**Paweł FORCZMAŃSKI**
WEST POMARANIAN UNIVERSITY OF TECHNOLOGY, FACULTY OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGIES,
ul. Żołnierska 49, 71-210 Szczecin

# Strategies of covariance matrix calculation in the PCA method applied for three-dimensional data

**Ph.D. eng. Pawel FORCZMAŃSKI**

P. Forczmański (1974) received M.Sc. degree in 1998 and Ph.D. degree in 2002 from Faculty of Computer Science and Information Technology, Technical University of Szczecin (at present West Pomeranian University of Technology, Szczecin). His scientific interests include digital image processing, recognition and analysis, methods of dimensionality reduction and image classification in biometrical applications, content-based image retrieval and visual surveillance.

*e-mail: pforczmanski@wi.zut.edu.pl*

**Abstract**

The paper presents a problem of reducing dimensionality of data structured in three-dimensional matrices, like true-color RGB digital images. In this paper we consider an application of Principal Component Analysis to one of the most typical image processing tasks, namely image compression. Unlike the cases reported in the literature [5,11,12] the compression being an application of three-dimensional PCA is performed on image blocks organized as three-dimensional structures (see Fig.1). In the first step, an image, which is stored as a three-dimensional matrix is decomposed into non-overlapping 3D blocks. Then each block is projected into lower-dimensional representation (1D or 2D) according to the chosen strategy: concatenation of rows, concatenation of columns, integration of rows, integration of columns [13] and concatenation of slices. Next, the blocks are centered (subtraction of mean value) and covariance matrices are being calculated. Finally, the eigenproblem is being solved on the covariance matrices giving a set of eigenvalues and eigenvectors, which are a base for creation of transformation matrices. Each block is then multiplied by respective transformation functions created from truncated eigenvectors matrices giving its reduced representation. The experimental part of the paper shows the comparison of strategies of calculating covariance matrices in the aspect of image reconstruction quality (evaluated by Peak Signal-to-Noise Ratio).

**Keywords**: dimensionality reduction, Principal Component Analysis, covariance matrix, image compression.

## Strategie tworzenia macierzy kowariancji w metodzie PCA dla danych trójwymiarowych

**Streszczenie**

W niniejszym artykule przedstawiono problem redukcji wymiarowości danych zorganizowanych w trójwymiarowych macierzach za pomocą metody Analizy Głównych Składowych (PCA). W przeciwieństwie do znanych metod prezentowanych w literaturze [5,11,12] wybrane metody opisane w pracy zakładają wykonanie obliczeń dla danych zagregowanych, bez ich rozdzielania na kanały. W pierwszym kroku algorytmu obraz kolorowy (macierz trójwymiarowa) jest dekomponowany na niezależne sub-bloki (3D). Następnie każdy z bloków jest poddawany projekcji 1D lub 2D zgodnie z przyjętą strategią: poprzez konkatenację wierszy, konkatenację kolumn, integrację wierszy, integracje kolumn lub konkatenację warstw. W kolejnym kroku są one centrowane i obliczane są odpowiednie macierze kowariancji. Następnie obliczany jest ich rozkład, który służy do stworzenia macierzy transformacji 3D PCA. Za ich pomocą przeprowadzana jest redukcja wymiarowości danych obrazowych. W przypadku omawianym w niniejszej pracy kompresji poddany jest obraz RGB i oceniana jest jakość rekonstrukcji (PSNR) jako funkcja liczby pozostawionych współczynników przekształcenia.

**Słowa kluczowe**: redukcja wymiarowości, Analiza Głównych Składowych, macierz kowariancji, kompresja obrazu.

## 1. Introduction

The problem of dimensionality reduction in the tasks of image processing and recognition has been presented in the scientific literature for many years.

Subspace methods, which aim to reduce the dimension of the data while retaining the statistical separation property between distinct classes, have been a natural choice for these tasks. Color images of real scenes, however, are generally high dimensional and their within-class variations is much larger than the between-class variations, which will cause the serious performance degradation of classical subspace methods [1]. By far, various subspace methods have been proposed and applied to image recognition. One of the most often employed methods is Principal Component Analysis (PCA) also known as Hotelling Transform (HT) or Karhunen-Loeve Transform (KLT), depending of the realization and application.

