Sequential Principal Component Analysis -An Optimal and Hardware-Implementable Transform for Image Compression

Tuan A. Duong and Vu A. Duong
Jet Propulsion Laboratory,
California Institute of Technology
Pasadena, CA 91109

Abstract—
This paper presents the JPL-developed Sequential Principal Component Analysis (SPCA) algorithm for feature extraction / image compression, based on "dominant-term selection" unsupervised learning technique that requires an order-of-magnitude lesser computation and has simpler architecture compared to the state of the art gradient-descent techniques. This algorithm is inherently amenable to a compact, low power and high speed VLSI hardware embodiment. The paper compares the lossless image compression performance of the JPL's SPCA algorithm with the state of the art JPEG2000, widely used due to its simplified hardware implementability. JPEG2000 is not an optimal data compression technique because of its fixed transform characteristics, regardless of its data structure. On the other hand, conventional Principal Component Analysis based transform (PCA-transform) is a data-dependent-structure transform. However, it is not easy to implement the PCA in compact VLSI hardware, due to its highly computational and architectural complexity. In contrast, the JPL's "dominant-term selection" SPCA algorithm allows, for the first time, a compact, low-power hardware implementation of the powerful PCA algorithm. This paper presents a direct comparison of the JPL's SPCA versus JPEG2000, incorporating the Huffman and arithmetic coding for completeness of the data compression operation. The simulation results show that JPL's SPCA algorithm is superior as an optimal data-dependent-transform over the state of the art JPEG2000. When implemented in hardware, this technique is projected to be ideally suited to future NASA missions for autonomous on-board image data processing to improve the bandwidth of communication.

I. Introduction

For image compression, techniques based upon the Discrete Cosine Transform (DCT) [1], Discrete Wavelet Transform (DWT) as JPEG2000 based transform[2], and Karhunen-Loeve Transform (KLT) (or PCA)[3] are among popular transforms that provide effective coefficients set based on orthogonal bases that can span the full data set for reconstruction. Only PCA is a *data-dependent-transform* technique [4-5] that is used to find *the most representable basis of the given data (not fixed transform), and is* based on the second-order statistics. This implies that PCA is also *the most optimal linear transform* compared with DCT and DWT as concluded by the inventor of the wavelet transform, Daubechies [6].

Principal Component Analysis (PCA), as a feature based approach [7-8] to data-dependent transform for data compression, is quite effective; however, its computational complexity $O(N^3)$ (N^2 is the size of the covariance matrix) [3] remains a challenge in implementing the PCA in software, and its architectural complexity is the inhibiting factor in implementing it in a dedicated hardware.

A better alternative, the Sequential PCA has appeared recently with promising results [8-15]. The computational complexity in the sequential PCA still holds it back from becoming effective when implemented in software, however, architecturally, the sequential PCA-based gradient descent learning technique provides a simpler and

implementable hardware solution. However, it suffers in learning convergence for consecutive principal vectors when previous principal components are extracted and removed from the energy function [16-17]. The left energy function that is continuously used to extract the next component is reduced, from which the learning based on gradient search will require longer time duration to settle in the shallow energy attractor as compared with the previous ones. This obstacle discourages a hardware implementation for this technique.
In this study, we have focused only between the state of the art-JPEG2000 and PCA transform and find the optimal transform between two of them.

II. Technical approach

Mathematical formulation

At JPL, we have developed an optimal PCA learning technique [7-8] which serves two purposes: 1) simplified hardware implementation, specially in VLSI as a System-On-A-Chip approach; 2) fast and reliable convergence due to simplifying computation technique namely Dominant Component Gradient and Dynamic Energy Learning Rate (DOGDYN) as compared with counterpart gradient descent. Furthermore, our new technique requires much less computation and has optimized architecture to be more suitable for hardware implementation as a real time adaptive learning system.
We adapt the objective function [12] below:

$$J(w) = \sum_{i=1}^{m} J_i(w_i) = \sum_{i=1}^{m} \sum_{t=1}^{k} \left| x_t - w_i w_i^T x_t \right|^2 \qquad (1)$$

where m is the number of principal components and k is the number of measurement vectors. x_t is a measured vector at time t and w_i is the ith principal vector (or eigen vector).
With

$$J_i(w_i) = \sum_{t=1}^{k} \left| y_i^t - w_i w_i^T y_i^t \right|^2$$

And $y_i^t = x_t - \sum_{j=1}^{i-1} w_j w_j^T x_t$ (2)

