www.dergipark.gov.tr
ISSN:2148-3736
El-Cezerî Fen ve Mühendislik Dergisi
Cilt: 9, No: 2, 2022 (424-435)

El-Cezerî Journal of Science and Engineering
Vol: 9, No: 2, 2022 (424-435)
DOI :10.31202/ecjse.951417

ECJSE

Research Paper / Makale

Lossy Image Compression Using Karhunen-Loeve Transform Based Methods

Serkan KESER[1a]

[1]Electrical and Electronics Engineering Department, Kırsehir Ahi Evran University, Kırsehir, Türkiye
skeser@ahievran.edu.tr

Received/Geliş: 12.06.2021 **Accepted/Kabul:** 10.01.2022

Abstract: Karhunen-Loeve Transform (KLT) is generally not preferred much because it is a signal-dependent transform in signal compression. However, by developing effective algorithms, image compression that provides good performance can be achieved. In this study, two effective image compression methods based on eigenvector matrices obtained using KLT are applied. Image compression is performed by creating small-sized codebooks for the two methods. The first method is based on the number of eigenvector matrices used in the training phase. Some of the highest numbers of used eigenvector matrices are used for image compression. The second compression method uses eigenvectors of autocorrelation matrices for quantization. In this approach, quantization is performed using the principal components of the eigenvector matrices. Various codebook sizes have been tested for image compression. The qualities of the reconstructed test images were compared with DCT-based JPEG and Wavelet Transform-based JPEG2000 compression methods using the PSNR metric. Experimental results show that the PSNR values of the proposed methods give better results than that of JPEG.

Keywords: Image compression, eigenvector matrices, codebooks, KLT

Karhunen-Loeve Dönüşümüne Dayalı Yöntemleri Kullanarak Kayıplı Görüntü Sıkıştırma

Öz: Karhunen-Loeve Dönüşümü (KLT) genellikle sinyal sıkıştırmada sinyal bağımlı bir dönüşüm olduğu için fazla tercih edilmemektedir. Ancak etkili algoritmalar geliştirilerek iyi performans sağlayan görüntü sıkıştırma gerçekleştirilebilir. Bu çalışmada KLT kullanılarak elde edilen öz vektör matrislerine dayalı etkili iki görüntü sıkıştırma yöntemi uygulanmıştır. İki yöntem için küçük boyutlu kod kitapları oluşturularak görüntü sıkıştırma gerçekleştirilmiştir. İlk metot, eğitim aşamasında kullanılan öz vektör matrislerinin sayısını temel almaktadır. En yüksek sayıda kullanılan öz vektör matrislerinin bir kısmı görüntü sıkıştırma için kullanılmıştır. İkinci sıkıştırma yöntemi nicemleme için oto korelasyon matrislerinin öz vektörlerini kullanır. Bu yaklaşımda nicemleme, öz vektör matrislerinin temel bileşen yönleri kullanılarak gerçekleştirilir. Görüntü sıkıştırma için çeşitli kod kitabı boyutları test edilmiştir. Yeniden oluşturulmuş test görüntülerinin kaliteleri, PSNR metriği kullanılarak DCT tabanlı JPEG ve Wavelet Dönüşümü tabanlı JPEG2000 sıkıştırma yöntemleriyle karşılaştırılmıştır. Deneysel sonuçlar, önerilen yöntemlerin PSNR değerlerinin JPEG'in PSNR değerlerinden daha iyi sonuçlar verdiğini göstermiştir.

Anahtar Kelimeler: Görüntü sıkıştırma, öz vektör matrisleri, kod kitapları, KLT

1. Introduction

Transform coding is a popular approach to signal compression through compacting the energy of the signal to fewer coefficients, hence enabling representation of the signal with as few bits as possible [1, 2]. Various transforms have been applied to signal and image compression, including very classical transforms like Discrete Fourier Transform (DFT), Walsh-Hadamard Transform

How to cite this article
Keser, S., "*Lossy Image Compression Using Karhunen-Loeve Transform Based Methods*" El-Cezerî Journal of Science and Engineering, 2022, 9(2); 424-435.

Bu makaleye atıf yapmak için
Keser, S., "Karhunen-Loeve Dönüşümüne Dayalı Yöntemleri Kullanarak Kayıplı Görüntü Sıkıştırma" El-Cezerî Fen ve Mühendislik Dergisi 2022, 9(2); 424-435.
ORCID : [a] 0000-0001-8435-0507

(WHT), Discrete Cosine Transform (DCT), Wavelet Transform (WT), and Karhunen Loeve Transform (KLT) [1,4-8,10,11,14,15]. Usually, transform coding-based compression methods are preferred for lossy compression cases. Among these, DCT- and Wavelet Transform (WT)-based methods are widely used for image compression [8,10,11,14,15]. DCT is the most popular transform in JPEG-based methods and WT also became popular with JPEG2000 [9,10,12,14,17].

Although KLT is known to be the most efficient orthogonal transform in terms of energy compaction and decorrelation [2,7], it was not used for actual compression purposes due to its signal-dependent construction. However, with the help of increased computing power of communication systems and computers, signal specific methods (such as KLT-based compression algorithms) could stand a chance for real image compression applications, provided that the corresponding subspace domain could be accompanied by smart sample representation techniques.

Effective image coding studies are using KLT in the literature. In [21], one of them was a method proposed by Waldemar and Ramstad that uses the energy compression properties between the Karhunen-Loeve transform (KLT) and singular value decomposition (SVD). In [22], a method using the set partitioning in hierarchical trees (SPIHT) and KLT for color image coding was proposed by Kassim and Lee. In [23], image compression including SPIHT, KLT, DCT, discrete 9/7 biorthogonal wavelet transform was performed by Ashin. Blanes and Serra proposed a new method based on KLT that gives better coding performance than wavelets for the codification of remote-sensing imagery in [24].