The PCA is a mathematical way of determining such linear transformation of a sample of points in multi-dimensional space, which exhibits the properties of the sample along the coordinate axes [2]. Along them the sample variances are extremes (maxima and minima), and uncorrelated. The name comes from the principal axes of an ellipsoid (e.g. the ellipsoid of inertia). By their definition, the principal axes will include those along which the point sample has little or no spread (minima of variance). Hence, an analysis in terms of principal components can show (linear) interdependence in data. Using a cutoff on the spread along each axis, a sample may thus be reduced in its dimensionality.

Principal component analysis has in practice been used to reduce the dimensionality of different data [3-8], and to transform interdependent coordinates into significant and independent ones [9, 10]. It requires input data to be organized in a specific manner, which leads to the different strategies of calculations, which finally gives slightly different results.

The review of the literature shows that the authors do not deal with a problem of organizing input data for PCA method and do not investigate the influence of chosen method on the results of their algorithms. Only a few authors use the term "3D PCA"[11] or "3D KLT" [12], thus the data organization is not a main issue discussed in that papers.

The paper is organized as follows: the section two presents the general algorithms for covariance matrix creation in the case of three-dimensional data, section three includes an application of these methods for a sample task of image reconstruction and the last section presents some conclusions.

## 2. PCA for three-dimensional data

In this paper we consider an application of PCA to the reduction of dimensionality of 3D data in one of the most typical image processing tasks, namely image compression. In many cases reported in the literature the compression is performed on image blocks and most of authors deal with single-channel images, which leads to several simplifications.

In our case we assume that the analyzed image is given as a matrix of $Q \times P \times R$ elements. Typical approach to the image processing assumes that input data are divided into $R$ channels or slices, like in RGB or YCbCr color representations. Further, the methods like JPEG decompose it into independent structures and process them separately. On the other hand, there is a possibility to treat input three-dimensional data as coherent structures and do the processing on whole matrices. This approach is suitable for multispectral data collected from remote-sensing devices like satellites and dedicated airplanes, where each image element is represented by more than 3 channels (often dependent on each other).

In the first step, an image matrix is decomposed into $L$ non-overlapping blocks of $M\times N\times K$ points each. Each block can be represented by a set of $K$ $M\times N$ matrices (slices, channels, layers):

$$\underset{k\in K}{\forall} X^{k}_{M\times N} = \begin{bmatrix} x_{1,1,k} & \cdots & x_{1,N,k} \\ \vdots & \ddots & \vdots \\ x_{M,1,k} & \cdots & x_{M,N,k} \end{bmatrix}, \qquad (1)$$

A graphical representation of image block is presented in Fig. 1.

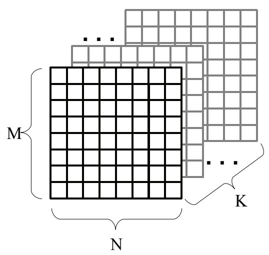


Fig. 1. Structure of image block with *M* columns, *N* rows and *K* slices
Rys. 1. Struktura bloku obrazu z *M* wierszami, *N* kolumnami *i K* warstwami

Then, each block is transformed into low dimensional representation, i.e. for $K$=1 – one-dimensional vector being a result of concatenation of image rows or columns [4] (of size $1\times MN$ or $MN\times 1$) - see case *a* below, or two vectors being a results of row or column decomposition (of size $1y\times M$ and $N\times 1$) [13] – case *b*. For $K\geq 2$, the most obvious is a concatenation of all rows, columns or slices, thus a one dimensional vector's dimensions are $MNK \times 1$ – case *c*.

Another possible representation of three-dimensional matrix suitable for classical matrix algebra is a set of three matrices of $N\times MK$, $M\times NK$ and $K\times MN$ elements – case *d*.