Based on $J_i(w_i)$, w_i is extracted as a principal component vector when $J_i(w_i)$ is minimized.

a) PCA learning technique

From equation (2), the learning algorithm can be processed sequentially for each principal vector that is based on the gradient descent as follows:

$$\Delta w_{ij} = -\frac{\partial J_i}{\partial w_{ij}} = -\frac{\partial(\left| y_i^t - w_i w_i^T y_i^t \right|^2)}{\partial w_{ij}} \qquad (3)$$

From equation (3), only the dominant term [7] is used; the weight update can be obtained as follows:

$$w_{ij}^{new} = w_{ij}^{old} + \zeta \Delta w_{ij} = w_{ij}^{old} + \zeta \varepsilon_{ij} (w_i^T y_i^t + w_{ij} y_{ij}^t) \quad (4)$$

Where $\zeta = \dfrac{E_0}{E_{i-1}}$ and $\hat{y}_i^t = w_i w_i^T y_i^t$

E_0 is the initial energy when the network starts learning and E_{i-1} is the energy of the *(i-1)*th extracted principal component. This learning approach is amenable to a simple hardware architecture by cascading the cell learning building block.

b) Cell learning architecture

From equation (4), the learning architecture is realized as in Figure 1.

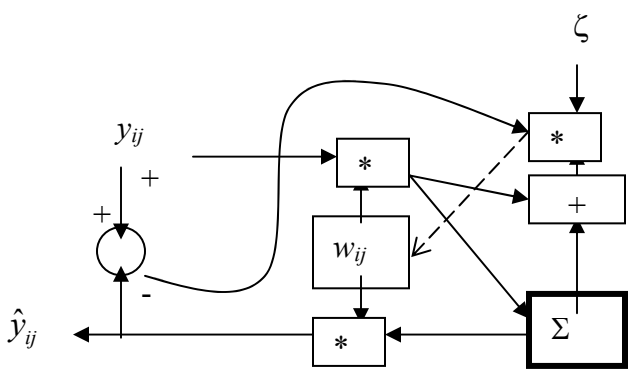


Figure 1: Single New PCA learning unit.

In Figure 1, the input data y_{ij} (i is the index of the component vector and j is the index of a value in the component vector i) is as defined in the equation (2). The Σ box provides the inner product between vectors $\boldsymbol{y}$ and $\boldsymbol{w_i}$. The result of the Σ box operation will again be summed with the previous multiplication of y_{ij}^t and w_{ij} and its output will be multiplied with the learning rate ζ before updating to w_{ij} as described in equation (4). This single unit can be cascaded into n units to obtain a PCA learning vector and this learning vector can be cascaded to obtain as many parallel eigenvector extractors as needed for each application.

III. Applications and Results

We now investigate the lossless compression based on the JPEG2000 transform and compare it to PCA-based transform. Based on compactness of the histogram it will allow us to interpret the optimal one even for the lossy compression as well.

For PCA transform, we used a 64-pixel (8x8 pixel array) as a basic vector size for this work. For example, the 256x256 pixel image can be transformed into the 1024x64 matrix and the total number of principal component analysis vector is 64.

For JPEG2000 based transform, we use 3-stage DWT using Daubechies in MATLAB.

We are focusing on two output results (shown in yellow box in Figure 2): histogram and compressed data of each technique.

- The histogram will reveal the effectiveness of the transform via its energy distribution by which regardless of the coding technique the compression factor can be correlated for comparison purposes.

- The compressed data which is based on two coding techniques: Huffman and Arithmetic demonstrates its real results.

