

Image Sequence Decomposition Based on the Truncated Hierarchical SVD

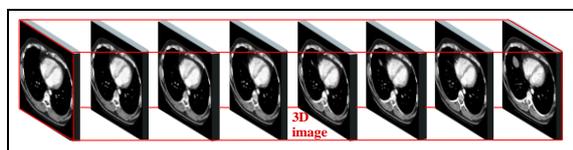
Roumen Kountchev¹, Roumiana Kountcheva²
 TU-Sofia¹ Bulgaria
 T&K Engineering², Bulgaria

Abstract

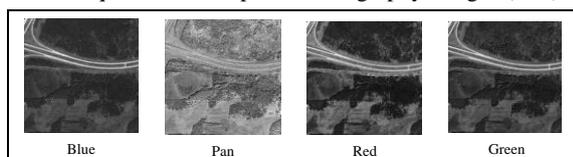
In this work is presented new algorithm, called Truncated Hierarchical SVD, which is the advance of the algorithm Radix₂² Hierarchical SVD, developed earlier by the authors of this paper. The new approach is aimed at the processing of sequences of correlated images, represented as 3rd-order tensors. It is based on the multiple execution of SVD for elementary tensors (ET) of size $2 \times 2 \times 2$, which build the tensor of size $N \times N \times N$, when $N=2^n$. The new approach is compared here to closest famous hierarchical SVD method for ET - the Sequential Unfolding SVD. The comparison proves, that the new algorithm has lower computational complexity. To reduce the computational complexity of the algorithm, here is used "truncation" of tensor components with small weights, based on new adaptive approach. As a result of the "truncation" and of the parallel implementation, the processing of image sequences, represented by 3rd order tensors, is significantly accelerated. These advantages open new abilities for real-time application in image processing systems.

1. Introduction

The decomposition of third-order tensors is one of the methods for representation and processing of multidimensional signals from various sources, such as: sequences of correlated 3D images: multispectral (MSI), multi-view, and computer tomography images (CTI), and different kinds of signals: radiolocation, hydro location, seismic, etc.



a. A sequence of Computer-Tomography Images (CTI)



b. A group of satellite Multi-Spectral Images (MSI)

Figure 1. Examples for two kinds of 3D images

Figures 1a and b show examples for two kinds of correlated 3D images: CTI and MSI. The tensor representation of such signals offers better abilities for efficient processing, compared to vector and matrix representations, because correlations between tensor elements are taken into account in three orthogonal directions, almost at same time. As a result of the tensor decomposition, each 3D image is represented as a sum of third-order tensors of same size, whose variances (respectively, weights) rapidly decrease. Each tensor in this sum is an outer product of 3 mutually orthogonal vectors. The aim of the tensor decomposition is to achieve full decorrelation of the sum components. Each tensor is defined by relatively small number of parameters, and from this it follows, that the general representation of the first several tensors in the decomposition, which have maximum weight, is significantly smaller than that of the decomposed tensor.

2. Related Work

2.1. Basic Tensor SVD Methods for Reduced Representation of Image Sequences

Decompositions of the kind Singular Value Decomposition (SVD) [1, 2, 3, 4] permit to reduce the tensor representation through cutting-off tensors with smallest weights, retaining minimum mean-square approximation error. To achieve maximum decorrelation of the sum components, different multilinear extensions of the matrix SVD are used, called Multilinear SVD (MSVD), or generalizations of the SVD matrix for higher-order tensors, called Higher-Order SVD (HOSVD) [5, 6, 7, 8]. Such are the famous methods: CANDECOMP/PARAFAC or Canonical Polyadic Decomposition (CPD), where the tensor is represented as a sum of rank-one tensors; the Tucker Decomposition (TD), which is a higher-order form of the Principal Component Analysis (PCA); the Kruskal decomposition [9], etc. The components of the tensor decomposition are calculated by using various iterative methods, whose iterations stop, when the pre-defined accuracy is achieved. Such are, for example: the tensor power iteration; the QR-factorization followed by the Householder transforms, or Gram-Schmidt process, Givens rotations; the Jacobi method; the Higher-Order Eigenvalue

Decomposition (HOEVD); the SVD calculation based on its relation to PCA, etc.

2.2. Famous Hierarchical Methods for Reduction of the Computational Complexity of the Tensor SVD

The tensor decomposition based on the use of iterative SVD methods needs significant number of computational operations. To overcome the problem, various hierarchical methods are already offered, based on the algorithms Hierarchical Tucker Decomposition (HTD) [10, 11], Sequentially Truncated HOSVD (ST-HOSVD) [12] and Sequential Unfolding SVD (SUSVD) [13].

One of the approaches aimed at the CPD computational complexity reduction is the Hierarchical Tucker Decomposition (HTD) or the Hierarchical SVD (HSVD) [10, 11]. The hierarchical structure of this decomposition [10] for 4th order tensor (\mathbf{B}_{1234}) is shown in Figure 2. Here \mathbf{B}_{12} and \mathbf{B}_{34} are second-order tensors; $[\mathbf{U}_1], [\mathbf{U}_2], [\mathbf{U}_3], [\mathbf{U}_4]$ - orthonormal matrices of various size, and r_1, r_k, r_{1k} - the ranks of the corresponding matricizations. This decomposition is based on the analysis of the approximation tensor error, which permits to achieve best approximation through cutting-off selected hierarchical singular values. The properties of HTD for tensors of order ≥ 2 are similar to those of the matrix SVD. The basic difference between HTD and TD is that because of the hierarchical tree-like structure used for the tensor representation, the needed memory volume is significantly reduced.

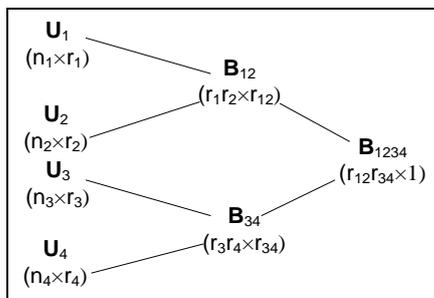


Figure 2. Example HTD for the 4th-order tensor \mathbf{B}_{1234} (4-dimensional body)

One more approach for CPD computational complexity reduction is based on the algorithm ST-HOSVD [12], illustrated by the example from Figure 3.