All these representations are presented below and denoted with letters form (a) to (b):

a) $K$ one-dimensional row concatenation $MN\times 1$, one for each slice $k$:

$$\hat{X}^{k}_{MN\times 1} = \left[x_{1,1,k},\ldots,x_{1,N,k},\ldots,x_{M,1,k},\ldots,x_{M,N,k}\right]^{T}, \qquad (2)$$

which is equal to one-dimensional column concatenation $MN$x1, one for each slice:

$$\hat{X}^{k}_{MN\times 1} = \left[x_{1,1,k},\ldots,x_{M,1,k},\ldots,x_{1,N,k},\ldots,x_{M,N,k}\right]^{T}, \qquad (3)$$

b) $K$ two-dimensional representations: $N$ columns of $M\times 1$ and $M$ rows of $N\times 1$ elements, respectively, for each slice $k$:

$$\hat{X}^{(k,r)}_{N\times M} = \begin{bmatrix} x_{1,1,k} & \cdots & x_{M,1,k} \\ \vdots & \ddots & \vdots \\ x_{1,N,k} & \cdots & x_{M,N,k} \end{bmatrix}, \qquad (4)$$

$$\hat{X}^{(k,c)}_{M\times N} = \begin{bmatrix} x_{1,1,k} & \cdots & x_{1,N,k} \\ \vdots & \ddots & \vdots \\ x_{M,1,k} & \cdots & x_{M,N,k} \end{bmatrix}, \qquad (5)$$

c) single representation $MNK\times 1$ consisting of all columns concatenation:

$$\hat{X}_{MNK\times 1} = \left[x_{1,1,1},\ldots,x_{M,1,1},\ldots,x_{M,N,1},\ldots,x_{M,N,k}\right]^{T}, \qquad (6)$$

or rows concatenation:

$$\hat{X}_{MNK\times 1} = \left[x_{1,1,1},\ldots,x_{1,N,1},\ldots,x_{M,N,1},\ldots,x_{M,N,k}\right]^{T}, \qquad (7)$$

or slices concatenation:

$$\hat{X}_{MNK\times 1} = \left[x_{1,1,1},\ldots,x_{1,1,k},\ldots,x_{M,N,1},\ldots,x_{M,N,k}\right]^{T}, \qquad (8)$$

which are equal in the sense of information content;

d) three two-dimensional representations $M\times NK$, $N\times MK$ and $K\times MN$:

$$\hat{X}^{(r)}_{N\times MK} = \begin{bmatrix} x_{1,1,1} & \cdots & x_{M,N,1} & \cdots & x_{M,N,K} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{1,N,1} & \cdots & x_{M,N,1} & \cdots & x_{M,N,K} \end{bmatrix}, \qquad (9)$$

$$\hat{X}^{(c)}_{M\times NK} = \begin{bmatrix} x_{1,1,1} & \cdots & x_{1,N,1} & \cdots & x_{M,N,K} \\ \vdots & & \vdots & & \vdots \\ x_{M,1,1} & \cdots & x_{M,N,1} & \cdots & x_{M,N,K} \end{bmatrix}, \qquad (10)$$

$$\hat{X}^{(b)}_{K\times MN} = \begin{bmatrix} x_{1,1,1} & \cdots & x_{1,N,1} & \cdots & x_{M,N,1} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ x_{1,1,K} & \cdots & x_{1,N,K} & \cdots & x_{M,N,K} \end{bmatrix}, \qquad (11)$$

Mean vector is calculated as follows:

a)

$$\underset{k\in K}{\forall} \overline{X}^{k}_{MN\times 1} = \frac{1}{L}\sum_{l}^{L}\hat{X}^{k}_{l}, \qquad (12)$$

b)

$$\underset{k\in K}{\forall} \overline{X}^{(k,r)}_{N\times M} = \frac{1}{L}\sum_{l}^{L}\hat{X}^{(k,r)}_{l}, \quad \overline{X}^{(k,c)}_{M\times N} = \frac{1}{L}\sum_{l}^{L}\hat{X}^{(k,c)}_{l}, \qquad (13)$$

c)

$$\overline{X}_{MNK\times 1} = \frac{1}{L}\sum_{l}^{L}\hat{X}_{l}, \qquad (14)$$

d)

$$\overline{X}^{(r)}_{N\times MK} = \frac{1}{L}\sum_{l}^{L}\hat{X}^{(r)}_{l}, \qquad (15)$$

$$\overline{X}^{(c)}_{M\times NK} = \frac{1}{L}\sum_{l}^{L}\hat{X}^{(c)}_{l}, \qquad (16)$$

$$\overline{X}^{(b)}_{K\times MN} = \frac{1}{L}\sum_{l}^{L}\hat{X}^{(b)}_{l}, \qquad (17)$$