The data flow is shown in Figure 2 below:

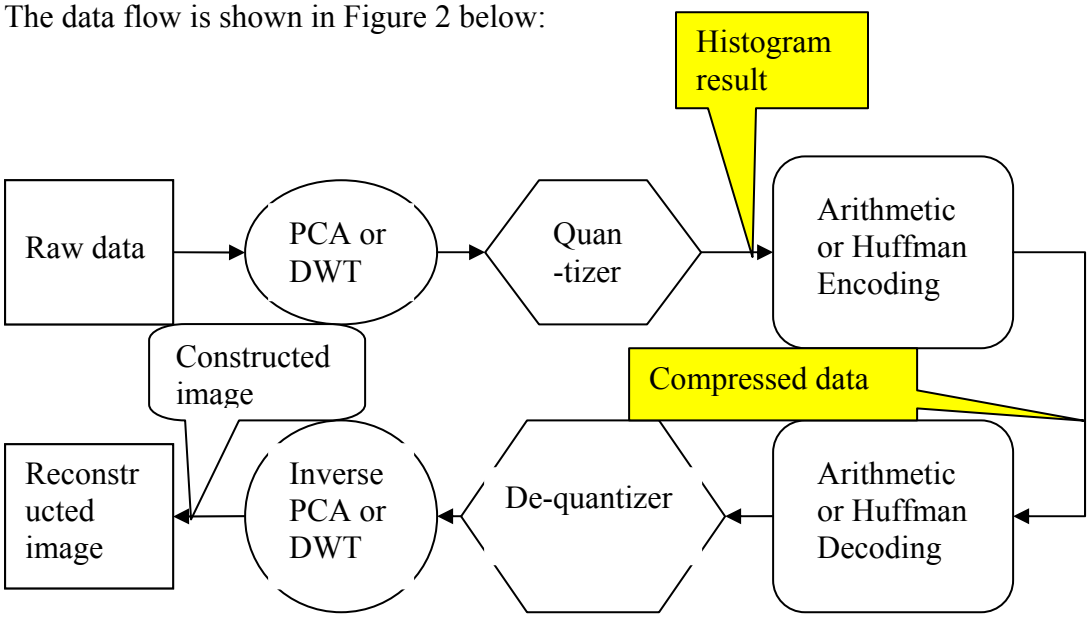


Figure 2. Data flow of our study.

We have used seven gray scale images for this comparison study which are shown in Table I below:

Table I: A list of the images and their sizes which are used in this study

	Size
Lena	256x256
Baboon	512x512
Tank1	512x512
Tank2	512x512
Tank3	512x512
Airplane	1024x1024
Airport	1024x1024

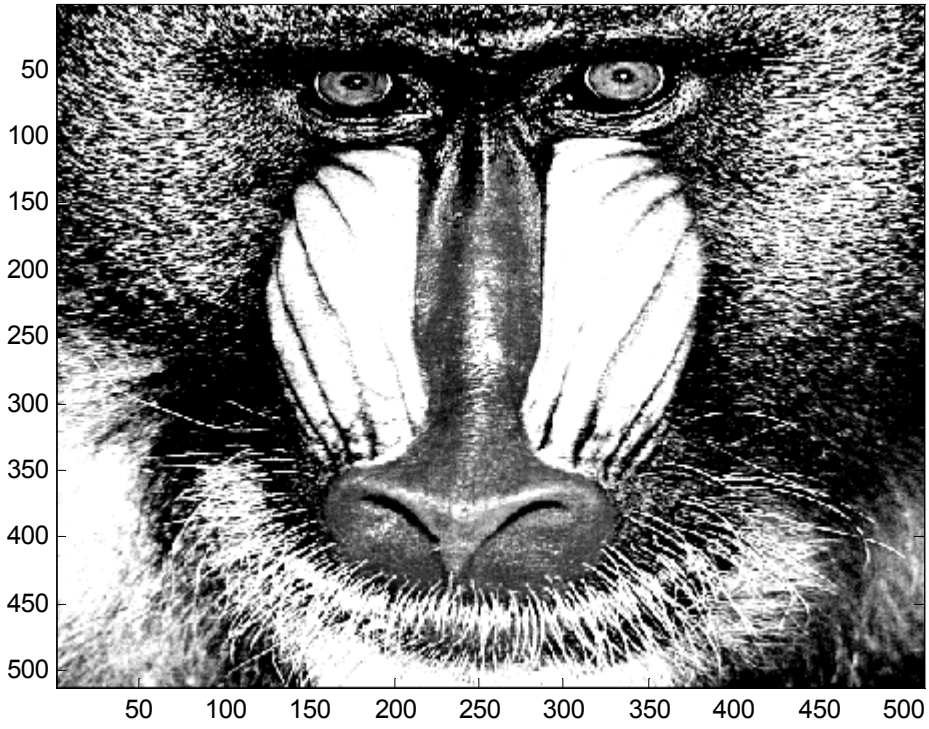
Due to the space constraints, we are only presenting the details of two images: baboon and airport and we are providing the only results of the rest of images.

