE values of both training and test images disagreed among initial conditions by at most 10^{-4}. Hence only plots for SPCA computed via GLRAM-BD are shown in the figure. Second, both BD-PCA and NGLRAM provided very good, easy-to-compute, approximate SPCA solutions. In each case, the approximate solutions were almost indistinguishable from the GLRAM-BD solution. Third, 2D-PCA was clearly majorized in both reconstruction error and classification error by all other algorithms. Hence we will not comment on it further. Now some more specific observations. The SPCA projection computed via BD-PCA was faster to compute and required less memory than PCA. For Digits, full dimensional training required 8.66 ± 1.02 seconds for PCA and $(1.1 \pm 0.072) \times 10^{-3}$ seconds for BD-PCA. Similarly, for Yale: $(9.0 \pm 0.13) \times 10^{-2}$ (PCA), $(3.7 \pm 0.12) \times 10^{-3}$ (BD-PCA); and for ORL: 1.1 ± 0.003 (PCA), $(1.71 \pm 0.03) \times 10^{-2}$ (BD-PCA). The left column plots show the reconstruction error for training and test images. As expected, SPCA was more resistant to over fitting. This accords with previous reports [3, 4, 5, 6]. However, as shown in the right column plots, the reduced over fitting of SPCA did not improve classification performance over that of PCA. To explore this further, we ran an experiment with only one training image per class. Using YALE, the *best* test error rates were: 0.30 ± 0.04 (PCA), 0.21 ± 0.03 (SPCA); a slight improvement at the cost of requiring a higher projection dimension. But for ORL the results remained indistinguishable (*best* test error: 0.14 ± 0.03 (PCA), 0.13 ± 0.02 (SPCA)).

5. CONCLUSION

SPCA unifies recently proposed matrix-based dimension reduction methods. We have shown empirically, that fast algorithms such as BD-PCA and NGLRAM give very good approximate SPCA solutions and that the robustness to over fitting of SPCA is an advantage in image approximation applications. Our results also suggested how to select the projection dimensions p and q, subject to other constraints, for best results. For $m \approx n$, the memory allocation is $O(Nm^2)$ for PCA and $O(m^2)$ for BD-PCA; potentially orders of magnitude difference. Hence SPCA may be useful as a substitute or precursor to PCA for high dimensional problems or in resource constrained applications. However, SPCA did not improve classification performance over PCA in the examples studied. This suggests that, at least for these data sets, the nearest neighbor classifier is robust to the projection errors caused by PCA over fitting. A major open question is whether there are data sets in which SPCA does improve classification performance. There are also several open technical questions such as uniqueness of the SPCA solution and finding an efficient computation method. We believe the SPCA framework provides the context to address these questions.

6. REFERENCES

[1] M. Turk and A. Pentland, "Eigenfaces for recognition," *J. Cog. Neuroscience*, vol. 3, no. 1, pp. 71–86, 1991.

[2] P.N. Belhumeur, J.P. Hespanha, and D.J. Kriegman, "Eigenfaces vs. fisherfaces: recognition using class specific linear projection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 711–720, Jul 1997.

[3] J. Yang, D. Zhang, A. F. Frangi, and J.Y. Yang, "Two-dimensional pca: A new approach to appearance-based face representation and recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 1, pp. 131–137, 2004.

[4] W.M. Zuo, K.Q. Wang, and D. Zhang, "Bi-directional pca with assembled matrix distance metric," *IEEE Int. Conf. on Image Proc.*, vol. II, pp. 958–961, 2005.

[5] J.P. Ye, "Generalized low rank approximations of matrices," *Machine Learning*, vol. 61, pp. 167–191, 2005.

[6] W.M. Zuo, D. Zhang, and K.Q. Wang, "Bidirectional pca with assembled matrix distance metric for image recognition," *IEEE Trans. on Sys. Man and Cybernetics*, vol. 36, no. 4, pp. 863–872, 2006.

[7] Z.Z. Liang, D. Zhang, and P.F. Shi, "The theoretical analysis of glram and its applications," *Pattern Recogn.*, vol. 40, no. 3, pp. 1032–1041, 2007.

[8] M. Visani, C. Garcia, and C. Laurent, "Comparing robustness of two dimensional pca and eigenfaces for face recognition," *Lect. Notes Comp. Sci.*, vol. 3212, pp. 717–724, 2004.

[9] A. K. Jain, *Fundamentals of Digital Image Processing*, Prentice-Hall, Inc., NJ, USA, 1989.