Based on all vectors/matrices and mean elements, the covariance matrix/matrices, which correspond to the variation of blocks in the whole set [4, 5], are being calculated according to the following:

a)

$$Cov(\hat{X}^{k}) = \frac{1}{L}\sum_{l}^{L}\left(\hat{X}^{k}_{MN\times 1} - \overline{X}^{k}_{MN\times 1}\right)\left(\hat{X}^{k}_{MN\times 1} - \overline{X}^{k}_{MN\times 1}\right)^{T}; \qquad (18)$$

b)

$$Cov(\hat{X}^{(k,r)}) = \frac{1}{L}\sum_{l}^{L}\left(\hat{X}^{(k,r)}_{N\times M} - \overline{X}^{(k,r)}_{N\times M}\right)\left(\hat{X}^{(k,r)}_{N\times M} - \overline{X}^{(k,r)}_{N\times M}\right)^{T}; \qquad (19)$$

$$Cov(\hat{X}^{(k,c)}) = \frac{1}{L}\sum_{l}^{L}\left(\hat{X}^{(k,c)}_{M\times N} - \overline{X}^{(k,c)}_{M\times N}\right)\left(\hat{X}^{(k,c)}_{M\times N} - \overline{X}^{(k,c)}_{M\times N}\right)^{T}; \qquad (20)$$

c)

$$Cov(\hat{X}) = \frac{1}{L}\sum_{l}^{L}\left(\hat{X}_{MNK\times 1} - \overline{X}_{MNK\times 1}\right)\left(\hat{X}_{MNK\times 1} - \overline{X}_{MNK\times 1}\right)^{T}; \qquad (21)$$

d)

$$Cov(\hat{X}^{(r)}) = \frac{1}{L}\sum_{l}^{L}\left(\hat{X}^{(r)}_{N\times MK} - \overline{X}^{(r)}_{N\times MK}\right)\left(\hat{X}^{(r)}_{N\times MK} - \overline{X}^{(r)}_{N\times MK}\right)^{T}; \qquad (22)$$

$$Cov(\hat{X}^{(c)}) = \frac{1}{L}\sum_{l}^{L}\left(\hat{X}^{(c)}_{M\times NK} - \overline{X}^{(c)}_{M\times NK}\right)\left(\hat{X}^{(c)}_{M\times NK} - \overline{X}^{(c)}_{M\times NK}\right)^{T}; \quad (23)$$

$$Cov(\hat{X}^{(b)}) = \frac{1}{L}\sum_{l}^{L}\left(\hat{X}^{(b)}_{K\times MN} - \overline{X}^{(b)}_{K\times MN}\right)\left(\hat{X}^{(b)}_{K\times MN} - \overline{X}^{(b)}_{K\times MN}\right)^{T}; \quad (24)$$

After diagonalization of the covariance matrix/matrices, we get adequate matrices of eigenvectors and eigenvalues, which satisfy the following condition [6]:

a)

$$\underset{k\in K}{\forall} D^{k} = \left(V^{k}\right)^{T} Cov(\hat{X}^{k})\, V^{k}, \quad (25)$$

b)

$$\underset{k\in K}{\forall} D^{(k,r)} = \left(V^{(k,r)}\right)^{T} Cov(\hat{X}^{(k,r)})\, V^{(k,r)}, \quad (26)$$

$$\underset{k\in K}{\forall} D^{(k,c)} = \left(V^{(k,c)}\right)^{T} Cov(\hat{X}^{(k,c)})\, V^{(k,c)}, \quad (27)$$

c)

$$D = \left(V\right)^{T} Cov(\hat{X})\, V, \quad (28)$$

d)

$$D^{(r)} = \left(V^{(r)}\right)^{T} Cov(\hat{X}^{(r)})\, V^{(r)}; \quad (29)$$

$$D^{(c)} = \left(V^{(c)}\right)^{T} Cov(\hat{X}^{(c)})\, V^{(c)}; \quad (30)$$

$$D^{(b)} = \left(V^{(b)}\right)^{T} Cov(\hat{X}^{(b)})\, V^{(b)}; \quad (31)$$

where $D$ is a matrix, which diagonal contains eigenvalues and $V$ is a matrix of eigenvectors (located in its columns).

In the next step, the eigenvalues are sorted in a decreasing order. The columns of $V$ are placed according to the eigenvalues order so the columns associated with the highest eigenvalues are placed in the beginning.