1. <u>Baboon</u>

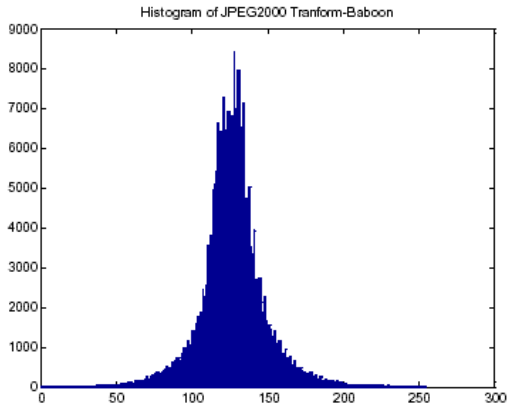
Figure 3a is a baboon image and the histogram of JPEG2000 based transform and PCA based transform are shown in Figure 3b and 3c, respectively. Obviously, the histogram indicated that PCA based transform is much more compact than the JPEG2000 based transform.

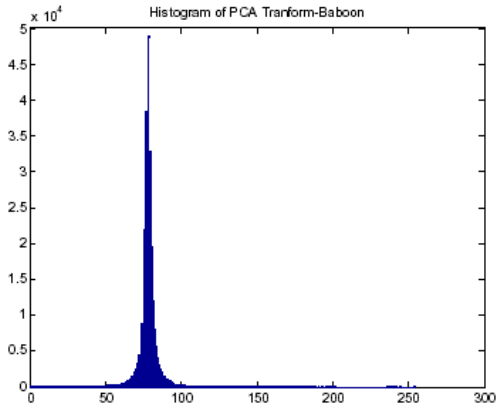
These results have demonstrated that the PCA based transform is more optimal than JEPG2000 based transform and it agreed with the theoretical prediction (PCA is data

dependent transform which is more optimal and effective than JPEG2000 known as a fixed transform).



(3a)