In this paper, we propose two novel KLT-based methods which use eigenvector quantization for codebook generation. One of the proposed methods (Method-2) is the same as the method [13] previously used to code the speech signal. By using these quantized eigenvectors as the basis for the signal representation, an alternative to the classical approach of “block transform followed by sample quantization” is aimed to be achieved. The coding and construction efficiencies of these codebooks are expected to contribute positively to the image compression literature. The compaction of the basis vectors is achieved through several process steps. First, spatially similar blocks of training set images are assigned to the same cluster. In this way, the blocks with high correlations are collected in the same cluster. For each cluster, an autocorrelation matrix and the corresponding eigenvector matrix are obtained. The transform to be used in compression is constructed by selecting, grouping, and quantizing these eigenvectors using vector quantization. Test images are reconstructed using the resultant transform matrices corresponding to different codebook sizes. The results are compared with standard DCT-based JPEG and WT-based JPEG 2000. The qualities of the reconstructed test images were compared with DCT-based JPEG and Wavelet Transform-based JPEG2000 compression methods using the PSNR metric. Experimental results show that the PSNR values of the proposed methods give better results than that of JPEG.

2. Experimental Methods

We proposed two different quantization methods to produce eigenvector matrix codebooks. The main goal is to use small-size codebooks to reduce the calculation time and to allocate fewer bits per block, whilst maintaining the highest possible image fidelity.

2.1 Proposed KLT-based Methods

The proposed KLT-based methods depend on the construction of transform matrices using judiciously selected, and quantized eigenvectors of the autocorrelation matrices. A reason for KLT not to appear in most image processing techniques is its dependence on data statistics. However, after refining the eigenvectors for a certain class of images, and overall useful transform matrix that can be used for test images could be obtained. The process starts with training images (we had 2929 training images with size 256x256 in our experiments) and splitting them into non-overlapping 8×8 blocks. The first refinement stage is a grouping of these matrices using K-means quantization [16] according to the Frobenius norm as a distance metric. In our experiments, the process yields a total of 8192 clusters. The *p-th* data matrix $\mathbf{X}_p$ is obtained by combining the rows of all blocks with the size of 8x8 in a cluster and, this data matrix is used to compute the *p*-th autocorrelation matrix $\boldsymbol{R_P}$

$$\boldsymbol{R_p} = \boldsymbol{X_p X_p^T}, \qquad p=1,2,...,r \tag{1}$$

where r is the total number of autocorrelation matrices (or clusters) $\boldsymbol{R_p} \in \boldsymbol{R^{8x8}}$, and $\boldsymbol{X_p} \in \boldsymbol{R^{8x8T}}$. Then, eigenvectors are calculated for each autocorrelation matrix. The autocorrelation matrix $\boldsymbol{\Phi}_p$ is formed by stacking the eigenvectors corresponding to the largest K eigenvalues of $\mathbf{R}_p$ as shown in Eq. (2)

$$\boldsymbol{\Phi}_p = \{\phi_1, \phi_2, \ldots, \phi_K\}, \tag{2}$$

where $\boldsymbol{\phi}$'s are 8-dimensional eigenvectors and $\boldsymbol{\Phi}_p \in \mathbf{R}^{8\times K}$. The eigenmatrix codebook ($\boldsymbol{\Phi}^{cb}$) can be written as a stack of the eigenvector matrices $\boldsymbol{\Phi}_p$ as in Eq.3 :

$$\boldsymbol{\Phi}^{cb} = \{\boldsymbol{\Phi}_1, \boldsymbol{\Phi}_2, \ldots, \boldsymbol{\Phi}_r\}, \tag{3}$$

where r is the total number of eigenvector matrices in the codebook. The next stage is the codebook generation step from these 8192 eigenvector matrices. The number of eigenvector matrices is decreased from 8192 to 512, 256, 128, 64, 32, 16, and 8 separately using the KLT-based quantization methods as described below. Thus, seven different codebooks are constructed.

2.1.1 The Proposed Method-1 (PM1)

In the first proposed method, quantization is directly applied to the codebook which contains 8192 eigenvector matrices. Here, all blocks of the training image set are reconstructed using all the eigenvector matrices in the eigenmatrix codebook.

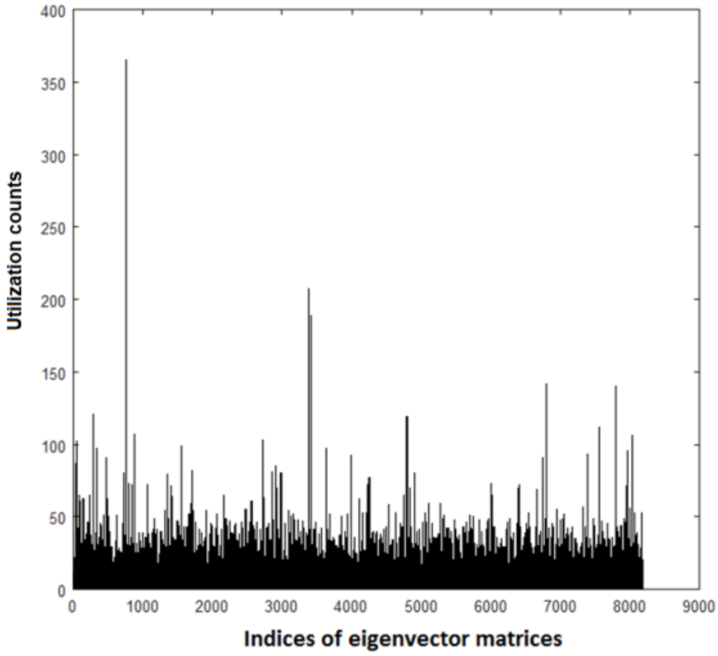


Figure 1. The histogram of the training blocks

Then, the most frequently used eigenvector matrices are selected from the codebook. The number of the selected eigenvector matrices are taken as 8, 16, 32, 64, 128, 256 and 512, for different compression ratios. The histogram of the utilization counts of training blocks is given in Figure 1.