Finally we truncate respective matrices and get transformation matrices $F$ according to the following:

a)

$$\underset{k\in K}{\forall} V^{k}_{MN\times MN} \xrightarrow{\text{truncate}} F^{k}_{MN\times p}, \quad (32)$$

b)

$$\underset{k\in K}{\forall} V^{(k,r)}_{N\times N} \xrightarrow{\text{truncate}} F^{(k,r)}_{N\times p}, \quad (33)$$

$$\underset{k\in K}{\forall} V^{(k,c)}_{M\times M} \xrightarrow{\text{truncate}} F^{(k,c)}_{M\times q}, \quad (34)$$

c)

$$V_{MNK\times MNK} \xrightarrow{\text{truncate}} F_{MNK\times p} \quad (35)$$

d)

$$V^{(r)}_{N\times N} \xrightarrow{\text{truncate}} F^{(r)}_{N\times p} \quad (36)$$

$$V^{(c)}_{M\times M} \xrightarrow{\text{truncate}} F^{(c)}_{M\times q} \quad (37)$$

$$V^{(b)}_{K\times K} \xrightarrow{\text{truncate}} F^{(b)}_{K\times s} \quad (38)$$

In the tasks of pattern recognition or image compression, the actual dimensionality reduction is performed for each block extracted from the input image, producing its reduced representation $Y$, according to the following:

a)

$$\underset{k\in K}{\forall} Y^{k}_{p\times 1} = \left(F^{k}_{MN\times p}\right)^{T}\left(\hat{X}_{MN\times 1} - \overline{X}_{MN\times 1}\right), \quad (39)$$

b)

$$\underset{k\in K}{\forall} Y^{k}_{p\times q} = \left(F^{(k,c)}_{M\times p}\right)^{T}\left(\hat{X}^{k}_{M\times N} - \overline{X}^{k}_{M\times N}\right)F^{(k,r)}_{N\times q}, \quad (40)$$

c)

$$Y^{k}_{p\times 1} = \left(F^{k}_{MNK\times p}\right)^{T}\left(\hat{X}_{MNK\times 1} - \overline{X}_{MNK\times 1}\right), \quad (41)$$

d)

$$Y^{(r)}_{p\times MK} = \left(F^{(r)}_{N\times p}\right)^{T}\left(\hat{X}^{(r)}_{N\times MK} - \overline{X}^{(r)}_{N\times MK}\right), \quad (42)$$

$$Y^{(r)}_{p\times MK} \xrightarrow{\text{reorganize}} \hat{Y}^{(r)}_{M\times pK} \quad (43)$$

$$Y^{(r,c)}_{p\times qK} = \left(F^{(c)}_{M\times q}\right)^{T}\hat{Y}^{(r)}_{M\times pK}, \quad (43)$$

$$Y^{(r,c)}_{q\times pK} \xrightarrow{\text{reorganize}} \hat{Y}^{(r,c)}_{K\times pq}, \quad (44)$$

$$Y^{(r,c,b)}_{s\times pq} = \left(F^{(b)}_{K\times s}\right)^{T}\hat{Y}^{(r,c,b)}_{K\times pq}. \quad (45)$$

The operation “reorganize” changes the order of elements in the matrix in order to enable matrix multiplication.

It should be noticed that the reverse transformations involve the same multiplications with transposed matrices $F$.

In cases (a) and (b) each image is decomposed into $K$ slices and processed in an independent manner. In case (c) an input image is treated as one data structure, while in case (d) it is reorganized in such a way that three intermediate reduced representations are obtained, thus an output representation has dimensions $s \times pq$.

## 3. Experiments

The experimental part of the paper is related to the comparison of the results of image reconstruction based on the methods presented above. We employ each one of the described methods of covariance matrix calculation in the PCA reduction as a main part of simplified compression algorithm. Next we perform reverse transformation and calculate typical objective measure of image quality, namely Peak Signal-to-Noise Ratio (PSNR) [7]. The results of the compression experiments are presented on a sample image “Mandrill” (see Fig. 2), which is one of the de-facto standards in the digital image processing field, since its color- and texture-characteristics bring out any weaknesses of the analyzed algorithm.

