



A hybrid randomized algorithm for image compression

Farshid Mehrdoust

Department of Applied Mathematics, Faculty of Mathematical Sciences, University of Guilan, Rasht, Iran
fmehrdoust@guilan.ac.ir

Abstract

In this paper, we present a randomized algorithm for image compression by improving the Markov chain Monte Carlo algorithm and by applying the principal component analysis method. Some Numerical examples are provided to show that the proposed algorithm is effective.

Keywords: Image compression, Markov chain, Principal component analysis, Robust Monte Carlo algorithm

1 Introduction

Image compression addresses the problem of reducing the amount of data required to represent a digital image. The image transformation from color to the gray-level (intensity) image I belongs to the most common algorithms. Its implementation is usually based on the weighted sum of three color components R , G , and B according to relation [1]

$$I = \alpha R + \beta B + \gamma G$$

The R , G , and B matrices contain image color components, the weights α , β , and γ were determined with regards to the possibilities of human perception [2]. There is a huge amount of algorithms [1, 2, 5, 7] based on various principles leading to the image compression. Algorithms based on the image color reduction are mostly lossy but their results are still acceptable for some applications. Among various qualified statistical procedures, the principal component analysis (PCA) is the most classical and popular one. Since it uses only simple linear algebra and avoids any probabilistic assumption on the data, PCA has been widely used in applications [2, 5-7]. Our objective in this paper is to derive randomized PCA method from robust Markov chain Monte Carlo algorithm, with the aim of providing a better dimension reduction tool for real images. Consequently, the proposed algorithm it is more efficient in applications.

2 The PCA method and computation of PCs

Principal component analysis (PCA) is an exploratory statistical method for graphical description of the information present in large data sets. In most applications, PCA consists of studying p variables measured on n individuals. When n and p are large, the aim is to synthesize the huge quantity of information into an easy and understandable form. With minimal additional effort PCA provides a road map for how to reduce a complex data set to a lower dimension to reveal the sometimes hidden, simplified structure that often underlie it. Finding principal components (PCs) reduces to finding the eigenvalues and eigenvectors of a positive-semidefinite matrix [6, 7].

Suppose that X is a vector of p random variables with a known covariance matrix Σ , and that the variances of the p random variables and the structure of the covariances or correlations between the p variables are of interest. The first step is to look for a linear function $v_1^T x$ of the elements of x having maximum variance, where v_1 is a vector of p constants $v_{11}, v_{12}, \dots, v_{1p}$ and so that

$$v_1^T x = v_{11}x_1 + v_{12}x_2 + \dots + v_{1p}x_p$$

Next, look for a linear function $v_2^T x$, uncorrelated with $v_1^T x$ having maximum variance, and so on, so that at the k^{th} stage a linear function $v_k^T x$ is found that has maximum variance subject to being uncorrelated with $v_1^T, v_2^T, \dots, v_{k-1}^T$. The k^{th} derived variable, $v_k x$ is the k^{th} PC. Up to p PCs could be found, but it is hoped, in general, that most of the variation in x will be accounted for by m PCs, where $m \ll p$. Suppose that z be the vector whose k^{th} element is z_k , the k^{th} PC, $k = 1, 2, \dots, p$. Let $z = A^T x$, where A is the orthogonal matrix whose k^{th} column, v_k , is the k^{th} eigenvector of Σ . Thus, the PCs are defined by an orthonormal linear transformation of x . We can write [6]

$$\Sigma A = A \Lambda,$$

where Λ is the orthogonal matrix whose k^{th} diagonal element is λ_k , the k^{th} eigenvalue of Σ and $\lambda = var(v_k^T x) = var(z_k)$.

Theorem 2.1 Suppose that we wish to predict each random variable, x_j in x by a linear function of y , where $y = B^T x$, and B is a $p \times q$ matrix, If σ_j^2 is the residual variance in predicting x_j from y , then $\sum_{j=1}^p \sigma_j^2$ is minimized if $B = A_q$, where A_q consist of the first q column of A .

Proof. See [6]

According to the above theorem, if we wish to get the best linear predictor of x in a q -dimensional subspace, in the sense of minimizing the sum over elements of x of the residual variances, then this optimal subspace is defined by the first q PCs. It can be shown that [7], finding the PCs reduces to finding the eigenvalues and eigenvectors of a positive-semidefinite matrix. Also, in [7] is presented PCA-MCMC algorithm for computing the PCs. We now propose a robust MCMC (RMCMC) algorithm based on control variates variance reduction that can be used to solve such an eigenpair problem.

