RATE-DISTORTION OPTIMIZED IMAGE COMPRESSION USING GENERALIZED PRINCIPAL COMPONENT ANALYSIS

Dohyun Ahn†*, Chang-Su Kim*‡ *and Sang-Uk Lee*†

†Signal Processing Lab., School of Electrical Engineering and Computer Science
Seoul National University, Seoul 151-741, Korea. E-mail: {dhahn,sanguk}@ipl.snu.ac.kr.
‡Dept. Electronics Eng., Korea University, Seoul 136-701, Korea. E-mail: cskim@ieee.org.

ABSTRACT

A novel image compression algorithm based on generalized principal component analysis (GPCA) is proposed in this work. Each image block is first classified into a subspace and is represented with a linear combination of the basis vectors for the subspace. Therefore, the encoded information consists of subspace indices, basis vectors and transform coefficients. We adopt a vector quantization scheme and a predictive partial matching scheme to encode subspace indices and basis vectors, respectively. We also propose a rate-distortion optimized quantizer to encode transform coefficients efficiently. Simulation results demonstrate that the proposed algorithm provides better compression performance than JPEG, especially at low bitrates.

1. INTRODUCTION

Recently, generalized principal component analysis (GPCA) was proposed in [1, 2], which clusters signals into subspaces based on linear polynomial algebra. Also, a recursive and robust version of GPCA was proposed in [3] to tolerate noises in data and handle outliers. It provides a guiding criterion to choose an effective model among all candidates. Note that these algorithms were originally devised to solve vision problems, such as motion segmentation, and machine learning problems.

In [4], Huang *et al*. showed that GPCA can be also used for image compression. More specifically, they clustered image blocks using GPCA and obtained the optimal set of basis vectors to represent the input image in a compact way. They compared GPCA with other popular transforms for image compression. However, they did not take into account additional bits to describe basis vectors. Furthermore, their coding system was not complete and did not include the quantizer and the entropy coder for transform coefficients.

In this work, we propose a complete image coding system based on GPCA. In contrast to [4], all the required components, including quantizer and entropy coder, are implemented and incorporated into the coding system. Furthermore, to maximize image quality subject to the constraint on bit budget, we propose a rate-distortion (R-D) optimization scheme for the subspace clustering and the quantization. In GPCA coding, basis vectors are adaptively chosen, and thus they should be transmitted to the decoder as side information. We employ vector quantization (VQ) and predictive partial matching (PPM) to encode the side information efficiently.

This paper is organized as follows. Section 2 describes the proposed algorithm. Section 3 presents simulation results. Finally, concluding remarks are given in Section 4.

2. PROPOSED ALGORITHM

Fig. 1 shows the block diagram of the proposed image compression system. Image blocks are first projected onto a lower dimensional vector space and then clustered by GPCA. The side information for GPCA is encoded using VQ and PPM, whereas transform coefficients are encoded by an adaptive arithmetic coder. Let us describe each function block in more detail.

2.1. Subspace Clustering and Basis Vector Generation

An input image is first divided into a set of non-overlapping blocks of size $l \times l$. In this work, l is set to 8. Then, each block is regarded as a column vector in $\mathbb{R}^{l^2}$ and is projected onto a lower dimensional space $\mathbb{R}^K$ to reduce the computational complexity of the subsequent procedures. K is a positive integer between 4 and $l^2/2$. K can be also changed to control the overall bitrate.

Those projected vectors are clustered into two subspaces recursively, generating a balanced binary tree of height 2, as shown in Fig. 2. Each node corresponds to a subspace. Specifically, the ith node at depth d, S_i^d, is the subspace that is spanned by the basis vectors of cluster i at recursion level d. In the recursion procedure, we adopt the robust GPCA algorithm in [3] to cluster data, which does not require any preliminary knowledge on the number of subspaces and the dimension of each subspace.

For each subspace, which corresponds to an internal or leaf node in the tree, we obtain basis vectors using the standard PCA technique. Then, each image block is encoded using the basis vectors of the corresponding subspace. Therefore, we encode the subspace index for each block as side information, as illustrated in Fig. 3. This information is compressed using a PPM coder. Note that PPM was originally proposed to compress binary facsimile images in a lossless manner [5].