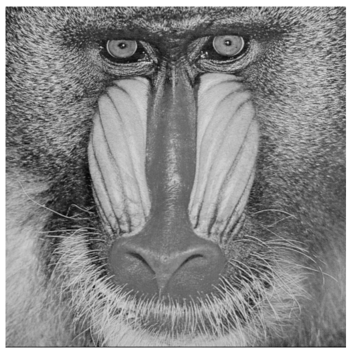

Fig. 2. Image “Mandrill” used for experiments
Rys. 2. Obraz „Mandrill” stosowany w eksperymentach

Let us assume, that the dimensions of the test image (represented in RGB color space) are 512 x 512 x 3, thus $M$=$N$=8 and $K$=3, giving 4096 three-dimensional image blocks. The dimensions of covariance matrices equal also to dimensions of matrices of eigenvalues and eigenvectors are presented in Table 1.

Tab. 1. Dimensions of covariance matrices for respective strategies
Tab. 1. Wymiary macierzy kowariancji dla poszczególnych strategii

| Case | Dim(Cov) |
|---|---|
| a | $3^* 64 \times 64$ |
| b | $3^* 8 \times 8$ |
| c | $192 \times 192$ |
| d | $3\times3 + 8\times8 + 8\times8$ |

The testing procedure is as follows. We decompose an input image into non-overlapping blocks, calculate respective covariance matrices, reduce each block and do the inverse transformation producing and output image. We calculate mean

Peak-Signal-to-Noise Ratio (PSNR) of output images in each channel to evaluate the reconstruction quality. We do the same operation for different reduction parameters: *p*, *q*, *s*.

The results of these experiments are presented below. In each case mean PSNR calculated for all RGB channels are presented. The Fig. 3 presents the mean reconstruction quality for one-dimensional reduction according to the *a* strategy. In comparison to the results presented in Fig. 4 (two-dimensional reduction) it gives higher quality for similar reduction rate. It should be noticed that for the *b* case, we set *p*=*q*.

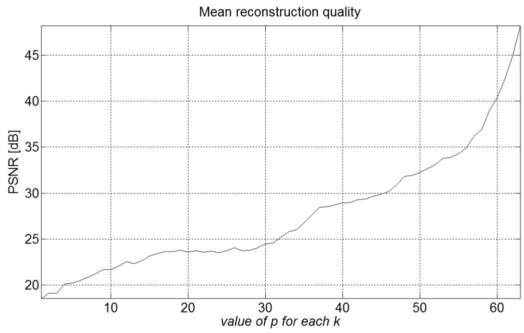


Fig. 3. Mean reconstruction error for different values of *p* – case (a)
Rys. 3. Średni błąd rekonstrukcji dla różnych wartości parametru *p* – przypadek (a)

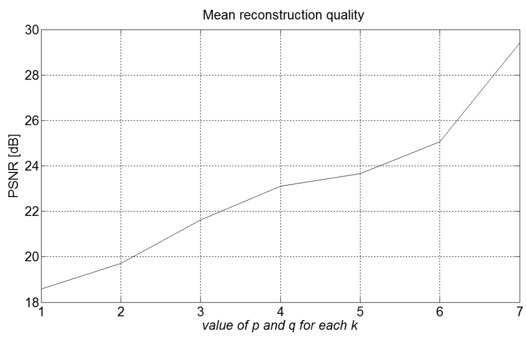


Fig. 4. Mean reconstruction error for different values of *p* and *q*– case (b)
Rys. 4. Średni błąd rekonstrukcji dla różnych wartości parametrów *p* i *q* – przypadek (b)

Fig. 5. presents a reduction involving concatenation of all elements in RGB matrix which gives the best reconstruction quality, however this approach involves the calculation of the biggest covariance matrix.

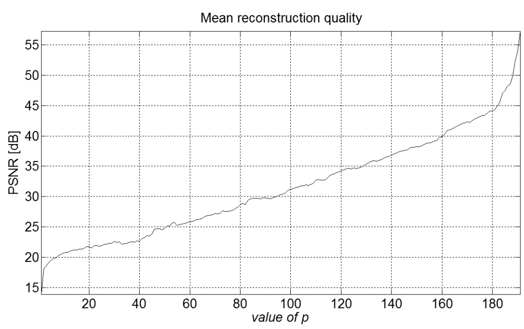


Fig. 5. Mean reconstruction error for different values of *p* – case (c)
Rys. 5. Średni błąd rekonstrukcji dla różnych wartości parametru *p* – przypadek (c)

Fig. 6. presents the results of three-dimensional reduction which gives acceptable quality for rather low compression. In this case the parameter *s* was left unmodified (*s*=*K*=3), since its lowering influenced the number of color-channels. In cases, where initial value of *K* is more than 3 its may give better results.