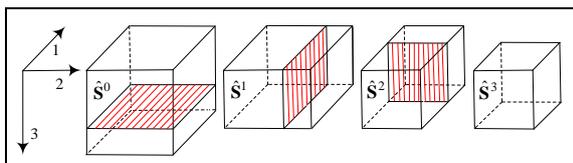
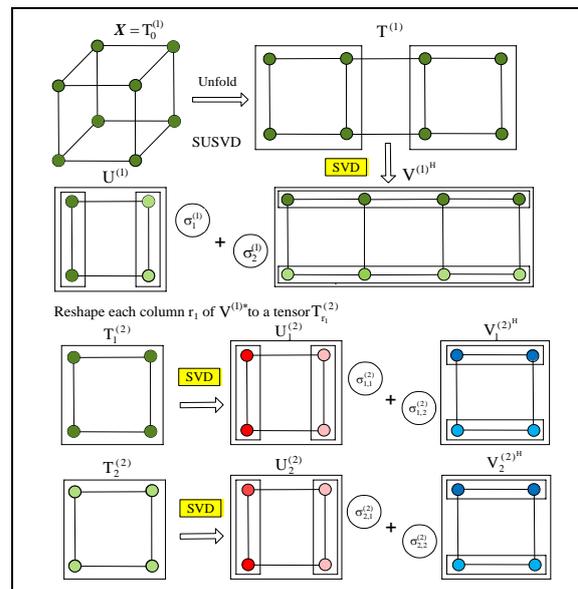


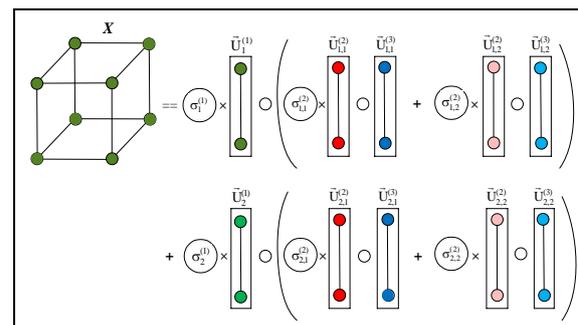
Figure 3. Example for truncation of the third-order tensor \mathbf{S} , by using the ST-HOSVD

The 3rd order tensor \mathbf{S} is approximated by the tensor $\hat{\mathbf{S}}^3$, obtained after consecutive cutting-off the components $\hat{\mathbf{S}}^0, \hat{\mathbf{S}}^1, \hat{\mathbf{S}}^2$ in its HOSVD decomposition in three mutually orthogonal directions. To achieve minimum mean-square approximation error for the tensor \mathbf{S} , all possible permutations (1,2,3; 1,3,2, etc.) should be executed, so that best truncation order in all three orthogonal directions (in this case, 3,2,1) to be defined.

Computational complexity reduction could be also achieved by using the hierarchical algorithm Sequential Unfolding SVD (SUSVD) [13], which is the closest to the Radix_{2x2}HSVD. The structure of SUSVD for ET of size $2 \times 2 \times 2$ is shown in Figures 4a and b.



a. Decomposition



b. Reconstruction

Figure 4. Structure of SUSVD algorithm for ET of size $2 \times 2 \times 2$

The decomposition is executed by consecutive applying SVD on three different matrices of size $4 \times 2, 2 \times 2$ and 2×2 , as shown in Figure 4a. Initially, for the tensor $\mathbf{X}_{2 \times 2 \times 2}$ of size $2 \times 2 \times 2$ is executed mode-1 unfolding, in result of which is calculated the matrix $[\mathbf{T}^{(1)}]$ of size 4×2 . For it is calculated SVD, and from

each column of the so obtained right-hand singular vectors are constituted the matrices $[T_1^{(2)}]$ and $[T_2^{(2)}]$ of size 2×2 , coloured in dark and light green. For each is calculated SVD and in result are got four matrices $[U_1^{(2)}]$, $[V_1^{(2)}]$, $[U_2^{(2)}]$, $[V_2^{(2)}]$ each of size 2×2 (their weights are given in the circles). The total number of parameters needed to calculate the full ET decomposition, is $14+10.2=34$ (in total - 14 in the first level, and 20 - in the second level).

The reconstruction of tensor $\mathbf{X}_{2 \times 2 \times 2}$ in accordance to Figure 4b starts by summing the outer products of the weighted couples of basic vectors, given in the brackets, obtained after the rearrangement of SVD matrix components for a block of size 2×2 ($SVD_{2 \times 2}$) in the second decomposition level, as shown in Figure 4a. The elements of the rearranged couples of vectors from Figure 4b are colored in red and blue, and pink and light blue, correspondingly. Each of the calculated two matrices is multiplied with the corresponding weighted vectors, whose elements are colored in green and light green. On the other hand, each matrix is a sum of two matrices of same size - the components of the corresponding $SVD_{2 \times 2}$. The outer product of the vector of size 1×2 with a matrix of size 2×2 (which is orthogonal for the vector), needs two outer products for a couple of orthogonal vectors of size 1×2 to be calculated. As a result, two tensors of rank $R=1$ are got, and each is the outer product of three basic vectors. In correspondence to Figure 4b the reconstructed tensor $\mathbf{X}_{2 \times 2 \times 2}$ is a sum of two tensors, which are the outer products of the couples of vectors of size 1×2 and of the matrices of size 2×2 .

In the presented approach the tensor decomposition is executed hierarchically from the top to the base of the tree-like structure, which represents the decomposition. The matrices calculated in the last level could be of various size, as shown in Figure 4a.

The objective of this work is the development of efficient, non-iterative algorithm, which to reduce the computational complexity of the tensor decomposition as a result of the neglecting of its low-informative components, in accordance to the pre-defined accuracy. This new approach for the calculation of tensors decomposition components is a new version of the algorithm Radix $_{2 \times 2}$ HSVD [14], called Truncated HSVD (THSVD).

The paper is structured as follows: in Section 3 is presented the decomposition of elementary tensor through THSVD; Section 4 is devoted to the THSVD algorithm for a tensor of size $N \times N \times N$; in Section 5 is evaluated the computational complexity of the new algorithm, Section 6 contains experimental results, and in Section 7 are given the conclusions.

3 Decomposition of Elementary Tensor through THSVD

The proposed THSVD algorithm is based on the decomposition of Elementary Tensors (ET) of size $2 \times 2 \times 2$ and is described in detail in the following subsections.

3.1. Decomposition of Elementary Tensor Based On the Radix $_{2 \times 2}$ HSVD

In accordance to Kountchev and Kountcheva [14], the structure of the Radix $_{2 \times 2}$ HSVD algorithm for the tensor $\mathbf{X}_{2 \times 2 \times 2}$ based on $SVD_{2 \times 2}$ and binary two-level tree, is shown in Figure 5.

