FAST SIGNAL ANALYSIS AND DECOMPOSITION ON GRAPHS USING THE SPARSE MATRIX TRANSFORM

Leonardo R. Bachega , Guangzhi Cao, and Charles A. Bouman

Purdue University, School of Electrical and Computer Engineering
West Lafayette, IN, 47907-2035, USA
{lbachega, gcao, bouman}@purdue.edu

ABSTRACT

Recently, the Sparse Matrix Transform (SMT) has been proposed as a tool for estimating the eigen-decomposition of high dimensional data vectors [1]. The SMT approach has two major advantages: First it can improve the accuracy of the eigen-decomposition, particularly when the number of observations, n, is less the the vector dimension, p. Second, the resulting SMT eigen-decomposition is very fast to apply, i.e. $O(p)$.

In this paper, we present an SMT eigen-decomposition method suited for application to signals that live on graphs. This new SMT eigen-decomposition method has two major advantages over the more generic method presented in [1]. First, the resulting SMT can be more accurately estimated due to the graphical constraint. Second, the computation required to design the SMT from training data is dramatically reduced from an average observed complexity of p^3 to $p \log p$.

Index Terms— covariance estimation, eigen-images, eigen-faces, sparse matrix transform, Givens rotations.

1. INTRODUCTION

Decorrelation and analysis of high dimensional signals are of great importance in a wide variety of applications [2]. For example, whitening filters and block transforms such as the DCT are widely used to approximately decorrelate stationary signals in time and space for applications such as image and audio source coding. If the signal being processed is not stationary, then techniques such as eigen-image and eigen-signal analysis can be used to decompose high-dimensional signals into approximately decorrelated components for applications such as anomaly detection [3] or face recognition [4].

However, one disadvantage of eigen-signal analysis or equivalently Karhunen-Loéve decomposition methods is that they require a knowledge of the high dimensional signal's covariance. Since a signal of dimension p has an associated covariance matrix of dimension p^2, the amount of data required to estimate this covariance tends to grow as p grows. More specifically, in order to form a good unbiased estimate of the covariance, one needs the number of observed vectors, n, to be substantially larger then their dimension, p [2]. In practice, one often does not have this luxury.

Recently, methods have been proposed which allow high dimensional eigen-signal analysis even when the number of observations is much less than the dimension of the signal. These approaches estimate the eigen-decomposition (and associated covariance) by imposing some type of regularizing constraint [5, 6, 7, 8].

In particular, the method of [1] estimates the eigen-decomposition of a high dimensional signal by assuming that the eigen-transformation can be represented as a sparse matrix transform (SMT). The SMT is formed by a finite product of Givens rotations, so it decomposes the eigen-decomposition into a product of very sparse transformations.

The SMT eigen-decomposition assumption has two major advantages. First, the approach can improve the accuracy of the estimated transform for a fixed quantity of data [9]. Second, the eigen-decomposition then has the form of an SMT, which is very fast to apply, i.e. is $O(p)$. However, one limitation of this approach is that it requires $O(p^3)$ to design the SMT from the observed training data. While applying the SMT is always fast, designing it can therefore be burdensome when p is very large.

In this paper, we present an SMT eigen-decomposition method suited for application to signals that live on graphs, i.e. signals y_i where $i \in \{1, \cdots, p\}$ indexes nodes in a graph. This SMT eigen-decomposition method has two major advantages over the more generic method presented in [1]. First, the resulting SMT can be more accurately estimated due to the graphical constraint. Second, the computation required to design the SMT from training data is dramatically reduced from an average observed complexity of p^3 to $p \log p$.

In practice, many forms of data have a natural graphical structure which can be exploited to make the SMT design fast. For example, in images we can assume that neighboring pixels are connected in a graph structure to make SMT design practical for large images. We show that the resulting sparse transform can be used for eigen-image analysis in applications such as face recognition, and demonstrate that it results in a more accurate fit (as measured via the cross-validated log-likelihood) for real face image data.