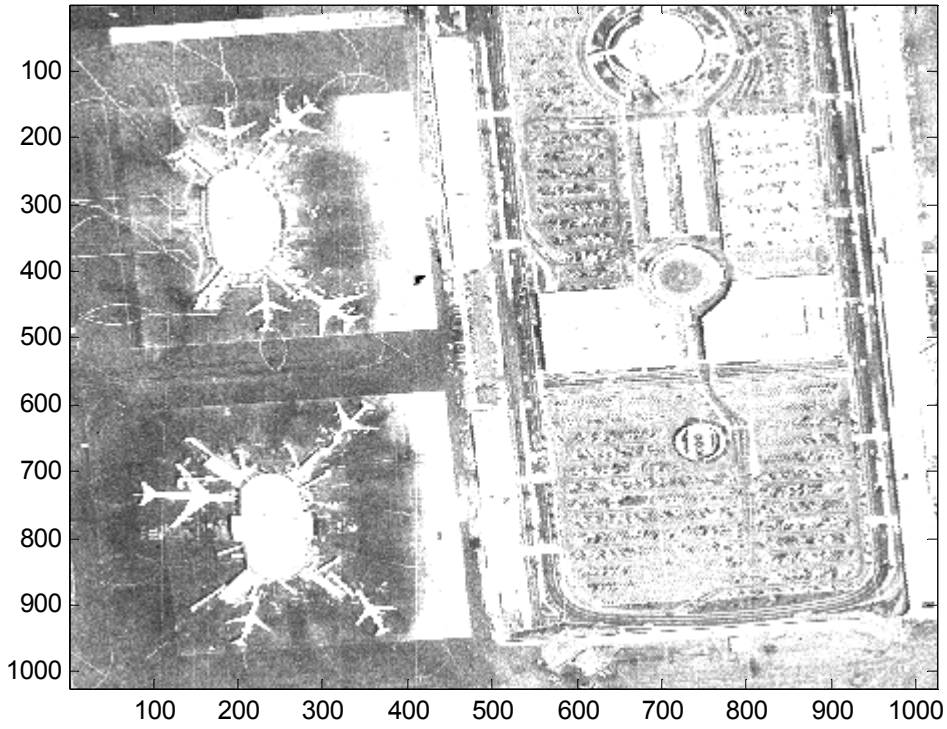


(3b) (3c)

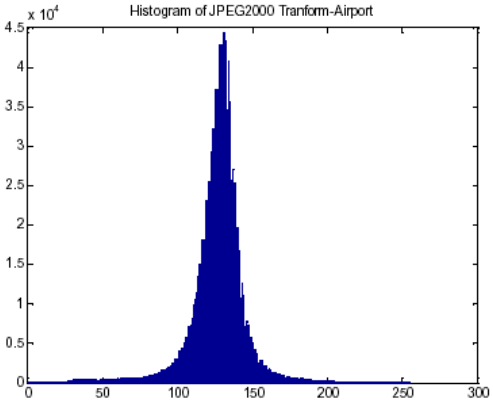
Figure 3: Original baboon image and histogram with quantized data after the transform.
3a) Original Baboon image; 3b) JPEG2000 based technique; and 3c) PCA based technique

2. <u>Airport</u>

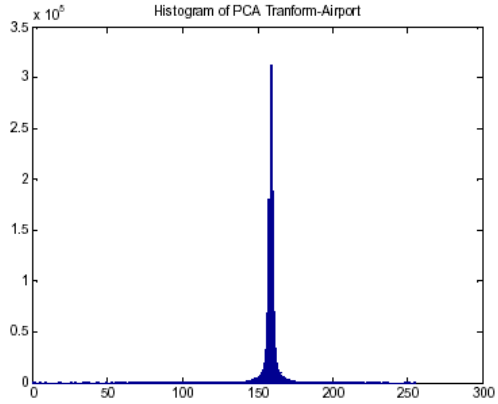
Likewise, for the airport image shown in Figure 4a, PCA based transform (Figure 4c) has more advantage in compression than the JPEG2000 based transform which is in Figure 4c.



(4a)



(4b)



(4c)

Figure 4: Lossless compression of airport (1024x1024) and histogram of airport quantized data after the transform.
4a) Lossless image; 4b) JPEG2000 based technique; and 4c) PCA based technique

A summary of our simulation results is given in Table II below:
Table II: The comparison of simulation results between PCA based and JPEG2000 based transforms

	JPEG2000		**PCA**		**Ratio C_{pca}/C_{Jpeg}**	
	Huffman	*Arithmetic*	*Huffman*	*Arithmetic*	*Huffman*	*Arithmetic*
Compressed image-Elaine (in bytes)	**46590**	**47653**	**24226**	**25661**		
Compression ratio	**1.4067**	**1.3753**	**2.7052**	**2.5539**	**1.9231**	**1.8570**
Compressed image-Baboon (in bytes)	**205709**	**206652**	**122923**	**126591**		
Compression ratio	**1.2743**	**1.2685**	**2.1326**	**2.0708**	**1.6735**	**1.6325**
Compressed image Tank1 (in bytes)	**187718**	**189618**	**96950**	**99904**		
Compression ratio	**1.3965**	**1.3825**	**2.7039**	**2.6240**	**1.9362**	**1.8980**
Compressed image –Tank2 (in bytes)	**199249**	**200989**	**108233**	**112415**		
Compression ratio	**1.3157**	**1.3043**	**2.4220**	**2.3319**	**1.8408**	**1.7879**
Compressed image-Tank3 (in bytes)	**191601**	**192810**	**98305**	**104881**		
Compression ratio	**1.3682**	**1.3596**	**2.6666**	**2.4994**	**1.9490**	**1.8383**
Compressed image-airplane (in bytes)	**696518**	**699014**	**299536**	**287358**		
Compression ratio	**1.5055**	**1.5001**	**3.5007**	**3.6490**	**2.3253**	**2.4325**
Compressed image-airport (in bytes)	**753092**	**752522**	**381675**	**380588**		
Compression ratio	**1.3924**	**1.3934**	**2.7473**	**2.7551**	**1.9731**	**1.9772**