In the *first level* of the algorithm, after mode-1 unfolding of the tensor $\mathbf{X}_{2 \times 2 \times 2}$, on the matrices $[X_1]$ and $[X_2]$ is applied the $SVD_{2 \times 2}$. As a result, are calculated matrices $[C_{11}]$ and $[C_{21}]$ which have high weights (σ_{11}, σ_{21}) and their elements are coloured in red, and matrices $[C_{12}]$ and $[C_{22}]$ whose weights (σ_{12}, σ_{22}) are small and their elements are coloured in blue. After rearrangement of the matrices in accordance to the lessening of their weights, the tensors $\mathbf{X}_{1(2 \times 2 \times 2)}$ and $\mathbf{X}_{2(2 \times 2 \times 2)}$ are reconstructed, and as a result is got the decomposition:

$$\mathbf{X}_{2 \times 2 \times 2} = \mathbf{X}_{1(2 \times 2 \times 2)} + \mathbf{X}_{2(2 \times 2 \times 2)}. \quad (1)$$

In the *second level* of the algorithm, on tensors $\mathbf{X}_{1(2 \times 2 \times 2)}$ and $\mathbf{X}_{2(2 \times 2 \times 2)}$ is applied mode-2 unfolding and are got the couples of matrices $[X_{11}]$, $[X_{12}]$ and $[X_{21}]$, $[X_{22}]$, for which are calculated corresponding $SVD_{2 \times 2}$. From $[X_{11}]$ and $[X_{12}]$ are obtained four matrices: $[C_{111}]$, $[C_{121}]$ - with high weights ($\sigma_{111}, \sigma_{121}$), and $[C_{112}]$, $[C_{122}]$ - with low weights ($\sigma_{112}, \sigma_{122}$). From $[X_{21}]$, $[X_{22}]$ are obtained other four matrices: $[C_{211}]$, $[C_{221}]$ - with high weights ($\sigma_{211}, \sigma_{221}$), and $[C_{212}]$, $[C_{222}]$ - with low weights ($\sigma_{212}, \sigma_{222}$). After rearrangement in accordance to their weights lessening, from these eight matrices $[C_{111}]$, $[C_{121}]$, $[C_{112}]$, $[C_{122}]$, $[C_{211}]$, $[C_{221}]$, $[C_{212}]$, $[C_{222}]$ are reconstructed four tensors $\mathbf{X}_1(1)$, $\mathbf{X}_1(2)$, $\mathbf{X}_2(1)$, $\mathbf{X}_2(2)$, each of size $2 \times 2 \times 2$. Then the decomposition is represented as follows:

$$\mathbf{X}_{2 \times 2 \times 2} = \mathbf{X}_1(1) + \mathbf{X}_1(2) + \mathbf{X}_2(1) + \mathbf{X}_2(2). \quad (2)$$

The first two tensors $\mathbf{X}_1(1)$, $\mathbf{X}_1(2)$, whose elements are coloured in pink in Figure 5, have high levels, while tensors $\mathbf{X}_2(1)$, $\mathbf{X}_2(2)$ whose elements are coloured in blue, have low weights. For the execution of the decomposition from Eq. (2) is needed 6-time calculation of $SVD_{2 \times 2}$.

3.2. Truncated HSVD for Elementary Tensor

To overcome the information redundancy problem, here is offered the algorithm Two-level Truncated HSVD_{2×2×2} (THSVD_{2×2×2}), which is based on the Radix_{2×2}HSVD_{2×2×2}. The structure of the new algorithm for the calculation of tensor $\mathbf{X}_{2 \times 2 \times 2}$ is shown in Figure 6. In the first decomposition level the algorithm is same as the one, already shown in Figure 5, and when all operations are finished, is got: $\mathbf{X}_{2 \times 2 \times 2} = \mathbf{X}_{1(2 \times 2 \times 2)} + \mathbf{X}_{2(2 \times 2 \times 2)}$. In the second level,

after the execution of operations similar to these from Figure 5, is obtained the *full decomposition* of the tensor $\mathbf{X}_{2 \times 2 \times 2}$, represented as a sum of four 1st-rank tensors, whose sequential numbers increase together with the lessening of their weights, i.e.:

$$\mathbf{X}_{2 \times 2 \times 2} = \sum_{r=1}^4 \mathbf{X}_{r(2 \times 2 \times 2)}, \quad (3)$$

where

$$\mathbf{X}_{1(2 \times 2 \times 2)} \Rightarrow \text{fold}_{\text{mod } e-2} ([C_{111}], [C_{121}]);$$

$$\mathbf{X}_{2(2 \times 2 \times 2)} \Rightarrow \text{fold}_{\text{mod } e-2} ([C_{211}] \cup [C_{112}], [C_{221}] \cup [C_{122}]); \quad (4)$$

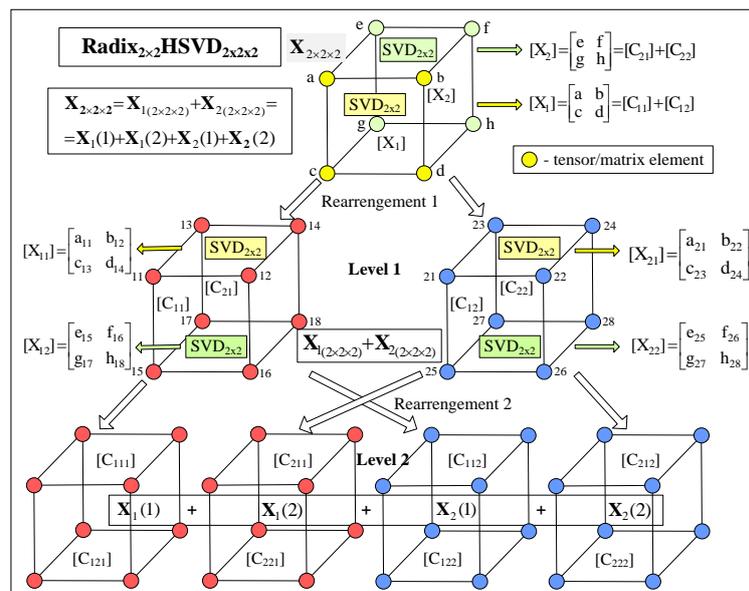


Figure 5. Structure of the two-level Radix_{2×2} HSVD_{2×2×2} for the tensor $\mathbf{X}_{2 \times 2 \times 2}$

$$\mathbf{X}_{3(2 \times 2 \times 2)} \Rightarrow \text{fold}_{\text{mod } e-2} ([C_{112}] \cup [C_{211}], [C_{122}] \cup [C_{221}]);$$

$$\mathbf{X}_{4(2 \times 2 \times 2)} \Rightarrow \text{fold}_{\text{mod } e-2} ([C_{212}], [C_{222}]). \quad (5)$$

The tensors $\mathbf{X}_{1(2 \times 2 \times 2)}$ and $\mathbf{X}_{4(2 \times 2 \times 2)}$ are always reconstructed on the basis of the couples of matrices $[C_{111}], [C_{121}]$ and $[C_{212}], [C_{222}]$, respectively. The tensor $\mathbf{X}_{2(2 \times 2 \times 2)}$ is reconstructed by using the couples of matrices $[C_{211}]$ and $[C_{221}]$, or respectively $[C_{112}]$ and $[C_{122}]$, and the tensor $\mathbf{X}_{3(2 \times 2 \times 2)}$ - by using the couples of matrices $[C_{112}], [C_{122}]$ or respectively - $[C_{211}], [C_{221}]$. The choice of the couple of matrices is defined by the selected truncation: mode-1 or mode-2. In the decomposition from Eq. (3) are retained two tensors only: $\mathbf{X}_{1(2 \times 2 \times 2)}$ and $\mathbf{X}_{2(2 \times 2 \times 2)}$. For their reconstruction, from the calculated eight matrices in the second level, are used the four matrices of highest weights only.