2. SMT ESTIMATION AND DESIGN

The SMT design consists of estimating the full set of eigen-vectors and associated eigenvalues for a general p-dimensional

signal. More specifically, the objective is to estimate the orthonormal matrix E and diagonal matrix Λ such that the signal covariance can be decomposed as $R = E\Lambda E^t$, and to compute this estimate from n independent training vectors, $Y = [y_1, \cdots, y_n]$. This is done by assuming the samples are i.i.d. Gaussian random vectors and computing the constrained maximum log-likelihood (ML) estimates of E and Λ. In [1], we show that these constrained ML estimates are given by

$$\hat{E} = \arg\min_{E \in \Omega_K} \left\{ |\mathrm{diag}(E^t S E)| \right\} \quad (1)$$

$$\hat{\Lambda} = \mathrm{diag}(\hat{E}^t S \hat{E}), \quad (2)$$

where $S = \frac{1}{n} Y Y^t$ is the sample covariance matrix, and Ω_K is the set of allowed orthonormal transforms.

If $n > p$ and Ω_K is the set of all orthonormal transforms, then the solution to (1) and (2) is the diagonalization of the sample covariance, i.e. $\hat{E}\hat{\Lambda}\hat{E}^t = S$. However, the sample covariance is a poor estimate of the covariance when $n < p$.

In order to improve the accuracy of the covariance estimate, we will impose the constraint that Ω_K be the set of sparse matrix transforms (SMT) of order K. More specifically, we will assume that the eigen-transformation has the form

$$E = \prod_{k=1}^{K} E_k = E_1 \cdots E_K, \quad (3)$$

where each E_k is an orthonormal transform known as a Givens rotation, and K is the model order parameter. Each Givens rotation operates on just two coordinates, (i_k, j_k), so

$$E_k = I + \Theta(i_k, j_k, \theta_k), \text{ where}$$

$$[\Theta]_{ij} = \begin{cases} \cos(\theta_k) - 1 & \text{if } i = j = i_k \text{ or } i = j = j_k \\ \sin(\theta_k) & \text{if } i = i_k \text{ and } j = j_k \\ -\sin(\theta_k) & \text{if } i = j_k \text{ and } j = i_k \\ 0 & \text{otherwise} \end{cases}.$$

$$(4)$$

Figure 1(a) illustrates the structure of the SMT. Intuitively, each Givens rotation, E_k, plays the same role as the butterflies of a fast Fourier transform (FFT). In fact, the SMT is a generalization of both the FFT and the orthonormal wavelet transform. However, since both the ordering of the coordinate pairs, (i_k, j_k), and the values of the rotation angles, θ_k, are unconstrained, the SMT can model a much wider range of transformations. It is often useful to express the order of the SMT as $K = rp$, where r is the average number of rotations per coordinate. Typically r is small (< 5), so that the computation to apply the SMT to a vector of data is very low, i.e. $2r + 1$ floating-point operations per coordinate.

The optimization of (1) is non-convex, so we use a greedy optimization approach in which we select each rotation, E_k, in sequence to minimize the cost. The greedy SMT design approach leads to the following very intuitive algorithm.

Initialize $S \leftarrow \frac{1}{n} Y Y^t$ and $\hat{E} \leftarrow I$.
For $k = 1$ to K,

1. Search for the two most correlated coordinates,

$$(i_k, j_k) \leftarrow \arg\max_{(i,j)} \left(\frac{S_{ij}^2}{S_{ii} S_{jj}} \right). \quad (5)$$

2. Compute the Givens rotation, E_k, that decorrelates the coordinates with the rotation angle

$$\theta_k = \frac{1}{2} \tan^{-1} \left(-2 S_{i_k j_k}, S_{i_k i_k} - S_{j_k j_k} \right). \quad (6)$$

3. Perform the updates

$$S \leftarrow E_k^t S E_k \quad (7)$$
$$\hat{E} \leftarrow \hat{E} \cdot E_k \quad (8)$$

The computation of the SMT design algorithm is dominated by the time required to search for the most correlated coordinate pairs in (5). This search requires that the correlation between all $p(p-1)/2$ combinations of coordinates be checked. Assuming that r and n are constant, this means the the SMT design algorithm requires $O(p^3)$ operations, and can be computationally expensive when p is large.