Let k_i^d be the dimension of S_i^d. In other words, the basis for S_i^d is composed of k_i^d vectors. Therefore, each $l \times l$ block, which is classified into the subspace, is represented as a linear combination of the k_i^d basis vectors. Since k_i^d is typically much smaller than $l \times l$, a coding gain is achieved. In the example of Fig. 3, those two blocks are represented as the combinations of three basis vectors, respectively. In general, a block, classified into S_i^d, is approximated by

$$\mathbf{x}_i^d = \sum_{j=1}^{k_i^d} c_j \mathbf{v}_{i,j}^d, \tag{1}$$

where $\mathbf{v}_{i,j}^d$ is the jth basis vector for S_i^d and c_j is the corresponding weighting coefficient.

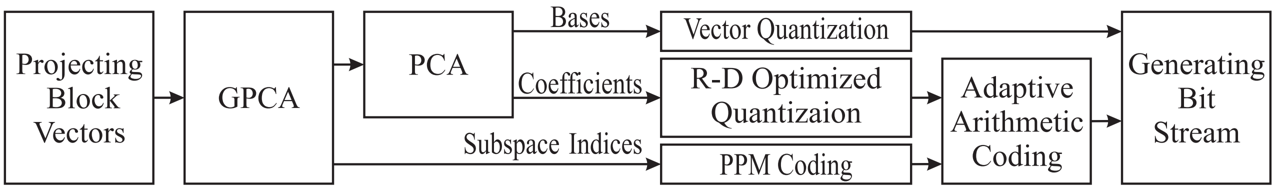


Fig. 1. Block diagram of the proposed algorithm.

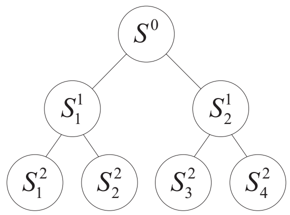


Fig. 2. Recursive clustering of subspaces.

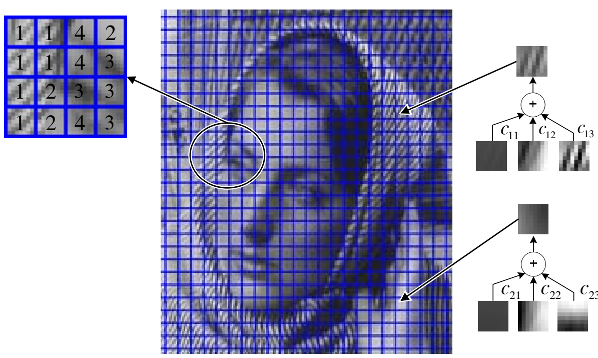


Fig. 3. Each block is classified into a subspace and then is approximated by a linear combination of the basis vectors for that space. The number in a block denotes the subspace index.

2.2. VQ Encoding of Basis Vectors

In GPCA-based image coding, subspaces and their basis vectors are adaptively obtained according to the characteristics of the input image. Therefore, all the basis vectors should be encoded and transmitted to the decoder. Fig. 4 illustrates the encoding of the "Barbara" image. All its blocks are classified into one of the two subspaces. The first subspace ($i = 1$) has 7 basis vectors, while the second subspace ($i = 2$) has 8 basis vectors. We encode those basis vectors using a VQ scheme.

As shown in Fig. 4, the first basis vector for a subspace can be well approximated by a DC component in most cases. Thus, it is not encoded in this work. The other vectors are encoded using the generalized Lloyd algorithm (GLA). We observed by experiments that the first two vectors ($j = 2$ or 3) are relatively smooth, while the remaining vectors contain high-frequency textures. Therefore, we use two VQ codebooks: one for $j = 2$ and 3 and the other for $j = 4$ to k_i^d.

After VQ, the reconstructed basis vectors do not form an orthonormal set. Therefore, as a postprocessing, we apply the Gram-Schmidt orthonormalization to the reconstructed vectors at both the encoder and the decoder. Then, the orthonormal basis is used to compute GPCA coefficients.

2.3. R-D Optimized Quantization of Coefficients

In this work, transform coefficients are uniformly quantized and the quantization indices are encoded with an adaptive arithmetic coder.

Let S_i denote a subspace and k_i be its dimension. Thus, if a block is classified into S_i, it is transformed into k_i coefficients. To achieve the best R-D performance, we optimize the set of quantizer step sizes $\mathcal{Q} = \{Q_j : j = 1, 2, \ldots, k_i\}$, where Q_j denotes the step size for the jth coefficient. More specifically, we attempt to minimize the total distortion of the image blocks in S_i

$$D_i = \sum_{m=1}^{N_i} D_m(\mathcal{Q}), \tag{2}$$

subject to the bitrate constraint

$$\sum_{m=1}^{N_i} R_m(\mathcal{Q}) \leq R_i, \tag{3}$$

where N_i is the number of blocks in S_i and R_i is the given bit budget for the blocks in S_i. $D_m(\mathcal{Q})$ and $R_m(\mathcal{Q})$ denote the distortion and the amount of bits for the mth block, respectively, when $\mathcal{Q}$ is chosen as the set of quantizer step sizes.