For a given approximation accuracy ϵ of the tensor $\mathbf{X}_{2 \times 2 \times 2}$, the remaining two tensors $\mathbf{X}_{3(2 \times 2 \times 2)}$ and $\mathbf{X}_{4(2 \times 2 \times 2)}$ in Eq. (3) could be neglected, and in result is got:

$$\mathbf{X}_{2 \times 2 \times 2} \approx \hat{\mathbf{X}}_{2 \times 2 \times 2} = \sum_{r=1}^2 \mathbf{X}_{r(2 \times 2 \times 2)} \text{ for } \|\mathbf{X}_{2 \times 2 \times 2} - \hat{\mathbf{X}}_{2 \times 2 \times 2}\| = \left\| \sum_{r=3}^4 \mathbf{X}_{r(2 \times 2 \times 2)} \right\| \leq \epsilon. \quad (6)$$

The decomposition of the tensor $\mathbf{X}_{2 \times 2 \times 2}$ in correspondence to Eq. (6) represents the THSVD_{2×2×2} algorithm. In the second decomposition level the SVD_{2×2} is executed four times, and each SVD_{2×2} is defined by four parameters only (three - for the first matrix component $[C_1]$, and one - for the second $[C_2]$). Each of the tensors from Eq. (3) contains the components of two selected SVD_{2×2} (shown in Figure 6 colored in red and blue). Then, the total number of parameters needed for the description of all four tensors from Eq. (3), is $4 \times 4 = 16$. For the "truncated" decomposition represented by Eq. (6), which contains the tensors $\mathbf{X}_{1(2 \times 2 \times 2)}$ and $\mathbf{X}_{2(2 \times 2 \times 2)}$ only, the total number of needed parameters is 20 or 14, depending on the kind of truncation, chosen for the neglecting of the components of low weights of the selected four SVD_{2×2}.

In the *first way*, for *truncation* (mode-1), the second retained tensor $\mathbf{X}_{2(2 \times 2 \times 2)}$ is defined by the matrices $[C_{211}]$ and $[C_{221}]$, and the first "truncated" tensor $\mathbf{X}_{3(2 \times 2 \times 2)}$ - by the matrices $[C_{112}]$ and $[C_{122}]$. In this case, the "truncated" decomposition is represented by 20 parameters.

In the *second way*, for *truncation* (mode-2), the second retained tensor $\mathbf{X}_{2(2 \times 2 \times 2)}$ is defined by the matrices $[C_{112}]$ and $[C_{122}]$, and the first "truncated" tensor $\mathbf{X}_{3(2 \times 2 \times 2)}$ - by the matrices $[C_{211}]$ and $[C_{221}]$. The tensor $\mathbf{X}_{4(2 \times 2 \times 2)}$ is cut-off in both cases (mode-1 and mode-2). Hence, for the THSVD mode-2 are needed $2 \times 3 + 2 \times 4 = 14$ parameters in total, because in the first decomposition level is calculated the first matrix component of each $SVD_{2 \times 2}$ only. Then, the total number of parameters, needed for the ET decomposition based on the THSVD $_{2 \times 2 \times 2}$ truncation mode-2 (equal to 14) is the smallest, when compared to algorithms SUSVD (with 34 parameters needed) and Radix $_{2 \times 2}HSVD_{2 \times 2 \times 2}$ (with 24 parameters needed).

For the execution of THSVD $_{2 \times 2 \times 2}$ truncation mode-2, in the second level have to be calculated only the two $SVD_{2 \times 2}$, colored in red in Figure 6.

The reconstruction of tensors $\mathbf{X}_{1(2 \times 2 \times 2)}$ and $\mathbf{X}_{2(2 \times 2 \times 2)}$ depends on the truncation mode (1 or 2), which is applied on the eight matrices in the second which is applied on the eight matrices in the second decomposition level.

The "truncation" could be fixed, or adaptive. The adaptive truncation criterion is defined on the basis of the total weight of the matrices, used for the reconstruction of tensor \mathbf{X}_2 , when compared to that of the matrices used for the tensor \mathbf{X}_3 . The *adaptive truncation rule* is defined by the relation:

$$\text{Truncation mode} \begin{cases} 1, & \text{if } (\sigma_{211} + \sigma_{221}) > (\sigma_{112} + \sigma_{122}), \\ 2, & \text{if } (\sigma_{211} + \sigma_{221}) \leq (\sigma_{112} + \sigma_{122}). \end{cases} \quad (7)$$

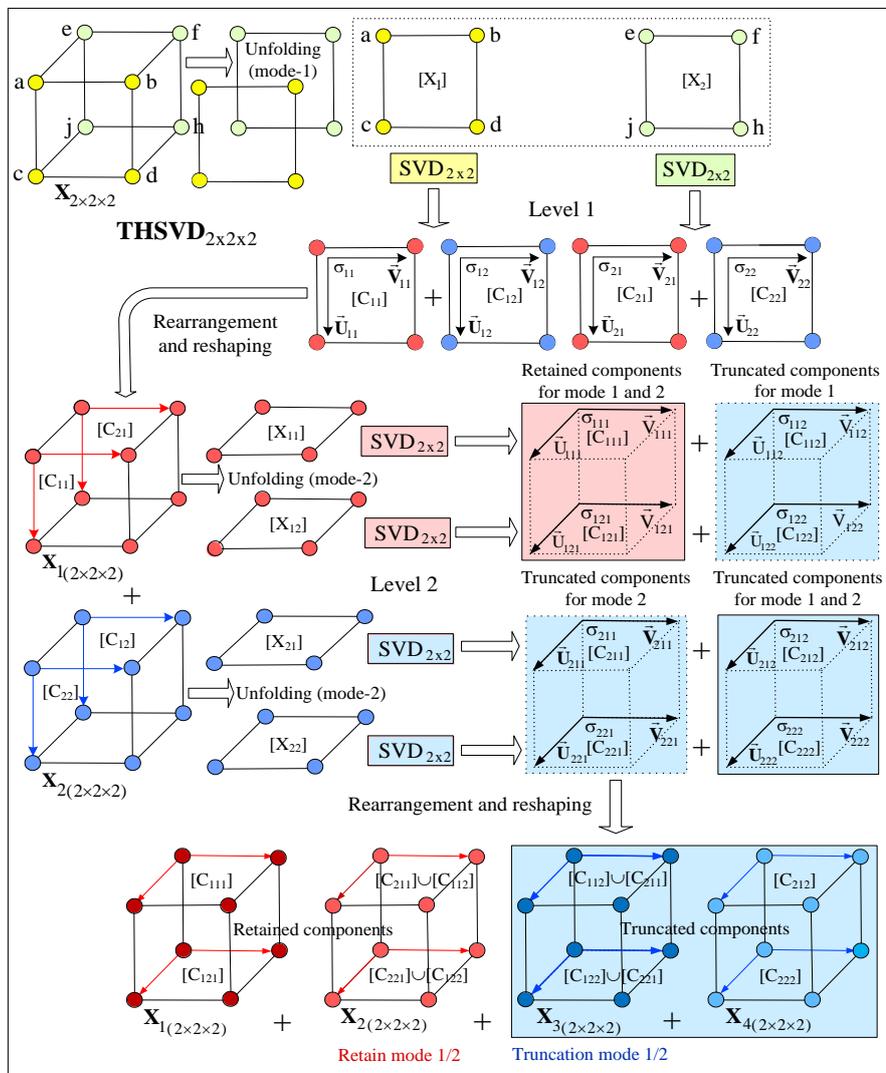


Figure 6. Structure of the Adaptive THSVD algorithm for the tensor $\mathbf{X}_{2 \times 2 \times 2}$, based on $SVD_{2 \times 2}$.