3. FAST SMT DESIGN FOR GRAPHICAL DATA

The general SMT design algorithm of Section 2 does not constrain the search for the coordinates (i_k, j_k). However, in many applications, such as social networks and images, the signal data have a natural graphical structure in which neighboring coordinates (i.e, those connected by an edge) are the ones expected to be correlated. We constrain the search for the pair (i_k, j_k) described above to these neighboring coordinates. This constraint to neighboring coordinates has two potential advantages: it can both reduce the computation of the SMT design and also improve the estimate of E.

Figure 1(b) illustrates this approach to SMT design for graphical data. For each value of k, the figure shows that the coordinates have a graph structure, with each node i pointing to a set of nodes ∂i.[1] Then Givens rotations are constrained to be between nodes (i, j) such that either $i \in \partial j$ or $j \in \partial i$. Of course, once a rotation is applied to a pair of coordinates, the neighborhood relations between nodes of the graph must be updated while maintaining the constraint that the maximum fan-out of the graph is bounded by $|\partial i| < M$, where generally $M << p$.[2]

Using this graphical constraint, we have the following algorithm for greedy SMT design.

For all i, initialize the list of neighbors ∂i, and for $\forall j \in \partial i$ compute the correlation C_{ij}.
For $k = 1$ to K,

---

[1] Here we do not assume that the neighborhood relationship is symmetric, so this is a directed graph.

[2] However, the fan-in may exceed this bound, i.e. the set $\{j : i \in \partial j\}$ may be large for some nodes in the graph.
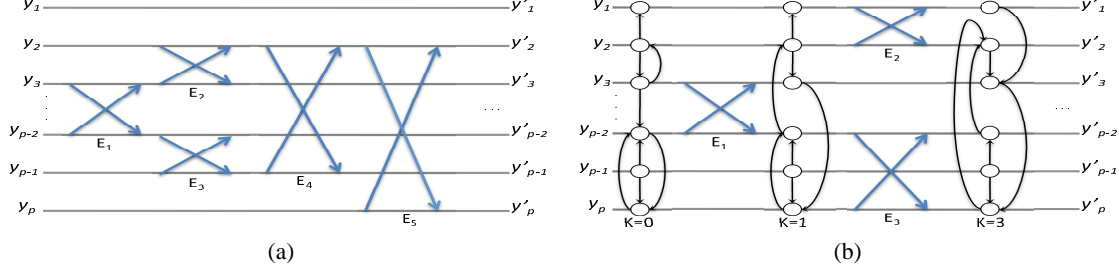
Fig. 1. Diagram of the (a) SMT transform (b) SMT transform with graphical structure.

1. Search for the two most correlated neighboring coordinates,

$$(i_k, j_k) \leftarrow \arg \max_{1 \leq i \leq p} \left(\max_{j \in \partial i} C_{i,j} \right) . \qquad (9)$$

2. Compute the Givens rotation, E_k, that decorrelates the coordinates with the rotation angle

$$\theta_k = \frac{1}{2} \tan^{-1} \left(-2 \sum_{l=1}^{n} Y_{i_k l} Y_{j_k l}, \sum_{l=1}^{n} Y_{i_k l}^2 - \sum_{l=1}^{n} Y_{j_k l}^2 \right) . \qquad (10)$$

3. Merge the neighborhoods ∂i and ∂j

$$\partial i_k \leftarrow \partial i_k \cup \partial j_k \qquad (11)$$
$$\partial j_k \leftarrow \partial i_k \cup \partial j_k . \qquad (12)$$

4. Prune the neighborhoods such that

$$\partial i_k \leftarrow \left\{ \begin{array}{c} M \text{ most correlated coordinates} \\ \text{between } i_k \text{ and } j \in \partial i_k \end{array} \right\} \qquad (13)$$
$$\partial j_k \leftarrow \left\{ \begin{array}{c} M \text{ most correlated coordinates} \\ \text{between } j_k \text{ and } i \in \partial j_k \end{array} \right\} . \qquad (14)$$

5. Perform the updates

$$Y \leftarrow E_k^t \cdot Y \qquad (15)$$
$$\hat{E} \leftarrow \hat{E} \cdot E_k . \qquad (16)$$

6. Update any correlations, $C_{i,j}$ that can be affected by the rotation on coordinates (i_k, j_k).

The computation of this algorithm is now dominated by the search of step 1, and the correlation update of step 6. A naïve implementation, results in an SMT design algorithm with complexity $O(p^2)$ when r, n, and M are fixed. This is because the design of each rotation, E_k, requires a search over all edges in the graph, which is $O(Mp)$.