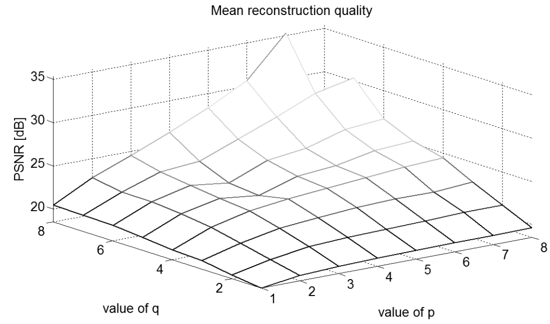


Fig. 6. Mean reconstruction error for different values of *p* and *q, s*=3– case (d)
Rys. 6. Średni błąd rekonstrukcji dla różnych wartości parametrów *p* i *q*, *s*=3 – przypadek (d)

## 4. Summary

In the paper we presented a problem of reducing dimensionality of data structured in three-dimensional matrices, like true-color digital images. Unlike the cases reported in the literature the processing is done by means of three-dimensional PCA on image blocks organized as three-dimensional structures (matrices). The eigenproblem is being solved on the covariance matrices giving a set of eigenvalues and eigenvectors, which are a base for creation of transformation matrices. The experiments showed that the reconstruction of input data is possible for all of the discussed strategies of data organization, giving the possibility to use it in the tasks of full-color image compression and recognition.

*The work was supported by finances of West Pomeranian Provincial Administration.*

## 5. References

[1] Zhao W., Chellappa R., Phillips P.J., Rosenfeld A.: Face recognition: a literature survey. ACM Computing Surveys, Vol. 35, No. 4, 399-458, 2003.
[2] Ahmed N., Rao K.R.: Orthogonal Transforms for Digital Signal Processing. New York, Springer Verlag, 2002.
[3] Wangmeng Zuo, Kuanquan Wang, Hongzhi Zhang: Subspace Methods for Face Recognition: Singularity, Regularization, and Robustness - State of the Art in Face Recognition, I-Tech, Vienna, Austria, January 2009.
[4] Turk M., Pentland A.: Eigenfaces for Recognition. Journal of Cognitive Neuroscience. Vol. 3 No.1, pp. 71-86, 1991.
[5] Reed I.S., Lan L.S.: A fast Karhunen Loeve transform (KLT) for data compression. J VCIR, 5, pp. 304-316, 1994.
[6] Partridge M., Calvo R.A.: Fast dimensionality reduction and simple PCA. Intelligent Data Analysis 2(3), 1998.
[7] Sayood K.: Introduction to Data Compression. Academic Press, 2000.
[8] Jolliffe I.T.: Principal Component Analysis. Springer Verlag, NY, 1986.
[9] Tsapatsoulis N., Alexopoulos V., Kollias S.: A Vector Based Approximation of KLT and Its Application to Face Recognition. Proc. of The IX European Signal Processing Conference EUSIPCO-98, Island of Rhodes, Greece, September 1998.
[10] Swets D.L., Weng J.: Using Discriminant Eigenfeatures for Image Retrieval. IEEE Trans. PAMI vol.18, 1996.
[11] Rychlik M., Stankiewicz W., Morzyński M.: Applications of 3D PCA Method For Extraction of Mean Shape and Geometrical Features of Biological Objects Set, Mathematical Modelling and Analysis, Vol. 13, Number 3, pages 413–420, 2008.
[12] Fritsch L., Pata P.: 3D KLT compression algorithm for camera security systems. Proceedings of SPIE, the International Society for Optical Engineering, Prague, Czech Republic, vol. 7138, pp. 713820.1-713820.6, 27-29 August 2008.
[13] Kukharev G., Forczmański P.: Data Dimensionality Reduction for Face Recognition Machine Graphics & Vision, vol.13, no. 1 /2, pp. 99-122, 2004.

*otrzymano / received: 14.07.2010*
*przyjęto do druku / accepted: 01.09.2010* *artykuł recenzowany*