Here $(\sigma_{211} + \sigma_{221})$ is the sum of the weights of the couple of matrix components in the second level $[C_{211}]$ and $[C_{221}]$, and $(\sigma_{112} + \sigma_{122})$ - the sum of the weights for the couple of matrices $[C_{112}]$ and $[C_{122}]$. In the case, when the ET represents a sequence of two images of size 2×2 , its elements (pixels) have non-negative values, and hence, its rank is $R=2$.

The basic operation in the calculation of the Radix 2×2 HSVD $2 \times 2 \times 2$ is the matrix SVD 2×2 , given in detail in the next sub-section.

3.3. Calculation of the Matrix SVD for Image Block of Size 2×2

The Matrix SVD 2×2 could be represented in two basic forms: algebraic and trigonometric.

- Algebraic form:

$$[X] = \begin{bmatrix} a & b \\ c & d \end{bmatrix} = \sigma_1(\bar{U}_1 \circ \bar{V}_1) + \sigma_2(\bar{U}_2 \circ \bar{V}_2) = [C_1] + [C_2] = \frac{1}{2A} \left\{ \sigma_1 \begin{bmatrix} \sqrt{rp} & \sqrt{sp} \\ \sqrt{rq} & \sqrt{sq} \end{bmatrix} + \sigma_2 \begin{bmatrix} \sqrt{sq} & -\sqrt{rq} \\ -\sqrt{sp} & \sqrt{rp} \end{bmatrix} \right\}, \quad (8)$$

where a, b, c, d are the elements of the matrix $[X]$, and $(\bar{U}_r \circ \bar{V}_r)$ is the outer product of vectors \bar{U}_r and \bar{V}_r ;

$$\sigma_1 = \sqrt{\frac{\omega + A}{2}}, \quad \bar{U}_1 = \frac{1}{\sqrt{2A}} \begin{bmatrix} \sqrt{r} \\ \sqrt{q} \end{bmatrix}; \quad \bar{U}_2 = \frac{1}{\sqrt{2A}} \begin{bmatrix} -\sqrt{q} \\ \sqrt{r} \end{bmatrix}; \quad (9)$$

$$\sigma_2 = \sqrt{\frac{\omega - A}{2}}, \quad \bar{V}_1 = \frac{1}{\sqrt{2A}} \begin{bmatrix} \sqrt{p} \\ \sqrt{s} \end{bmatrix}; \quad \bar{V}_2 = \frac{1}{\sqrt{2A}} \begin{bmatrix} -\sqrt{s} \\ \sqrt{p} \end{bmatrix};$$

$$\omega = a^2 + b^2 + c^2 + d^2; \quad r = A + v; \quad q = A - \mu;$$

$$p = A + \mu; \quad s = A - v; \quad A = \sqrt{v^2 + 4\eta^2}; \quad (10)$$

$$v = a^2 + c^2 - b^2 - d^2; \quad \mu = a^2 + b^2 - c^2 - d^2; \quad \eta = ab + cd. \quad (11)$$

Here σ_1 and σ_2 are the singular values of the matrix $[X]$, where $\sigma_1 \geq \sigma_2 > 0$; \bar{U}_1, \bar{U}_2 and \bar{V}_1, \bar{V}_2 are the left and right-eigen vectors of the matrix $[X]$ for which $[X][X]^t = \lambda_s \bar{U}_s$; $[X]^t[X] = \lambda_s \bar{V}_s$ when $s=1,2$; $\lambda_s = \sigma_s^2$ - the eigenvalues of the symmetrical matrices $[X][X]^t$ and $[X]^t[X]$.

The elements of matrices $[C_1]$ and $[C_2]$ are defined by four parameters only (ω, v, μ, η) , and from this it follows that the decomposition represented by Eq. (8) is not "overcomplete".

- Trigonometric form:

$$[X] = \begin{bmatrix} a & b \\ c & d \end{bmatrix} = [C_1] + [C_2] = \sigma_1 \begin{bmatrix} c_1 c_2 & c_1 s_2 \\ s_1 c_2 & s_1 s_2 \end{bmatrix} + \sigma_2 \begin{bmatrix} s_1 s_2 & -s_1 c_2 \\ -c_1 s_2 & c_1 c_2 \end{bmatrix}, \quad (12)$$

where

$$s_1 = \sin \theta_1, \quad c_1 = \cos \theta_1, \quad s_2 = \sin \theta_2, \quad c_2 = \cos \theta_2.$$

Correspondingly:

$$\theta_1 = \frac{1}{2} \arctg\left(\frac{2\eta}{v}\right), \quad \theta_2 = \frac{1}{2} \arctg\left(\frac{2\delta}{\mu}\right), \quad \delta = ac + bd. \quad (13)$$

Here $\sigma_1, \sigma_2, \eta, v, \mu$ are calculated in correspondence to Eqs. (9), (10), (11). As it follows from Eq. (12), the number of parameters needed for to calculate SVD 2×2 , is four $(\sigma_1, \sigma_2, \theta_1, \theta_2)$, i.e. in this case the decomposition is also not "overcomplete".

3.4. Fast Calculation of the Matrix SVD 2×2

The trigonometric representation of the matrix SVD 2×2 permits to accelerate the calculations based on the approximation of parameters $s_1, c_1, s_2, c_2, \sigma_1, \sigma_2$, when the distributed arithmetic (DA) principle is used [15]. In this case, the parameters are represented as code words of fixed number of bits. As a result, the computational complexity of SVD 2×2 becomes lower, retaining high approximation accuracy, which depends on the code words length. Each parameter s_1, c_1, s_2, c_2 could be represented as a K-bit binary word by corresponding approximation of $\hat{s}_1, \hat{c}_1, \hat{s}_2, \hat{c}_2$:

$$s_1 \approx \hat{s}_1 = -s_{1,0} + \sum_{k=1}^{K-1} s_{1,k} 2^{-k}, \quad c_1 \approx \hat{c}_1 = c_{1,0} + \sum_{k=1}^{K-1} c_{1,k} 2^{-k},$$

$$s_2 \approx \hat{s}_2 = -s_{2,0} + \sum_{k=1}^{K-1} s_{2,k} 2^{-k}, \quad c_2 \approx \hat{c}_2 = -c_{2,0} + \sum_{k=1}^{K-1} c_{2,k} 2^{-k}, \quad (14)$$

where

$$s_{1,k}, c_{1,k} \in \{0, \pm 1\}; \quad s_{2,k}, c_{2,k} \in \{0, \pm 1\}, \quad \text{for } k=0, 1, 2, \dots, K-1.$$