3 The proposed algorithm

Monte Carlo methods gives statistical estimates for the functional of the solution by performing random sampling of a certain random variable whose mathematical expectation is the desired functional [3]. The basic power method [8] is the traditional starting point for a Monte Carlo determination of the eigenpair associated with the eigenvalues of largest absolute value λ_1 . While various versions of the Monte Carlo power method often compute this dominant eigenvalue very well, computing subdominant eigenvalues $\lambda_2, \lambda_3, \dots$ has often proven much more difficult and is much less frequently attempted. We will present a robust Markov chain Monte Carlo (RMCMC) algorithm that allows the simultaneous determination of a few extremal eigenpairs of a very large matrix without the need to orthogonalize pairs of vectors to each other or store all the components of any vector.

Consider the following problem of calculating eigenvalues performed by,

$$Ax = \lambda x \tag{1}$$

Suppose $A \in R^{n \times n}$ is a symmetric matrix and also

$$\lambda_{min} = \lambda_n < \lambda_{n-1} \leq \dots \leq \lambda_2 < \lambda_1 = \lambda_{max}$$

Consider the following Markov chain T_i with length i :

$$T_i : k_0 \rightarrow k_1 \rightarrow \dots \rightarrow k_i$$

where $k_j \in \{1, 2, \dots, n\}$ for $j = 1, \dots, i$ are natural numbers. The statistical nature of constructing the above Markov chain follow as:

$$p(k_0 = \alpha) = p_\alpha, \quad p(k_j = \beta | k_{j-1} = \alpha) = p_{\alpha\beta}$$

where p_α and $p_{\alpha\beta}$ show the probability of starting chain at α and transition probability from state α to β , respectively [3].

Now, define the random variable W_j using the following recursion for

$$W_j = W_{j-1} \frac{a_{k_{j-1}k_j}}{p_{k_{j-1}k_j}}, \quad j = 1, 2, \dots, i, \quad W_0 = 1. \tag{2}$$

From all possible permissible densities, we choose the following

$$p_\alpha = \frac{|h_\alpha|}{\sum_{\alpha=1}^n |h_\alpha|}$$

$$p_{\alpha\beta} = \frac{|a_{\alpha\beta}|}{\sum_{\beta=1}^n |a_{\alpha\beta}|} \quad \alpha = 1, 2, \dots, n$$

where $h \in R^n$. It can be shown that [3] for a vector $f \in R^n$ and large value of i :

$$\lambda_1 \approx \frac{E[W_i f_{k_i}]}{E[W_{i-1} f_{k_{i-1}}]} \quad (3)$$

Using deflation techniques [7, 8], once having the eigenpair (λ_i, v_i) , we can obtain eigenpairs $(\lambda_{i+1}, v_{i+1}), i = 1, 2, \dots, n$.

In [4] is shown that the accuracy of the eigenpairs has been soared by using RMCMC. In this paper, we use this algorithm for finding PCs.

Algorithm. Execute the following steps:

- (1) Set $[M,N] = \text{size}(\text{data})$.
- (2) Set $mn = \text{mean}(\text{data},2)$.
- (3) Set $\text{data}=\text{data-repmat}(mn, 1, N)$.
- (4) Compute $\text{COV}=\text{covariance}(\text{data})$.
- (5) Compute $[\text{PC}, \text{V}] = \mathbf{RMCMC algorithm}$ (covariance).
- (6) $\text{V} = \text{diag}(\text{V})$.
- (7) $\text{V} = \text{V}$ (rindices).
- (8) $\text{PC} = \text{PC}(:, \text{rindices})$.
- (9) $\text{Signals} = \text{PC}^t * \text{data}$.
- (10) Input color image C .
- (11) Decompose C to R, G, B .
- (12) Call the MCMC-PCA algorithm for R, G, B .
- (13) Output the compressed image $\hat{C} = R \oplus G \oplus B$.

4 Numerical experiments

We performed two images using PCA-MCMC and PCA-RMCMC algorithms and some of results for finding PC's are presented. In Table 1 and Figs. 1 and 2 we can see that the RMCMC algorithm gives better results than the MCMC algorithm. Also, the original images and their compressed images are shown in Figs 3-6. The numerical tests are made on Intel(R), Core(TM)i5 CPU, 2.5 GHz, personal machine using MATLAB software.

Table 1: Relative error for computing three eigenvalue by using MCMC and RMCMC algorithms

$\lambda_1(MCMC)$	$\lambda_1(RMCMC)$	$\lambda_2(MCMC)$	$\lambda_2(RMCMC)$	$\lambda_3(MCMC)$	$\lambda_3(RMCMC)$
0.1600×10^{-2}	3.1289×10^{-4}	1.0526×10^{-2}	5.1689×10^{-4}	2.8726×10^{-2}	6.8916×10^{-4}

5 Conclusion

The paper presents a new algorithm for image compression based on PCA and RMCMC methods. The hybrid algorithm has enabled us to have better accuracy than the standard MCMC algorithm.

Acknowledgements

This research has been supported by University of Guilan.

References

- [1] C. Clausen and H. Wechsler, Color image compression using PCA and ackpropagation learning, Pattern Recognition, 2000.
- [2] S. Costa and S. Fiori, Image compression using principal component neural networks, Image and Vision Computing, 2011.