2.1.2 The Proposed Method-2

The second proposed method performs a novel quantization technique in which the eigenvector matrices are grouped into a cluster that contains an equal number of eigenvector matrices in their quantization regions [13]. Thus, the clusters consist of the eigenvector matrices including principal eigenvectors in similar directions.

An example illustration of this approach is as follows. Let $\mathbf{A}=[\mathbf{a}_1\ \ \mathbf{a}_2]$ and $\mathbf{B}=[\mathbf{b}_1\ \ \mathbf{b}_2]$ be matrices of principal component vectors $\mathbf{a}_i$ and $\mathbf{b}_i$, and $\mathbf{u}_1$ and $\mathbf{u}_2$ are the sum of the first $(\mathbf{a}_1, \mathbf{b}_1)$ and second $(\mathbf{a}_2, \mathbf{b}_2)$ principle components of **A** and **B** matrices as $\mathbf{u}_1 = (\mathbf{a}_1+\mathbf{b}_1)$ and $\mathbf{u}_2 = (\mathbf{a}_2+\mathbf{b}_2)$, so that $\mathbf{U}=[\mathbf{u}_1\ \ \mathbf{u}_2]$. The normalized matrix is $\mathbf{U}_{norm}=\left[\frac{\mathbf{u}_1}{\|\mathbf{u}_1\|}\ \frac{\mathbf{u}_2}{\|\mathbf{u}_2\|}\right]$. Two matrices **A** and **B** have the same cluster in the two-dimensional vector space and the normalization process is shown in Figure 2 [13].

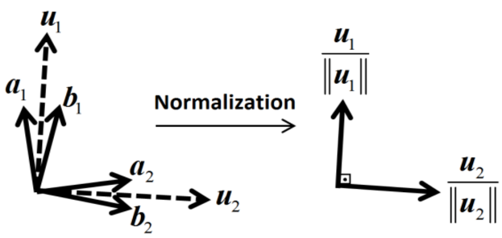


Figure 2. The vector quantization in the two- dimensional vector space.

The proposed codebook structure for the second method can be realized with the following algorithm:

Step-1) Set t=1.
Step-2) The scaler values (D_s) are found by the inner product of the first eigenvector matrix
($\boldsymbol{\Phi}_1 = [\phi_1^1\ \phi_2^1 \cdots \phi_K^1]$) and the remaining eigenvector matrices ($\boldsymbol{\Phi}_s = [\phi_1^s\ \phi_2^s \cdots \phi_K^s]$) in the eigenmatrix codebook:

$$D_s = \langle \phi_1^1, \phi_1^s \rangle + \langle \phi_2^1, \phi_2^s \rangle \cdots + \langle \phi_K^1, \phi_K^s \rangle, \quad s=2,3,\ldots,r$$

where r is the total number of eigenvector matrices in the eigenvector codebook. The size of the eigenvector matrices is, therefore, $8\times K$ (with $K<8$).

Step-3) Find L-1 eigenvector giving the biggest values of D_s, (s=2,3,..,r) among r–1 eigenvector matrices. Combine the first eigenvector matrix and L-1 eigenvector matrices, and form t-th cluster ($\boldsymbol{\psi}^t$), which consists of L eigenvector matrices.

Step-4) Find t-th total eigenvector matrix ($\boldsymbol{\Phi}^t$) by using the eigenvector matrices in the t-th cluster:

$$\mathbf{\Phi}^t = \sum_{j=1}^{L} \boldsymbol{\psi}_j^t$$

Step-5) The total eigenvector matrix ($\mathbf{\Phi}^t$) is normalized ($\mathbf{\Phi}^t_{norm}$) and then, the eigenvector matrices are removed from the eigenmatrix codebook.

Step-6) Increase t by 1. If $t=m$ (m is the number of clusters) terminate the algorithm. Otherwise, go to **Step 2**.

At the end of the algorithm, a new eigenmatrix codebook is constructed, which can be represented as $\mathbf{\Phi}^{cb}=\{\mathbf{\Phi}^{\mathbf{1}}_{norm},\mathbf{\Phi}^{\mathbf{2}}_{norm},\ldots,\mathbf{\Phi}^{\boldsymbol{m}}_{norm}\}$, where $m<<r$.

2.2. The Proposed Eigenvector Selection Rule

Once the codebooks are generated according to the above eigenvector quantization methods, as described above, the compression of all other images (i.e. the test images) is performed by selecting the most suitable eigenvector matrix from the codebooks. First, the test image is divided into 8×8 pixel blocks $\left(\mathbf{X}^1,\mathbf{X}^2,\ldots,\mathbf{X}^{\boldsymbol{U}}\right)$ and transformed to the frequency domain using a transform matrix ($\boldsymbol{\Phi}^l \in \boldsymbol{R}^{\mathbf{8}\times\mathbf{8}}$). As a result of the transformation, most of the signal energy lies at low frequency and is concentrated in just a few coefficients of the coefficient matrix ($\mathbf{Y}^{\boldsymbol{k},\boldsymbol{l}}$). To achieve this transformation, the *k-th* pixel block $\left(\mathbf{X}^{\boldsymbol{k}}\right)$ is projected onto the designed subspace, spanned by the eigenvector matrices ($\boldsymbol{\Phi}^l$) selected from the eigenmatrix codebook ($\mathbf{\Phi}^{cb}$). The *k-th* coefficient matrix ($\mathbf{Y}^{\boldsymbol{k},\boldsymbol{l}}$) for the *l-th* eigenvector matrix can be written as:

$$\mathbf{Y}^{\boldsymbol{k},\boldsymbol{l}}=\left(\mathbf{\Phi}^{\boldsymbol{l}}\right)^{\boldsymbol{T}}\mathbf{X}^{\boldsymbol{k}}\mathbf{\Phi}^{\boldsymbol{l}},\ l=1,2,\ldots,M \text{ and } k=1,2,\ldots,U \tag{4}$$

where $\mathbf{X}^{\boldsymbol{k}}$ is *k-th* data matrix with the size of 8×8 and M is the codebook size. Quantization of the transform samples is achieved using the quantization matrix (**Q**), then rounding to the nearest integer value:

$$\boldsymbol{C}^{k,l}_{u,v} = round\left(\frac{\boldsymbol{Y}^{k,l}_{u,v}}{\boldsymbol{Q}_{u,v}}\right),\ l=1,2,\ldots,M \text{ and } k=1,2,\ldots,U. \tag{5}$$