However, it requires too high computational complexity to solve the above constrained minimization problem exactly. To reduce the complexity, we obtain a suboptimal solution by refining the quantizer step sizes iteratively. First, we set each quantizer step size to the maximum value. Then, at each iteration, we update the step size Q_j for the jth coefficient to q, so that it maximizes the ratio

$$-\frac{\Delta D_i}{\Delta R_i}, \tag{4}$$

where ΔD_i denotes the amount of decrease in distortion, and ΔR_i denotes the amount of increase in bitrate. The iteration stops, when the target bitrate R_i is achieved. Intuitively speaking, we start from the upper left corner point on the convex hull of the achievable R-D region and attempt to move along the convex hull while increasing the bitrate. Note that a similar approach was adopted in the R-D optimization of JPEG in [6].

2.4. Pruning of Subspace Tree

In generating the set of subspaces in Fig. 2, the robust GPCA in [3] attempts to minimize the effective dimension only, which is the average number of scalar values to represent a block. The subspace configuration, however, may not be optimal in the R-D sense.

In this work, we consider all configurations, which can be obtained by pruning the full tree in Fig. 2. There are four possible configurations $\{S_1^1, S_2^1\}$, $\{S_1^1, S_3^2, S_4^2\}$, $\{S_1^2, S_2^2, S_2^1\}$ and $\{S_1^2, S_2^2, S_3^2, S_4^2\}$ as shown in Fig. 5.

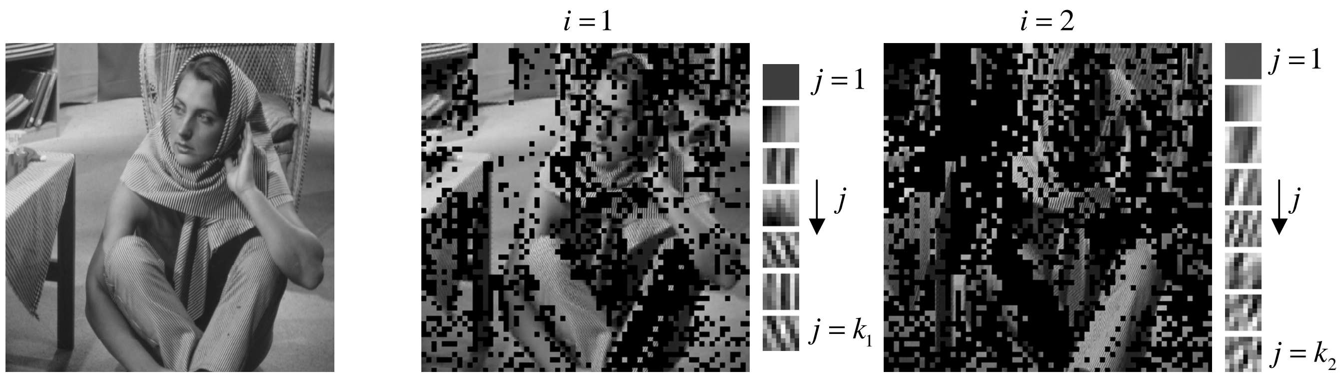


Fig. 4. The image blocks in the "Barbara" image are classified into two subspaces. The first subspace has 7 basis vectors, while the second subspace has 8 basis vectors. Note that the first subspace represents textures in one diagonal direction, whereas the second subspace represents textures in the other direction.

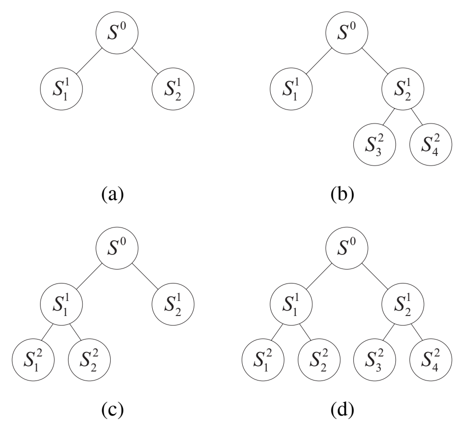


Fig. 5. Four possible configurations of subspace clustering: (a) $\{S_1^1, S_2^1\}$, (b) $\{S_1^1, S_3^2, S_4^2\}$, (c) $\{S_1^2, S_2^2, S_2^1\}$ and (d) $\{S_1^2, S_2^2, S_3^2, S_4^2\}$.