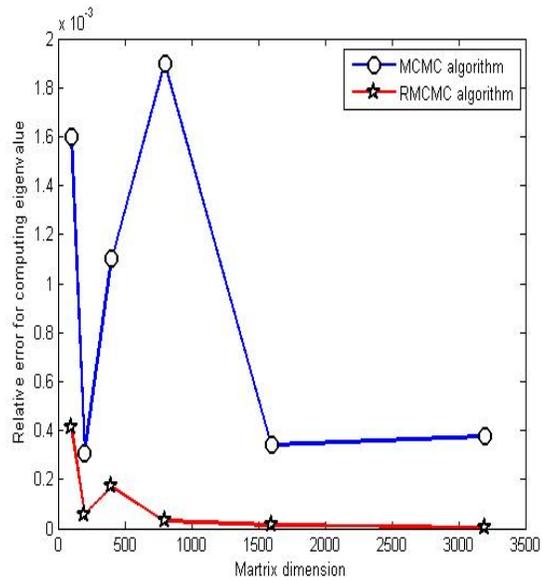


Figure 1: Comparison of relative error between MCMC and RMCMC algorithms

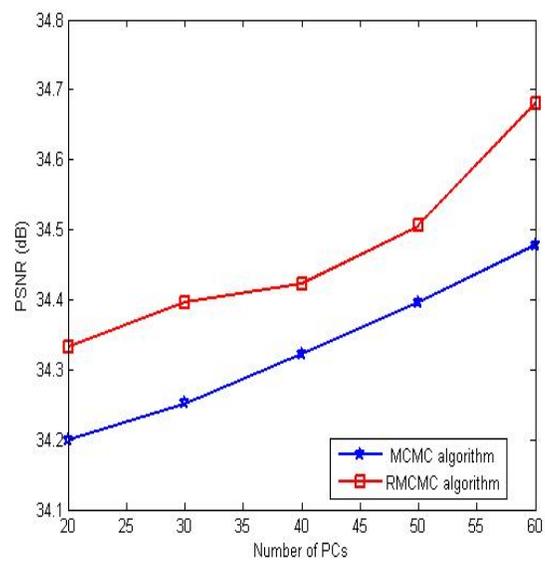


Figure 2: Average PSNR vs. number of PCs by using MCMC and RMCMC algorithms



Figure 3: Original image, Size= 845941 bytes

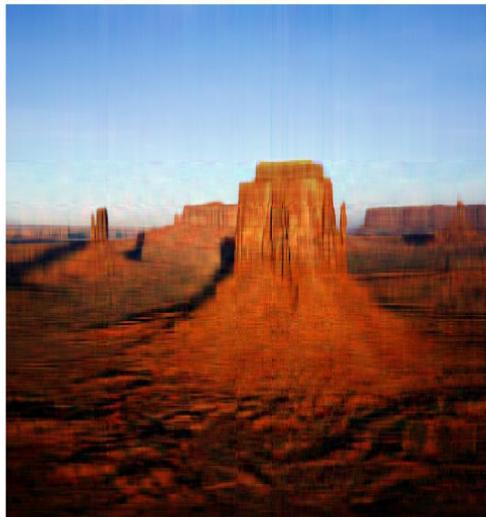


Figure 4: Compressed image with $q = 20$ PCs, Size=67910 bytes

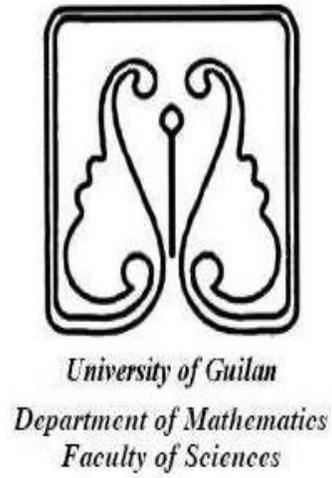


Figure 5: Original image, Size= 45,414 bytes



Figure 6: Compressed image with $q = 20$ PCs, Size= 14,101 bytes

- [3] I. Dimov and A. Karaivanova, Parallel computations of eigenvalues based on a Monte Carlo approach, Monte Carlo Method and Applications, 1998.
- [4] B. Fathi Vajargah and F. Mehrdoust, Matrix balancing and robust Monte Carlo algorithm for evaluating dominant eigenpair, Computer Science Journal of Moldova, 2010.
- [5] B. Huhle, Kernel PCA for Image Compression, Wilhelm-Schickard-Institut für Informatik Eberhard Karls Universität Tübingen, 2006.
- [6] I. T. Jolliffe, Principal Component Analysis, Springer-Verlag, 2002.
- [7] F. Mehrdoust, MCMC-PCA based compression algorithm for images, Journal of Advanced Research in Scientific Research, 2012.
- [8] Y. Saad, Numerical methods for large eigenvalue problems, Manchester University Press, 1991.