The reconstruction of the encoded image is achieved by multiplying each element of $\boldsymbol{C}^{k,l}_{u,v}$ by the corresponding element in the quantization matrix and reversing the operations. The dequantized subspace matrix for the *k-th* image block becomes:

$$\boldsymbol{S}^{k,l}_{u,v} = \boldsymbol{Q}_{u,v} \times \boldsymbol{C}^{k,l}_{u,v},\quad l=1,2,\ldots,M \tag{6}$$

The *k-th* decompressed image block ($\widehat{\boldsymbol{X}}^{k,l}$) for the *l-th* eigenvector matrix is, therefore:

$$\widehat{\boldsymbol{X}}^{k,l} = \boldsymbol{\Phi}^l\boldsymbol{S}^{k,l}(\boldsymbol{\Phi}^l)^T,\quad l=1,2,\ldots,M . \tag{7}$$

The index of the *l-th* eigenvector matrix in the eigenmatrix codebook is;

$$n = argmin\left(\left\|\boldsymbol{X}^k - \widehat{\boldsymbol{X}}^{k,l}\right\|\right),\quad l=1,2,\ldots,M . \tag{8}$$

For all blocks of the test image, these indices are sent to the decoder to select the most suitable eigenvector matrix in the decoder.

2.3. Compression Evaluation

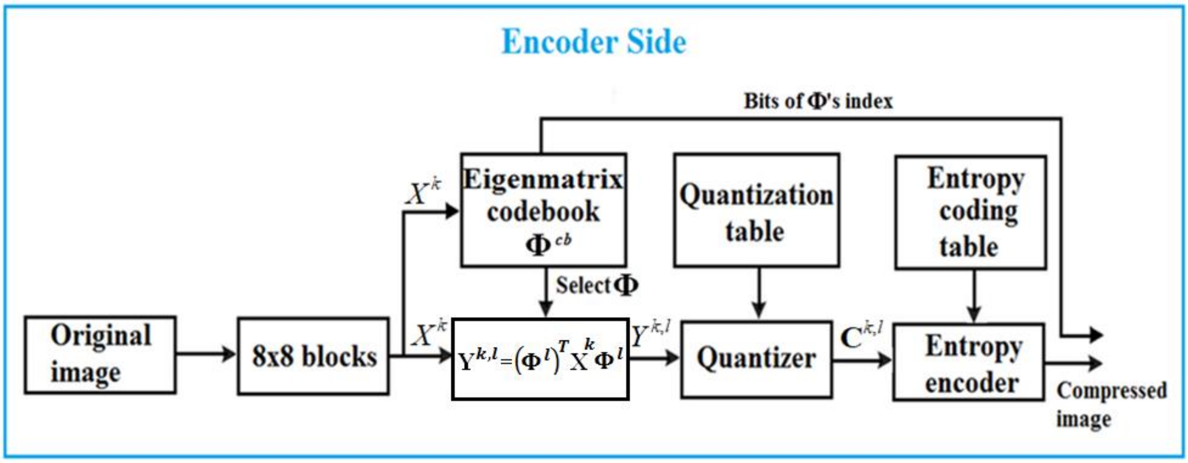
The final bitrate (BR) depends on the quantization of the eigenmatrices, which results in the codebook size. Without any further lossless compression, the codebook indices are binary represented. Firstly, a bit rate (Br_1) is found by using the Run-Length Coding (RLC) algorithm of the JPEG in the encoder side for all blocks of the test image. Subsequently, the bits of the indices belonging to the selected eigenvectors from the codebook are added Br_1 and the bits per pixel value of all system is equal to Br.

$$Br = Br_1 + \frac{log_2(M)}{N}, \tag{9}$$

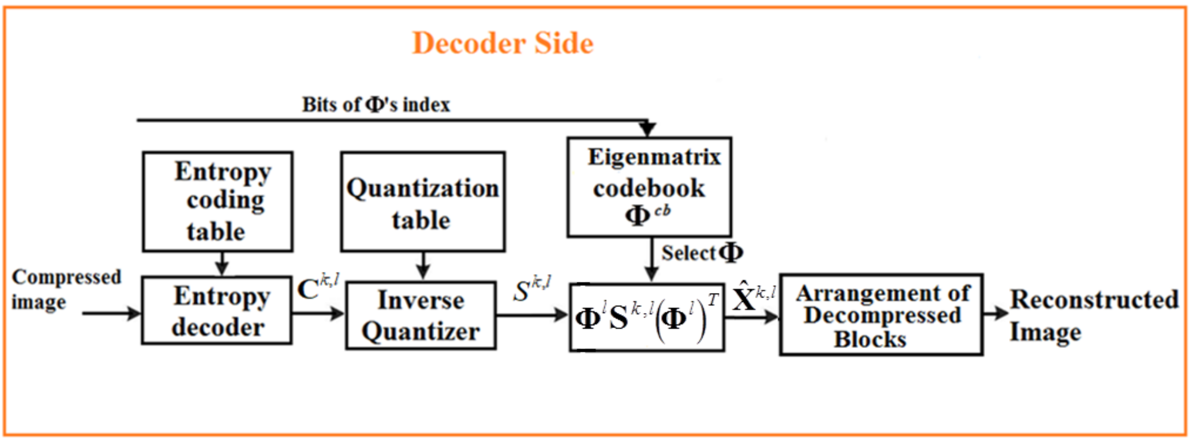
where N is equal to 64, which is the number of pixels in an 8×8 block, and M is the codebook size, taking the values of 8, 16, 32, 64, 128, 256 or 512.