Here $s_{1,0}, c_{1,0}, s_{2,0}, c_{2,0}$ are the bits, which define the sign (+ or -) of the corresponding binary number. The value of K defines the accuracy of the corresponding parameter. In this case:

$$[X] \approx [\hat{X}] = \hat{\sigma}_1 \begin{bmatrix} m & n \\ g & h \end{bmatrix} + \hat{\sigma}_2 \begin{bmatrix} h & -g \\ -n & m \end{bmatrix}, \quad (15)$$

where $m = \hat{c}_1 \hat{c}_2$; $n = \hat{c}_1 \hat{s}_2$; $g = \hat{s}_1 \hat{c}_2$; $h = \hat{s}_1 \hat{s}_2$. Each product m, n, g, h is a function of the couple of variables - the angles θ_1 and θ_2 , defined in the range between 0 and $\pm 2\pi$. After discretization, they could be represented as follows:

$$\theta_1(i) = \pm \pi i / 2^{K-1}, \quad \theta_2(j) = \pm \pi j / 2^{K-1} \quad \text{for } i, j = 0, 1, 2, \dots, 2^K - 1 \quad (16)$$

For each possible couple of values $\theta_1(i), \theta_2(j)$ could be calculated the product m, n, g, h. The so obtained values could be saved in corresponding 2D tables of 2^K rows and 2^K columns. The data from the tables $m[\theta_1(i), \theta_2(j)], n[\theta_1(i), \theta_2(j)], g[\theta_1(i), \theta_2(j)], h[\theta_1(i), \theta_2(j)]$ for each couple $\theta_1(i), \theta_2(j)$ ($i, j = 0, 1, 2, \dots, 2^K - 1$) are calculated in advance, and saved in a special memory, 2D Look-Up-Table (2D-LUT). To access the content of the cell (i, j) in this memory,

is used the address, defined by $\theta_1(i), \theta_2(j)$. Each 2D-LUT memory should have 2^{2K} cells. For example, if $K=8$, the 2D-LUT should be of $2^{16}=64$ KB cells, where the values of the products m, n, g, h are saved as 8-bit words.

As a result, the total number of multiplications needed for the execution of the $SVD_{2 \times 2}$ in correspondence to Eq. (15) are reduced by four, and each is replaced by the operation “reading” from the 2D-LUT. In this case, the memory volume is increased, but the calculation of each ET is accelerated, because the total number of operations is reduced by $6 \times 4 = 24$.

4. Decomposition of Tensor of Size $N \times N \times N$ through THSVD

To overcome the problem with the high computational complexity of THSVD tensor decomposition for a tensor of size $N \times N \times N$ (when $N=2^n$), here is offered inverse processing strategy - from the base, to the top. This means that the decomposition of the matrices obtained in result of tensor matricization, are used for the reconstruction of corresponding tensors, which are the decomposition components. At first, the algorithm THSVD is represented for the decomposition of tensor of size $4 \times 4 \times 4$, and after that it is generalized for a tensor of size $N \times N \times N$ ($N=2^n$).

4.1. Radix₂ THSVD Calculation for a Tensor of Size $4 \times 4 \times 4$

In the *first level* of the THSVD_{4×4×4} algorithm the tensor $T_{4 \times 4 \times 4}$ (for $N=4$), shown in Figure 7, is divided into eight ETs $T_{i(2 \times 2 \times 2)}$ for $i=1,2,\dots,8$ (cubes of size $2 \times 2 \times 2$). Each ET is decomposed into 2 new tensors $T_{i,j(2 \times 2 \times 2)}$ for $j=1,2$ of same size. In the 1st level of THSVD_{4×4×4}, for each group, composed of 8 elements of same color (yellow, red, green, blue, white, purple, light blue, or orange), is calculated THSVD_{2×2×2}. After rearrangement of all ETs and their reshaping, are obtained new 2 tensors $T_{j(4 \times 4 \times 4)}$, shown in Figure 8 in pink and light blue.

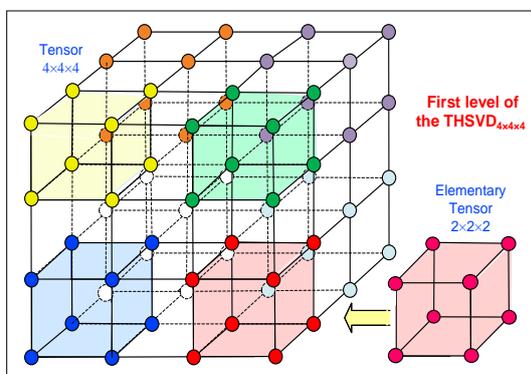


Figure 7. The tensor $T_{4 \times 4 \times 4}$ is divided into eight ETs $T_{i(2 \times 2 \times 2)}$ for $i=1,2,\dots,8$ in the first THSVD_{4×4×4} level, where the

THSVD_{2×2×2} is applied on each group of elements of same color (8 in total).

In the *second level* of the THSVD_{4×4×4} algorithm, each tensor $T_{j(4 \times 4 \times 4)}$ is divided into 8 sub-tensors $T_{j,k(2 \times 2 \times 2)}$ for $k=1,2$ in the way, defined by the spatial net for elements interlacing as shown in Figure 8. The color of the elements in each cube corresponds to that from the first level of the THSVD_{4×4×4} algorithm. For each expanded elementary tensor (double-size tensor) shown in Figure 8, is calculated once again the THSVD_{2×2×2} algorithm. After execution of the 1st decomposition level, the tensor $T_{4 \times 4 \times 4}$ is approximated by a sum of two components:

$$T_{4 \times 4 \times 4} \approx \sum_{j=1}^2 T_{j(4 \times 4 \times 4)} \quad (17)$$

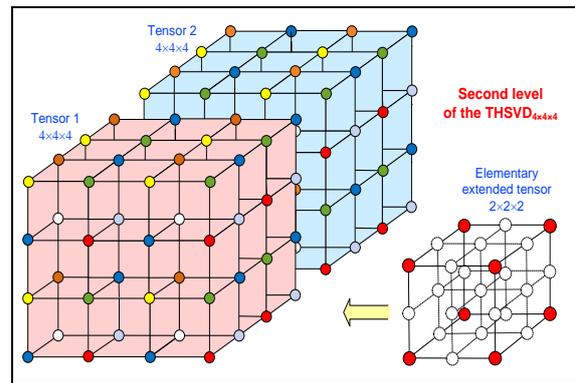


Figure 8. Division of tensors $T_{j(4 \times 4 \times 4)}$ into extended ETs $T_{i,j(2 \times 2 \times 2)}$ in the second THSVD_{4×4×4} level, where the THSVD_{2×2×2} is applied on each group of elements of same color (8 in total)

The so calculated two tensors $T_{j(4 \times 4 \times 4)}$ are arranged in correspondence to the lessening of the mean singular values (the energy) of the elementary tensors $T_{j,k(2 \times 2 \times 2)}$ for $j=1,2$ and $k=1,2,\dots,8$. The tensors $T_{j(4 \times 4 \times 4)}$ are rearranged in accordance to energy decrease of the ETs $T_{j,k(2 \times 2 \times 2)}$, which build them.