However, a careful implementation with a red-black search tree can make the search of step 1 order $\log p$ average complexity. In this case, step 6 only requires the update of any coordinates which are either in the fan-out of the nodes $\{i_k, j_k\}$ or in the fan-in of these nodes. In practice, the sum
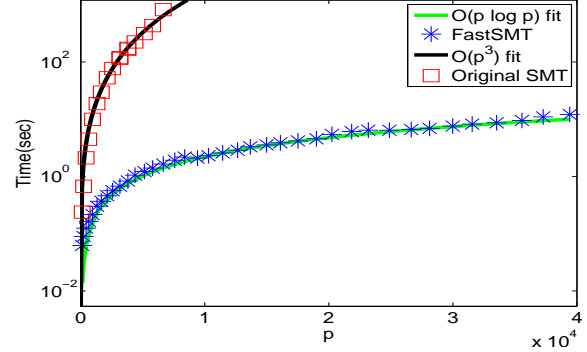

Fig. 2. Execution time of both the original and the fast algorithms as the dimensionality increases, and $r = 1.0$, $M = 8$, $n = 40$ remain fixed.

of the fan-out and fan-in to a coordinate is, on average, of order M. However, in a worst-case scenario the fan-in to any node of the graph is only bounded by p.

So in summary, when r, n, and M are fixed, the empirically observed complexity of the graph based SMT design is experimentally measured to be $p \log p$. However, the theoretical worse-case complexity is $O(p^2)$.

4. EXPERIMENTAL RESULTS

Figure 2 compares the computation of the fast SMT design for graphical data, with the one of the original SMT. Notice that the running time for the fast SMT design requires dramatically less computation than the original method as p becomes large, and that the two algorithms fit the proposed complexity of p^3 and $p \log p$ quite closely over a large range of p.

We applied the fast SMT algorithm to a face image dataset from 40 different subjects from the *ORL Face Database* [10], with the images re-scaled to 28×23 pixels ($p = 644$). Table 1 shows the results of the cross-validated average log-likelihood values of the face dataset split in 10 ways(10 fold cross-validation) for both SMT algorithms and the PCA+Shrinkage method. It also shows the difference of the SMT log-likelihood values from the value obtained using PCA+Shrinkage, and the number of SMT rotations needed to obtain the stated log-likelihood value. As the results suggest, the fast SMT method produces the highest average log-likelihood value.

method	log-likelihood	Δ	r_{max} (K_{max})
PCA+Shrinkage	-2885.71	0	-
SMT	-2805.81	79.88	1.41(910)
fast SMT($M = 8$)	-2793.33	92.37	1.57(1010)
fast SMT($M = 32$)	-2802.36	83.34	1.41(910)
diagonal	-3213.10	-327.40	-

Table 1. Comparison of the maximum expected cross-validation log-likelihood values for all methods studied using face images (with p=644). Notice that SMT with $M = 8$ has the largest log-likelihood, which is 92.37 greater than the value produced by the PCA+Shrinkage method.

Figure 3 shows the results of the SMT eigen-decomposition applied to a small subset of 20 face images from the ORL database. Figure 3(a) shows examples of the images used in this experiment. Figure 3(b) and Fig. 3(c) show the results of the estimated eigenvectors respectively estimated with the SMT and the PCA-based eigen-decompositions. Notice that the SMT decomposition produces a full set of eigen-faces as opposed to only 20 eigen-faces produced by PCA. Unlike PCA, the SMT eigen-images capture local spatial structure of the faces resembling their anatomical parts. We believe such a structure may yield to higher discriminative power than the PCA-based decomposition.