2.4. The Structure of the Decoder and Encoder

The encoder and decoder structures used in this study are given schematically in Figure 3. It can be noticed that the structures resemble the DCT-based JPEG method, where the eigenvector matrices are used in the eigenmatrix codebook instead of the fixed DCT matrix.



(a)



(b)

Figure 3. The encoder (a) and decoder (b) structures of the proposed KLT-based methods

At the encoder side of Figure 3, the test image samples are firstly grouped into non-overlapping and consecutive 8×8 blocks. Then, each block is transformed into a set of coefficients. To achieve this process, the most suitable eigenvector matrix is selected from the codebook by using the selection rule that is described in subsection 2.2. The coefficient matrix ($\mathbf{Y}^{k,l}$) for the *l-th* eigenvector matrix is obtained by using Eq.4. Then, the coefficients of ($\mathbf{Y}^{k,l}$) are quantized individually by using the quantization matrix ($\mathbf{Q}$) as given in Eq.5 and the quantized coefficient block ($\mathbf{C}^{k,l}$) is obtained in this way.

Finally, each quantized coefficient block is entropy coded by using Run-Length Coding [5] with different run-lengths for transmission and sent to the decoder. In addition, the bits of the selected eigenvector matrix are sent to the decoder side. On the decoder side, the operations are reversed. First, the compressed bits are entropy decoded to separate the individual coefficients and coefficient block ($\mathbf{C}^{k,l}$) is found.

Then, the coefficients ($\mathbf{C}^{k,l}$) are dequantized by using the inverse process as given in Eq.6 and dequantized subspace matrix ($\boldsymbol{S}^{k,l}$) is found. Next, the most suitable eigenvector matrix, which is selected from the eigenvector codebook, is found by using the indices corresponding to the bits of the selected eigenvector matrix on the encoder side. The *k-th* decompressed image block ($\widehat{\boldsymbol{X}}^{k}$) for the most suitable eigenvector matrix is found by using Eq.7. Next, the decompressed blocks are arranged to reconstruct the image. Finally, the test image is reconstructed by using all these decoding steps. The qualities of JPEG, JPEG2000 and the proposed KLT-based methods were compared by using the PSNR metric.

3. Experimental Studies

In the experimental studies, we used two image sets to evaluate the algorithms. The first set contains training images that are taken from the most widely used dataset in literature [8,18,19,20]. This dataset contains a wide range of indoor and outdoor images including the 15-scene categories. We used 9 categories which are CALsuburb (241 images), MITcost (360 images), MITforest (328 images), MIThighway (260 images), MITinsidecity (308 images), MITmountain (374 images), MITopencountry (410 images), MITstreet (292 images), MITtallbuilding (356 images) in the training set.

Therefore, the training set contains 2929 images and a total of 2.999.296 blocks are generated by using all the training images. The size of the images is 256 × 256 and all of them are 8-bit grayscale images. The second or test set contains 12 images that are collected from test images used frequently in the literature such as Lake, Pirate, House, Mountain, Church, Barbara etc. All these images are in gif format, and they have the size 256x256. As mentioned in Section 2, the blocks of the images in the training set are quantized using the k-means algorithm and thus 8192 clusters are obtained by the quantization.

Then, an eigenmatrix codebook with the size of 8192 is generated by using these clusters. Finally, the size of the codebook is reduced to 8,16,32,64,128,256 and 512 by using the proposed quantization methods. In the proposed first quantization method, the eigenvectors corresponding to the largest 4 eigenvalues are used. In this study, the value of 4 for K is experimentally found and gives the highest average PSNR value for the test images. In the experimental studies, a different bit rate for each codebook is found using Eq. (9).

The PSNR values of JPEG, JPEG2000 and the proposed methods are found according to these bit rates. The codebooks with different sizes are tested on 12 images. The performances of the JPEG,

JPEG2000 and the proposed methods are evaluated with PSNR values found for various bitrates (BPP) and, they are shown in Figure 4 and Figure 6. The horizontal axis indicates the bitrate (BPP), the vertical axis indicates PSNR values and, the numbers, which are in the parentheses and above the horizontal axis, are shown the sizes of codebooks such as 8, 16, 32, 64, 128, 256 and 512.

As can be seen from Figure 4, the PSNR values of the PM1 and JPEG are close to each other. JPEG2000 has better PSNR values than these two methods. The average PSNR values for each test image are given for JPEG2000, JPEG and the proposed methods in Figure 5 and Figure 7. In Figure 5 below, the average PSNR values for each test image are given for JPEG2000, JPEG and PM1.