IV. Discussion

Obviously, PCA based transform is the most optimal one in mean squared sense as stated in the previous report [4] and demonstrated in this paper (via histogram); however, it is a computational power intensive with unfriendly hardware approach. JPL has developed a real time adaptive PCA technique which satisfies the constraints such as fast computing and implementable in Very Large System Integration (VLSI) hardware.

V. Conclusions

As demonstrated earlier, the PCA based transform is advancement over JPEG200 to improve compression ratio for lossless and lossy compression. Moreover, the PCA is a feature based approach which can be even more practical not only for data compression, but also for target detection and recognition.

Acknowledgements

The research described herein was performed by the Jet Propulsion Laboratory, California Institute of Technology under contract with the National Aeronautics and Space Administration (NASA). The authors would like to thank Drs. A. Thakoor and T. Daud for their discussions and suggestions.

References

1. A. Oppenheim and R. Schafer, *Discrete-Time Signal Processing,* Prentice-Hall, Englewood, Cliffs, NJ (1989).
2. I. Daubechies, "Ten lectures on wavelets," in CBMS Conference on Waveltes, SIAM, 61 (1992).
3. A.K. Jain, *Fundamentals of Digital Image Processing.* Prentice Hall Information and System Sciences Series, 1989.
4. A.K. Jain, R.P.W. Duin, and J. Mao "Statistical Pattern Recognition: A Review", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 22, No. 1, January 2000.
5. R. Gonzalez and R. Woods, *Digital Image Processing*, 2nd Ed., Addison-Wesley, 2001.
6. I. Daubechies, "Recent Results in Wavelet Applications," J. Electronic Imaging, 7(4), 719-724, Oct. 1998.
7. T.A. Duong "Real Time On-Chip Sequential Adaptive Principal Component Analysis for Data Feature Extraction and Image Compression," GOMAC Tech-03, Vol. I, pp. 460-464, Tampa, Florida, 31 March - 3 April, 2003.
8. V. A. Duong and T. A. Duong, "*Real-Time Principal Component Analysis Software for Feature Extraction and Data Compression*", NTR# 40046, May. 03.
9. R.P.W. Duin, and J. Mao "Statistical Pattern Recognition: A Review", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 22, NO. 1, January 2000.
10. P. Baldi and K. Hornik, "Learning in linear neural networks: A survey," *IEEE Trans. Neural Networks*, Vol. 6, pp. 837-857, 1995.
11. S. Haykin, *Neural Network-A Comprehensive Foundation.* New York: Macmillan, 1994.
12. Y. Chauvin, "Principal component analysis by gradient descent on a constrained linear Hebbian cell," in *Proc. Join Int. Conf. Neural Networks*, Vol. I, San Diego, CA pp. 373-380, 1989.
13. Z. Fu and E.M. Dowling, "Conjugate gradient Eigen structure tracking for adaptive spectral estimation," *IEEE Trans. Signal Processing*, Vol. 43, pp. 1151-1160, 1995.
14. E. Oja and J. Karhunen, "On stochastic approximation of the eigenvectors and eigenvalues of the expectation of a random matrix." J. Math. Anal. Appl., vol. 106, pp. 69-84, 1985.
15. T.D. Sanger, "Optimal unsupervised learning in a single-layer linear feed-forward neural network," *Neural Networks*, Vol.2, pp. 459-473, 1989.
16. S. Bannour and M. R. Azimi-Sadjadi, "Principal Component Extraction Using Recursive Least Squares Learning," *IEEE Trans. On Neural Networks*, Vol. 6, No. 2, March 1995.
17. S.Y. Kung and K.I. Diamantaras, "Adaptive Principal Extraction and Applications," IEEE Transactions On Signal Processing, Vol. 42, No 5, pp. 1201-1217, May 1994.