

Article

# Third-Order Tensor Decorrelation Based on 3D FO-HKLT with Adaptive Directional Vectorization

Roumen K. Kountchev <sup>1,\*</sup>, Rumen P. Mironov <sup>1</sup> and Roumiana A. Kountcheva <sup>2</sup>

<sup>1</sup> Department of Radio Communications and Video Technologies, Faculty of Telecommunications, Technical University of Sofia, 1000 Sofia, Bulgaria; rmironov@tu-sofia.bg

<sup>2</sup> TK Engineering, 1582 Sofia, Bulgaria; kountcheva\_r@yahoo.com

\* Correspondence: rkountch@tu-sofia.bg; Tel.: +359-9924284

**Abstract:** In this work, we present a new hierarchical decomposition aimed at the decorrelation of a cubical tensor of size  $2^n$ , based on the 3D Frequency-Ordered Hierarchical KLT (3D FO-HKLT). The decomposition is executed in three consecutive stages. In the first stage, after adaptive directional vectorization (ADV) of the input tensor, the vectors are processed through one-dimensional FO-Adaptive HKLT (FO-AHKLT), and after folding, the first intermediate tensor is calculated. In the second stage, on the vectors obtained after ADV of the first intermediate tensor, FO-AHKLT is applied, and after folding, the second intermediate tensor is calculated. In the third stage, on the vectors obtained from the second intermediate tensor, ADV is applied, followed by FO-AHKLT, and the output tensor is obtained. The orientation of the vectors, calculated from each tensor, could be horizontal, vertical or lateral. The best orientation is chosen through analysis of their covariance matrix, based on its symmetry properties. The kernel of FO-AHKLT is the optimal decorrelating KLT with a matrix of size  $2 \times 2$ . To achieve higher decorrelation of the decomposition components, the direction of the vectors obtained after unfolding of the input tensor in each of the three consecutive stages, is chosen adaptively. The achieved lower computational complexity of FO-AHKLT is compared with that of the Hierarchical Tucker and Tensor Train decompositions.

**Keywords:** cubical tensor decomposition; adaptive KLT; 3D Frequency-Ordered Adaptive Hierarchical KLT; adaptive directional vectorization; correlation vector analysis



**Citation:** Kountchev, R.K.; Mironov, R.P.; Kountcheva, R.A. Third-Order Tensor Decorrelation Based on 3D FO-HKLT with Adaptive Directional Vectorization. *Symmetry* **2022**, *14*, 854. <https://doi.org/10.3390/sym14050854>

Academic Editor: Zhixun Su

Received: 25 March 2022

Accepted: 14 April 2022

Published: 21 April 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

The main tensor decompositions could be divided into two groups.

The *first group* comprises decompositions executed in the spatial domain of the tensor. These are the famous Canonical Polyadic Decomposition (CPD), Higher-Order Singular Value Decomposition (HOSVD) [1–4], Tensor Train (TT) Decomposition [5], Hierarchical Tucker (H-Tucker) algorithm [6] and some of their modifications [7,8], based on the calculation of the tensor eigenvalues and eigenvectors. Their most important feature is that they are optimal regarding the minimization of the mean square approximation error derived from the low-energy component “truncation”. The calculation of the retained components is based on iterative methods [9,10] that need a relatively small number of mathematical operations to achieve the requested accuracy. The hierarchical tensor decompositions based on the H-Tucker algorithm are presented in publications [8,11]. The compositional hierarchical tensor factorization introduced in [8] disentangles the hierarchical causal structure of object image formation, but the computational complexity (or Complexity) is not presented. In [11] is offered the TT-based hierarchical decomposition of high-order tensors, based on the Tensor-Train Hierarchical SVD (TT-HSVD). This approach permits parallel processing, which significantly accelerates the process. Unlike the TT-SVD algorithm, TT-HSVD is based on applying SVDs to matrices of smaller dimensions, which results in lower Complexity of TT-HSVD.