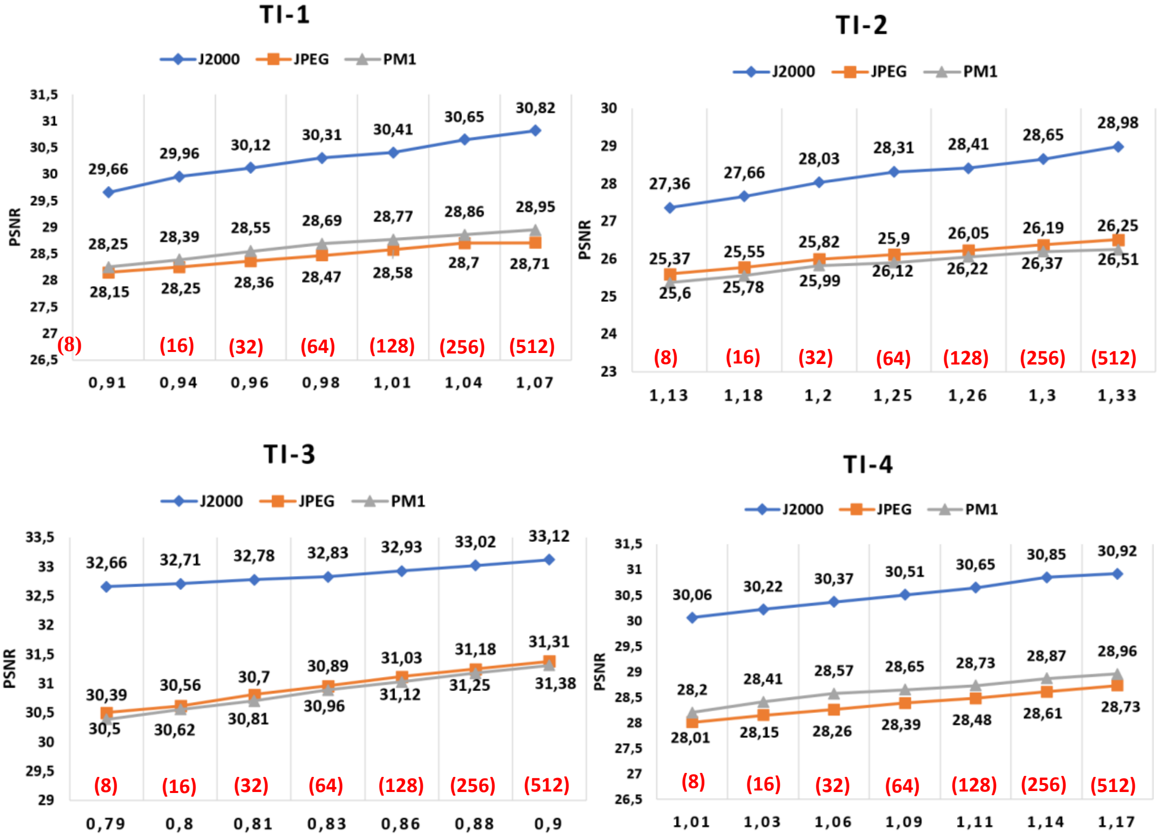


Figure 4. PSNR values of JPEG2000, JPEG and the proposed method-1 (PM1) for 4 test images

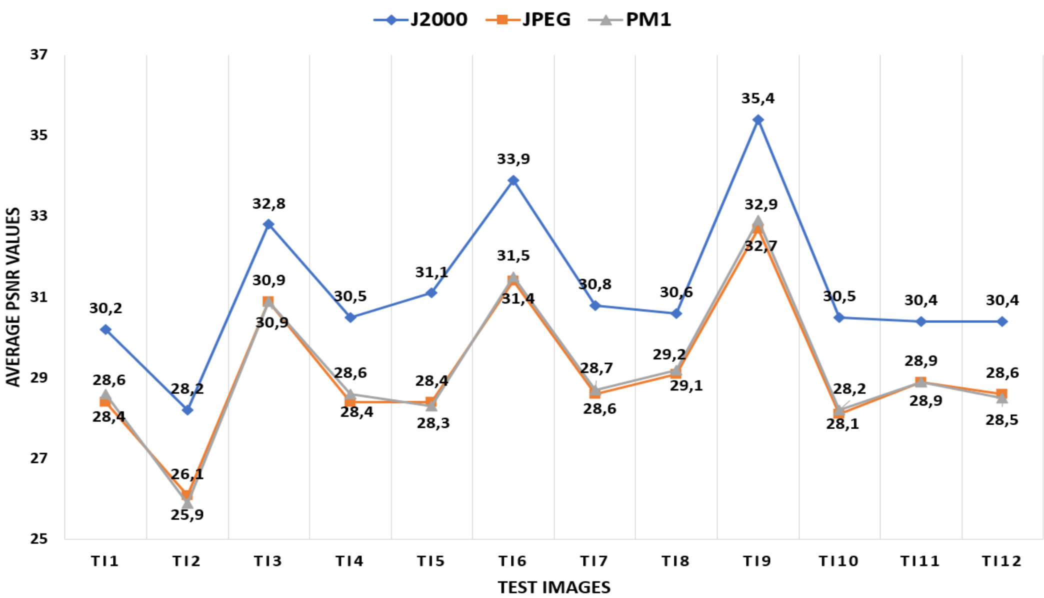


Figure 5. Average PSNR values of JPEG2000, JPEG and the proposed method-1 (PM1) for 12 test images.

In Figure 6, the PM2 gives higher PSNR values when compared with those of the JPEG for all codebook sizes and test images. From Figure 4 and Figure 6, it can be seen that the quality increases as we increase the codebook size which leads to an increase in bit rate.