In accordance to Figure 8, on each ET is applied THSVD_{2×2×2} again. After the execution of the second decomposition level, the tensor $T_{4 \times 4 \times 4}$ is approximated by a sum of four 3rd order tensors:

$$T_{4 \times 4 \times 4} \approx \sum_{j_1=1}^2 \sum_{j_2=1}^2 T_{j_1, j_2(4 \times 4 \times 4)} \quad (18)$$

The computational graph of the 2-level THSVD_{4×4×4} decomposition is shown in Figure 9. In the initial decomposition level for each ET is executed unfolding mode-1, and in the next level - unfolding mode-2. The so calculated two tensors $T_{j(4 \times 4 \times 4)}$ are arranged in accordance to the decreasing singular values of kernels $T_{j,k(2 \times 2 \times 2)}$, which compose them, for $j=1,2$ and $k=1,2,\dots,8$. If the possible truncations are not taken into account for the low-energy branches of the computational graph (colored in blue in Figure 9) in

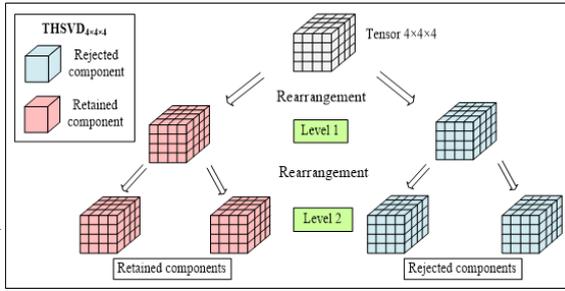


Figure 9. Structures of the full and truncated 2-level binary tree computational graphs for the 2-level THSVD_{4×4×4} algorithm based on the HSVD_{2×2×2}

the 1st level of the full THSVD_{4×4×4}, the SVD_{2×2} is executed 32 times, and in the 2nd level - 64 times.

4.2. THSVD Calculation for Third-Order Tensor of Size N×N×N

The decomposition of tensors $T_{4×4×4}$ could be generalized for the case, when the tensor $T_{N×N×N}$ is of size N×N×N for N=2ⁿ. As a result of the use of the THSVD_{N×N×N} full algorithm and for the pre-defined approximation accuracy ϵ , the tensor $T_{2^n \times 2^n \times 2^n}$ is represented as a sum of N² singular tensors:

$$T_{2^n \times 2^n \times 2^n} \approx \sum_{j_1=1}^2 \sum_{j_2=1}^2 \dots \sum_{j_n=1}^2 T_{j_1, j_2, \dots, j_n} (2^n \times 2^n \times 2^n). \quad (19)$$

The singular tensors $T_{j_1, j_2, \dots, j_n} (2^n \times 2^n \times 2^n)$ of size 2ⁿ×2ⁿ×2ⁿ are arranged in accordance to the decreasing energies of the tensors, which build them. The maximum number of hierarchical levels needed for the execution of the THSVD_{N×N×N} algorithm for N=2ⁿ, is n. The number of retained singular tensors in the decomposition is reduced at least two times, compared to HSVD_{N×N×N}, and in the last level this number is 2ⁿ. In case, that the rank of the tensor $T_{2^n \times 2^n \times 2^n}$ is R=2^r ≤ N=2ⁿ, Eq. (19) becomes:

$$T_{2^n \times 2^n \times 2^n} \approx \sum_{j_1=1}^2 \sum_{j_2=1}^2 \dots \sum_{j_r=1}^2 T_{j_1, j_2, \dots, j_r} (2^n \times 2^n \times 2^n) \text{ for } r = \log_2 R \quad (20)$$

In this case, the decomposition comprises R singular tensors. In the first level of THSVD_{2×2×2} for each ET is applied „unfolding mode-1“, as shown in Figure 6. In the next levels of THSVD_{N×N×N} are alternatively applied „unfolding mode-1,2,3 “for level 1 of ET. As a result, the elements of the decomposition tensors are mutually decorrelated in three orthogonal directions.

5. Evaluation of the Computational Complexity of THSVD

Unlike the Multi-linear SVD, THSVD does not require iterative calculations. The number of operations “addition” and “multiplication” O(*) needed to calculate the SVD_{2×2} (in correspondence to Eqs. (10-12) and the results in sub-section 3.3), is:

$$O_{SVD}(2 \times 2) = 40. \quad (21)$$

From Figure 6 it follows that

$$O_{TSVD}(2 \times 2 \times 2) = 40 \cdot 4 = 160. \quad (22)$$

Then, taking into account the computational structure, shown in Figure 9, we get the relations:

$$O_{THSVD}(2^2 \times 2^2 \times 2^2) = 2.8(1+2) \times O_{THSVD}(2 \times 2 \times 2) = 16 \times 3 \times 160. \quad (23)$$

$$O_{THSVD}(2^3 \times 2^3 \times 2^3) = 2^2 \cdot 8(1+2+4) \times O_{THSVD}(2 \times 2 \times 2) = 32 \times 7 \times 160. \quad (24)$$

The relations (23) and (24) could be generalized for a tensor of size 2ⁿ×2ⁿ×2ⁿ. In this case, the computational complexity of the full THSVD algorithm, represented by the number of needed arithmetic operations, could be evaluated in accordance to the relation:

$$O_{THSVD}(2^n \times 2^n \times 2^n) = 2^{n-1} 8 \left(1 + \sum_{i=1}^{n-1} 2^i\right) \times O_{THSVD}(2 \times 2 \times 2) = 2^{n+2} (2^{n-1} - 1) \times 160 \approx O(2^{2n+1}). \quad (25)$$

In case, that the tensor $T_{2^n \times 2^n \times 2^n}$ is of rank R, in accordance to Eq. (20) is calculated O(2R²). For comparison, the complexity to compute the Hierarchical SVD for H-Tucker tensors of constant rank R, mode size s and order d, is O(dsR²+(d-1)R⁴) [11]. Then, for 3rd-order tensor with hierarchical rank R and s=3 the complexity is O(9R²+2R⁴). In this case, the relative CC of the THSVD for a 3rd-order tensor is:

$$\psi(R) = \frac{O_{H-Tucker}}{O_{THSVD}} = \frac{O(9R^2 + 2R^4)}{O(2R^2)} \approx \frac{O(R^4)}{O(R^2)}. \quad (26)$$

Hence, the computational complexity of the THSVD is ≈ R² times lower than that of the H-Tucker tensor decomposition. However, the new decomposition has to use larger memory, which comprises O(2R) cells at least. According to equations (6) and (20), the ET decomposition is repeatedly executed in each hierarchical level. This permits to implement the decomposition by using similar sets of calculations, executed in parallel.

6. Experimental Results

For the evaluation of the proposed new method were executed some experiments for face recognition, based on the TensorFace method [16]. The method requires a training set of faces which to be compared to new faces. There are some obligatory preliminary operations: 1) all images in the training set are converted to grayscale; 2) the images are cropped so that to contain the face of the subject only; 3) the processed images are of same size.