5. CONCLUSIONS

We have introduced a fast algorithm for the design of SMT analysis transformations on graphical data. This approach has three major advantages: 1) It results in a more accurate estimate of the decorrelating transformation for some typical data cases, particularly when $n < p$; 2) The resulting decorrelating SMT transformation is computationally very efficient to implement, i.e. it requires only $2rp$ floating-point operations for a p dimensional vector; 3) When the observed data has a graphical structure, then the SMT design algorithm can be practically implemented with $p \log p$ computation.

6. REFERENCES

[1] G. Cao and C. A. Bouman, "Covariance estimation for high dimensional data vectors using the sparse matrix transform," in *Advances in Neural Information Processing Systems*. MIT Press, 2008.

[2] D. L. Donoho, "High-dimensional data analysis: The curses and blessings of dimensionality," in *Math Challenges of the 21st Century*. Los Angeles: American Mathematical Society, August 8 2000.

[3] J. Theiler, "Subpixel anomalous change detection in remote sensing imagery." Santa Fe, NM: 2008 IEEE Southwest Symposium on Image Analysis and Interpretation, March 24-26 2008, pp. 165–168.
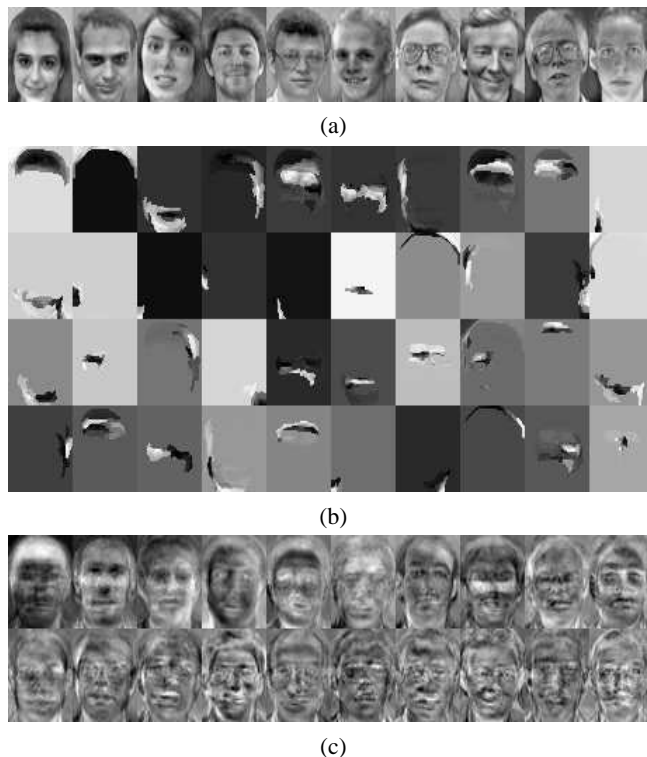
(a)

(b)

(c)

Fig. 3. Experimental results for eigen-faces: (a) Examples of faces in the dataset used in our experiments; (b) The first 40 eigen-faces from the SMT decomposition; (c) Eigen-faces from PCA decomposition.

[4] B. Moghaddam, T. Jebara, and A. Pentland, "Bayesian face recognition," *Pattern Recognition*, vol. 33, pp. 1771–1782, 2000.

[5] M. J. Daniels and R. E. Kass, "Shrinkage estimators for covariance matrices," *Biometrics*, vol. 57, no. 4, pp. 1173–1184, 2001.

[6] J. Friedman, T. Hastie, and R. Tibshirani, "Sparse inverse covariance estimation with the graphical lasso," *Biostatistics*, vol. 9, no. 3, pp. 432–441, Jul. 2008.

[7] C. Chennubhotla and A. Jepson, "Sparse PCA: Extracting multi-scale structure from data," *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, vol. 1, pp. 641–647 vol.1, 2001.

[8] A. B. Lee, B. Nadler, and L. Wasserman, "Treelets-an adaptive multi-scale basis for sparse unordered data," *Annals of Applied Statistics*, vol. 2, no. 2, pp. 435–471, 2008.

[9] G. Cao, C. A. Bouman, and J. Theiler, "Weak signal detection in hyperspectral imagery using sparse matrix transformation (SMT) covariance estimation." First Workshop on Hyperspectral Image and Signal Processing, 2009.

[10] "The orl face database," http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html.