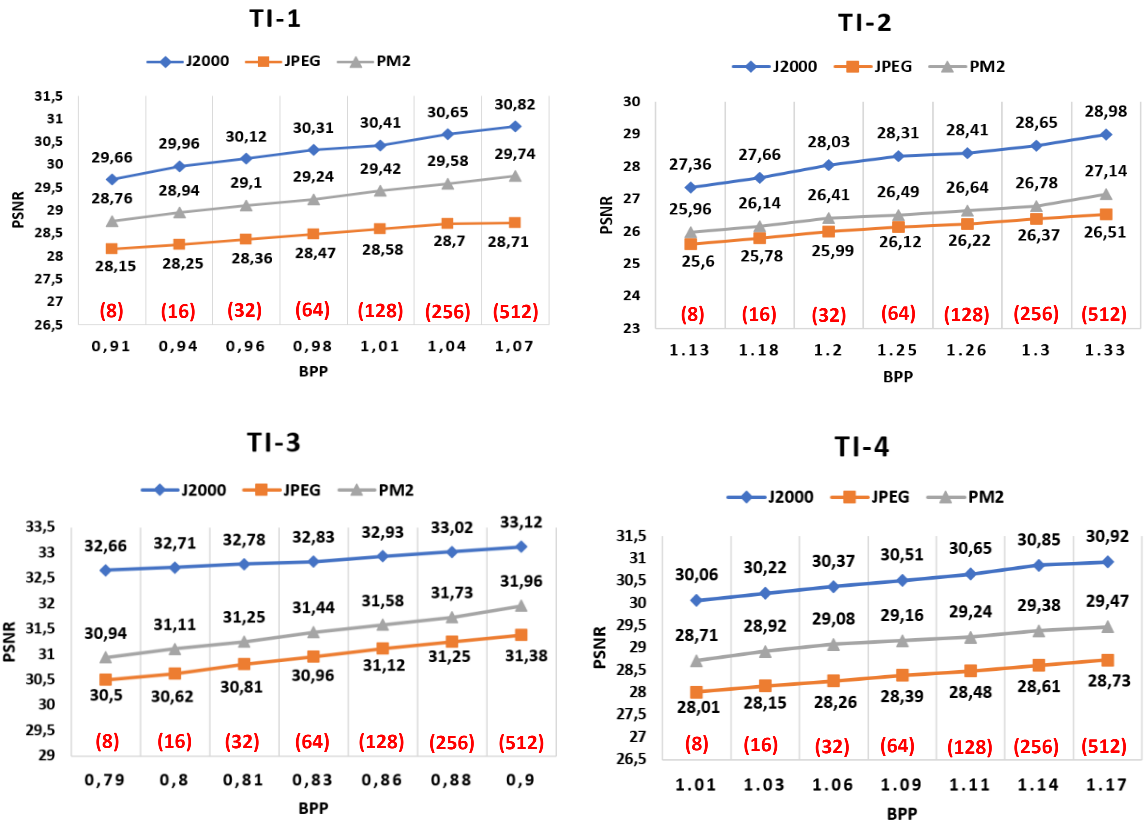


Figure 6. PSNR values of the JPEG2000, JPEG, and the proposed method-2 (PM2) for 4 test images

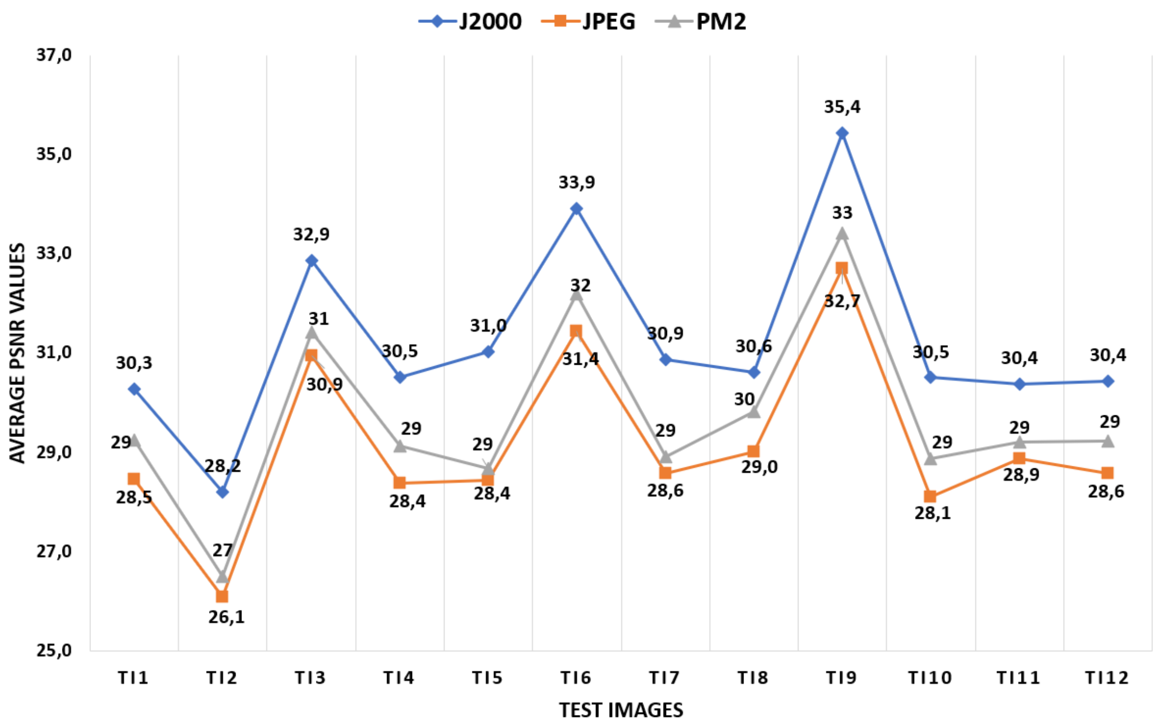


Figure 7. Average PSNR values of the JPEG2000, JPEG, and the PM2 for 12 test images

In Figure-7, the PSNR values of PM2 are higher than those of the JPEG for 12 test images. However, the PSNR values of JPEG2000 are higher than those of PM2 and the JPEG for all test images. The average PSNR values of 12 test images for each method, whose PSNR values are indicated in Figure 5 and Figure 7, are given in Table 1.

Table 1. Average PSNR values for 12 test images.

Average JPEG2000 PSNR	**Average JPEG PSNR**	**Average PM1 PSNR**	**Average PM2 PSNR**
31,25	29,13	29,18	29,72

It can be seen from Table 1 that the average PSNR values of PM1 and PM2 are higher than those of the JPEG.

4. Conclusions

Since KLT is a signal dependent transformation, special algorithms are required to use it in the compression of signals. When similar studies in the literature are examined, it is seen that the eigenvector matrix is sent to the decoder side in most studies and the codebook is not used in these studies. It is seen that the studies using the eigenvector codebook were created with different algorithms according to the second method proposed in this study.