Each image in the training set is represented as an $N \times M$ matrix, where N is the length of the image in vector format and M is the total number of images in the set. The main problem is, that image dimensionality is high: even for a small image of size 200×200 pixels, the dimensions are 40,000. One of the famous methods used to overcome the problem is the PCA which permits to retain the features that best represent the image. After the PCA execution, the processed images are represented by much smaller number of parameters and the recognition can start. Any unknown face is represented as a linear combination of the TensorFace in the training set, with a vector of coefficients that correspond to the weights of each TensorFace, and the distance of the unknown face to its nearest face in the training set is evaluated: if it is lower than a pre-defined threshold, the face is positively identified; in case that the distance is higher than the threshold - the face is classified as unknown.

Some test results are given below. In Table 1 is shown the recognition rate for different number of training and testing images using SVD with Hidden Markov Model (SVD-HMM) [17], and for THSVD.

In Table 1 is given the recognition rate for same training and testing images, but using THSVD. The table shows that the rate is increased and reached 100% along with changing of the number of training and testing images [16]. The 100% recognition rate follows from the use of THSVD instead of 2D SVD.

Table 1. Recognition rate using SVD-HMM for different number of training and testing images - YALE database

Number of images		Recognition rate (%) SVD-HMM	Recognition rate (%) THSVD
for training	for testing		
75	75	82.7%	89.45%
90	60	80%	97.9%
105	45	73.3%	98.43%
120	30	90%	100%
135	15	100%	100%

Figure 10 shows 38 subjects from the YALE dataset [18]. For the experiments, 20 images for each subject were used.



Figure 10. Facial images of YALE database used for the experiments



Figure 11. A group of 15 TensorFace basic vectors from the THSVD tensor decomposition using the YALE dataset

Figure 11 shows fifteen TensorFace basic vectors from the THSVD tensor decomposition from same dataset.

7. Conclusions

The comparison of algorithms "THSVD truncation mode-2" and "SUSVD" for a tensor of size $2 \times 2 \times 2$ shows, that the first requires the calculation of 14 parameters only, while for the second this number is 34. From this it follows that the total number of parameters needed for the calculation of THSVD is more than twice lower than that, needed for SUSVD. This also results in computational complexity reduction of THSVD compared to SUSVD for a tensor of size $N \times N \times N$.

One more advantage of THSVD is the need of multiple calculation of $SVD_{2 \times 2}$ only in each hierarchical level. This permits parallel implementation of THSVD based on same processors of relatively simple structure designed to calculate $SVD_{2 \times 2}$. Compared to HSVD, the computational complexity of THSVD for the preset accuracy is about R^2 times lower. These advantages of the new algorithm open new possibilities for its application in real-time systems working in the following areas: processing and compression of sequences of correlated images, digital watermarking, computer vision, machine learning, processing of multidimensional signals, etc. The future wider simulations and experimental investigations in various application areas will permit more accurate evaluation of its efficiency.

References

- [1] M. Moonen and B. de Moor, SVD and Signal Processing III, Elsevier Science B.V., Amsterdam, The Netherlands, 1995.
- [2] I. Jolliffe, Principal Component Analysis, 2nd ed., Springer-Verlag, New York, 2002.
- [3] S. Orfanidis, SVD, PCA, KLT, CCA, and all that, Rutgers University Electrical & Computer Eng. Dept., Optimum Signal Processing, 2007, pp. 1-77.
- [4] P. Fieguth, Statistical Image Processing and Multidimensional Modeling, Springer, Science + Business Media, 2011.
- [5] H. Lu, K. Plataniotis, A. Venetsanopoulos. MPCA: Multilinear Principal Component Analysis of Tensor Objects, IEEE Transactions on Neural Networks, Vol. 19, No. 1, 2008, pp. 18-39.
- [6] G. Bergqvist, E. Larsson, The Higher-Order Singular Value Decomposition: Theory and an Application, IEEE Signal Processing Magazine, Vol. 27, No. 3, 2010, pp. 151-154.
- [7] T. Kolda, B. Bader, Tensor decompositions and applications, SIAM Review, Vol. 51, No. 3, 2009, pp. 455–500.
- [8] A. Cichocki, D. Mandic, A. Phan, C. Caiafa, G. Zhou, Q. Zhao, L. De Lathauwer, Tensor Decompositions for Signal Processing Applications, IEEE Signal Processing Magazine, Vol. 32, No. 2, 2015, pp. 145-163.
- [9] J. Kruskal. Rank Decomposition, and Uniqueness for 3-way and N-way Arrays, Chapter in: Multiway Data Analysis, R. Coppi and S. Bolasco (Eds.), North-Holland, Amsterdam, 1989, pp. 7-18.
- [10] D. Kressner, C. Tolber, Algorithm 941: htucher - A MATLAB toolbox for tensor in hierarchical Tucker format, Journal ACM Transactions on Mathematical Software (TOMS), Vol. 40, No. 3, 2014, Article 22.
- [11] L. Grasedyck. Hierarchical Singular Value Decomposition of Tensors, SIAM Journal on Matrix Analysis and Application, Vol. 31, No. 4, 2010, pp. 2029-2054.
- [12] N. Vannieuwenhoven, R. Vandebril, K. Meerbergen, A New Truncation Strategy for the Higher-Order SVD, SIAM Journal on Scientific Computing (SISC), Vol. 34, No. 2, 2012, pp. A1027–A1052.
- [13] R. Salmi, A. Richter, V. Koivunen, Sequential Unfolding SVD for Tensors with Applications in Array Signal Processing, IEEE Transactions on Signal Processing, Vol. 57, No. 12, 2009, pp. 4719-4733.
- [14] R. Kountchev, R. Kountcheva, Hierarchical Decomposition of 2D/3D Images, based on SVD 2×2 , International Journal of Multimedia and Image Processing (IJMIP), Vol. 5, No. 3/4, 2015, pp. 286-296.
- [15] S. White, Applications of Distributed Arithmetic to Digital Signal Processing, IEEE ASSP Magazine, Vol. 6, No. 3, 1989, pp. 4-19.
- [16] R. Kountchev, S. Anwar, R. Kountcheva, M. Milanova, Face Recognition in Home Security System Using Tensor Decomposition Based on Radix-(2x2) Hierarchical SVD, Chapter in: Lecture Notes in Artificial Intelligence, F. Schweker and S. Scherer (Eds.), Springer, 2017, pp. 48-59.
- [17] P. Dinkova, P. Georgieva, M. Milanova, Face Recognition using SVD and Hidden Markov Models, 16th WSEAS Intern. Conference on Mathematical Methods, Computational Techniques and Intelligent Systems, Lisbon, Portugal, 2014, pp. 144-149.
- [18] A. Georghiades, P. Belhumeur, D. Kriegman. From Few to Many: Illumination Cone Models for Face Recognition under Variable Lighting and Pose, IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), Vol. 23, No 6, June 2001, pp. 643-660, <http://vision.ucsd.edu/content/yale-face-database>, (Access date 05.06.2017).