We choose one of these configurations so that it provides the best R-D performance. To avoid confusion from complex notations, let

$$\{S_i : i = 1, 2, \ldots, n\} \tag{5}$$

denote one of the configurations, where n is the number of subspaces in the configuration. Suppose that R_{total} is a total bit budget for the input image. Then, to each subspace S_i, R_i is assigned by the formula

$$R_i = R_{\mathrm{total}} \frac{N_i k_i}{\sum_{l=1}^{n} N_l k_l}. \tag{6}$$

In other words, the rate is set to be proportional to both the number of blocks in the subspace, N_i, and the dimension of the subspace, k_i. Given R_i, we perform the R-D optimized quantization for each subspace using the method in Section 2.3 and then compute the sum of the distortions for all subspaces.

Finally, the subspace configuration is selected to minimize the sum of the distortions.

Table 1. The amount of bits for encoding basis vectors.

(k_1, k_2, k_3, k_4)	No Comp. (Bytes)	VQ (Bytes)	Compression Ratio
(4,4,4,5)	8,120	65	124.9
(7,7,8,8)	1,552	130	104.2
(14,14,15,15)	22,960	270	85.0
(15,15,15,15)	23,520	280	84.0

3. SIMULATION RESULTS

Fig. 4 shows the segmentation result when GPCA is applied to the "Barbara" image of size 512×512. The block size is set to 8×8. It is observed that the textures in the basis vectors are consistent with those of the image blocks in the subspace. For the first subspace, the textures are dominated by the diagonal lines from upper left to lower right. For the second subspace, the textures contain the diagonal lines in the other direction.

For the VQ of the basis vectors, we use a training set of 189 images from the UCID database [7]. GPCA is applied to all images and then the basis vectors are divided into two training sets: those with $j = 2$ and 3 and those with $j = 4$ to k_i. Using these sets, two codebooks are generated to encode the basis vectors. Each codebook contains 1,024 codewords and the code vector size is 8×2. Therefore, a 8×8 basis vector is encoded with 4 codewords. Table 1 summarizes the bitrates, which are necessary to encode basis vectors. Before the compression, the basis vectors consume a significant portion of the overall bitrate. However, our VQ scheme effectively compresses the basis vectors and provides a compression ratio as high as 124.9. Therefore, after the compression, the additional bitrate for the basis vectors becomes negligible.

Fig. 6 compares the compression performance of the proposed algorithm with that of JPEG. Note that the proposed algorithm provides about $0.5 \sim 1.0$dB better PSNR performance than JPEG. Fig. 7 shows the reconstructed "Barbara" images, which are encoded at a bitrate of 0.143 bpp by JPEG and the proposed algorithm. It is observed that JPEG blurs high-frequency textures and yields severe blocking artifacts. On the other hand, the proposed algorithm provides a much more faithful image quality.

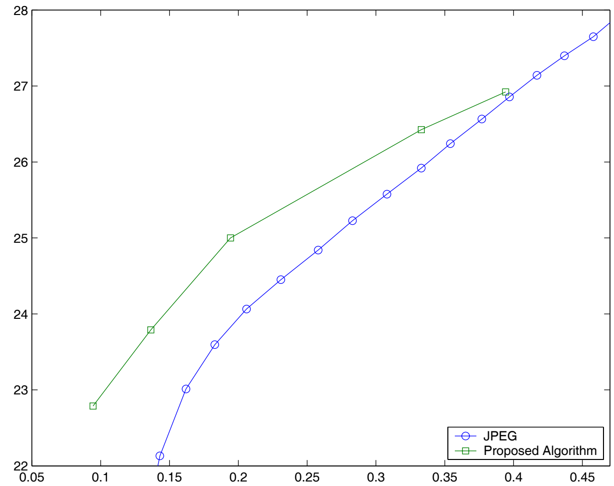


Fig. 6. The compression performances of JPEG and the proposed algorithm.