In this paper, image blocks are clustered according to their similarity in the spatial domain, and compression is performed by applying KLT to the blocks obtained from these clusters. Then, the eigenvectors of the image blocks are grouped on the basis of their principal components, and quantization was performed by obtaining a single eigenvector matrix from the eigenvector matrices of a cluster. As a result, image compression was performed using the eigenvector matrices found as a result of quantization. This method (PM2) offers a new approach for image compression. In addition, another KLT-based method (PM1) that compresses according to the number of eigenvector matrices used in the training phase is proposed. Thus, the proposed image compression methods with low computation time are implemented using eigenvector codebooks with small sizes. Moreover, the proposed methods can significantly reduce the complexity, and it can be used easy for hardware implementation.

To reduce calculation delay and to allocate lower bits per pixel, various codebook sizes were performed using these methods and the obtained PSNR values were compared to those of JPEG and JPEG2000. It has been seen that the second KLT-based method (PM2) is more efficient than the first KLT-based method (PM1) and produces images with better quality. Besides, the experimental results also show that the PM2 and PM1 gave better PSNR values when compared with those of DCT-based JPEG. However, the performance of the JPEG2000 is better than those of the JPEG, PM1, and PM2.

Author(s) Contributions

Author contributed to the final version of the manuscript.

Competing Interests

The authors declare that they have no competing interests.

References

[1]. Katharotiya, A., Patel, S., Goyani, M., Comparative analysis between DCT & DWT techniques of image compression. Journal of information engineering and applications, 2011, 1(2), 9-17.

[2]. Ballé, J., Chou, P. A., Minnen, D., Singh, S., Johnston, N., Agustsson, E., Toderici, G., Nonlinear transform coding. IEEE Journal of Selected Topics in Signal Processing, 2020, 15(2), 339-353.

[3]. Khan, S., Nazir, S., Hussain, A., Ali, A., Ullah, A., An efficient JPEG image compression based on Haar wavelet transform, discrete cosine transform, and run length encoding techniques for advanced manufacturing processes. Measurement and Control, 2019, 52(9-10), 1532-1544.

[4]. Roy A.B., Dey D., Mohanty B., Banerjee D., Comparison of FFT, DCT, DWT, WHT compression techniques on electro cardiogram and photo plethysmography signals. Int. J. Comp. Appl., 2012, 975-888.

[5]. Iqbal, Y., Kwon, O. J., Improved JPEG Coding by Filtering 8× 8 DCT Blocks. Journal of Imaging, 2021, 7(7), 117.

[6]. Singh H., Sharma S., Hybrid image compression using DWT, DCT and Huffman encoding techniques. Int J Emerg Technol Adv Eng, 2012, 2: 300-306.

[7]. Goyal, V. K., Theoretical foundations of transform coding. IEEE Signal Processing Magazine, 2001, 18(5), 9-21.

[8]. Oliva A., and Torralba A., "Modeling the shape of the scene: a holistic representation of the spatial envelope," International Journal of Computer Vision, 2001, vol.42, no. 3, pp. 145-175.

[9]. Xie Y., Jing X., Sun S., Hong L., A fast and low complicated image compression algorithm for predictor of JPEG-LS. In IEEE IC-NIDC, 2009, 353-356.

[10]. Gungor, M. A., Gencol, K., Developing a compression procedure based on the wavelet denoising and JPEG2000 compression. Optik, 2020, 218, 164933.

[11]. Dimitriadis, A., Taubman, D., Augmenting JPEG2000 With Wavelet Coefficient Prediction. In 2020 IEEE International Conference on Image Processing (ICIP), IEEE, 2020, pp. 1276-1280.

[12]. Agrawal,S., Finite-State Vector Quantization Techniques for Image Compression. International Research Journal of Innovations in Engineering and Technology, 2020, 4(7), 1.

[13]. Keser, S., Gerek, Ö. N., Seke, E., and Gülmezoğlu, M. B., A subspace based progressive coding method for speech compression. Speech Communication, 2017, 94, 50-61.

[14]. Abd-Elhafiez, W. M., Gharibi, W., Heshmat, M., An efficient color image compression technique. Telkomnika, 2020, 18(5), 2371-2377.

[15]. Zhang, S. Q., Zhang, S. F., Wang, X. N., and Wang, Y., The image compression method based on adaptive segment and adaptive quantified. In 2008 3rd International Conference on Innovative Computing Information and Control, IEEE, 2008, pp. 353-353.

[16]. Jain, A. K., Data clustering: 50 years beyond K-means. Pattern recognition letters, 2010, 31(8), 651-666.

[17]. Hartung, F., Girod, B., Watermarking of uncompressed and compressed video. Signal processing, 1998, 66(3), 283-301.

[18]. Fei-Fei, L., Perona, P., A bayesian hierarchical model for learning natural scene categories. In 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), IEEE, 2005, Vol. 2, pp. 524-531.

[19]. Das, D., Scene classification using pyramid histogram of multi-scale block local binary pattern. Int. J. Comput. Sci. Appl.(IJCSA), 2014, 4(4), 15-25.
[20]. Lazebnik, S., Schmid, C., Ponce, J., Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06) , IEEE, 2006, Vol. 2, pp. 2169-2178.
[21]. Waldemar, P., Ramstad, T. A., Hybrid KLT-SVD image compression. In 1997 IEEE International Conference on Acoustics, Speech, and Signal Processing, IEEE,1997, Vol. 4, pp. 2713-2716).
[22]. Kassim, A. A., Lee, W. S., Embedded color image coding using SPIHT with partially linked spatial orientation trees. IEEE Transactions on Circuits and Systems for Video Technology, 2003, 13(2), 203-206.
[23]. Ashin, R., Morimoto, A., Vaillancourt, R., Image compression with multiresolution singular value decomposition and other methods. Mathematical and Computer Modelling, 2005, 41(6-7), 773-790.
[24]. Blanes, I., Serra-Sagristà, J., Pairwise orthogonal transform for spectral image coding. IEEE Transactions on Geoscience and Remote Sensing, 2010, 49(3), 961-972.