4. CONCLUSION

In this paper, we proposed an image compression algorithm based on GPCA. Contrary to the previous work in [4], we implemented all components for a complete compression system, including quantizer and entropy coder. Subspace indices were encoded using a PPM method and basis vectors were encoded with a VQ scheme. Also, an R-D optimized quantizer was designed to improve coding gain. Simulation results demonstrated that the proposed algorithm yields better compression performance than JPEG, especially in low bitrates. Currently, we are investigating more general subspace configurations and their efficient pruning algorithm.

5. REFERENCES

[1] R. Vidal, Y. Ma, and S. Sastry, “Generalized principal component analysis (GPCA),” in *Proc. IEEE CVPR*, vol. 1, pp. 621-628, 2003.

[2] R. Vidal, Y. Ma, and J. Piazzi, “A new GPCA algorithm for clustering subspaces by fitting, differentiating and dividing polynomials,” in *Proc. IEEE CVPR*, vol. 1, pp. 510-517, 2004.

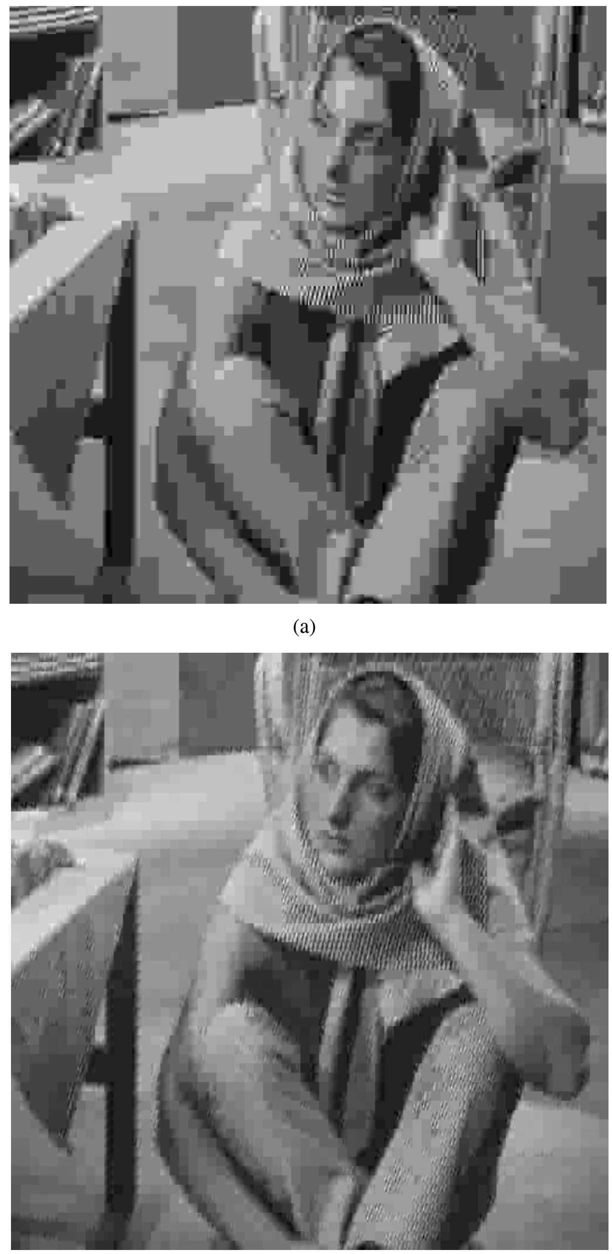
[3] K. Huang, Y. Ma, and R. Vidal, “Minimum effective dimension for mixtures of subspaces: a robust GPCA algorithm and its applications,” in *Proc. IEEE CVPR*, vol. 2, pp. 631-638, 2004.

[4] K. Huang, A. Y. Yang, and Y. Ma, “Sparse representation of images with hybrid linear models,” in *Proc. IEEE ICIP*, vol. 2, pp. 1281-1284, 2004.

[5] K. Sayood, *Introduction to Data Compression*, Academic Press, 2000.

[6] S. Wu and A. Gersho, “Rate-constrained picture-adaptive quantization for JPEG baseline coders,” in *Proc. IEEE ICASSP*, vol. 5, pp. 389-392, Apr. 1993.

[7] G. Schaefer and M. Stich, “UCID: an uncompressed color image database,” in *Proc. SPIE Storage and Retrieval Methods and Applications for Multimedia*, vol. 5307, pp. 472-480, Dec. 2003.



(a)

(b)

Fig. 7. The reconstructed “Barbara” images at 0.143 bpp: (a) JPEG and (b) the proposed algorithm.