The *second group* comprises tensor decompositions performed in the transform domain, which use reversible 3D linear orthogonal transforms such as the Fast Fourier Transform (FFT), Discrete Cosine Transform (DCT), etc. [12–14]. This approach is distinguished by its flexibility regarding the choice of the transform based on the processed data contents.

In this work, we present alternative new hierarchical 3D tensor decompositions based on the famous statistical orthogonal Karhunen–Loeve Transform (KLT) [15,16]. They are close to optimal (which ensures full decorrelation of the decomposition components), but do not need iterations and have lower computational complexity. As a basis, we present here the decomposition called 3D Frequency-Ordered Adaptive Hierarchical Karhunen–Loeve Transform (3D FO-AHKLT), whose efficiency is enhanced through adaptive directional tensor vectorization (ADV).

In Section 2, we present the method for 3D hierarchical adaptive transform based on the one-dimensional Frequency-Ordered Adaptive Hierarchical Karhunen–Loeve Transform (FO-AHKLT). Section 3 gives the details on the cubical tensor decomposition through separable 3D FO-AHKLT based on correlation analysis, and Section 4 explains the related algorithm. In Section 5, we analyze the computational complexity of the new approaches compared to that of the well-known H-Tucker and TT decompositions; Section 6 contains the conclusion.

## 2. Method for 3D Adaptive Frequency-Ordered Hierarchical KLT of a Cubical Tensor

The proposed method for decomposition of 3rd order cubical tensor  $\mathcal{X} \in \mathbb{R}^{N \times N \times N}$  is based on the 3D Frequency-Ordered Hierarchical Karhunen–Loeve Transform (3D FO-HKLT), defined by the relation below [17]:

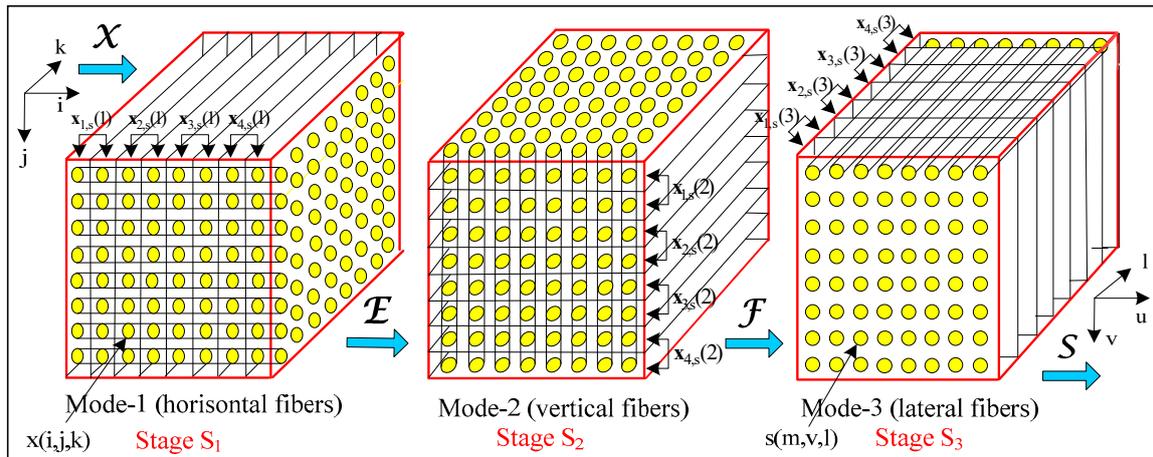
$$\mathcal{X} = (1/N^3) \sum_{m=0}^{N-1} \sum_{v=0}^{N-1} \sum_{l=0}^{N-1} s(m, v, l) [\mathbf{k}_m(1) \circ \mathbf{k}_v(2) \circ \mathbf{k}_l(3)]. \quad (1)$$

Here,  $N = 2^n$  is the size of the tensor  $\mathcal{X}$  with nonnegative components  $x(i, j, k)$ . The coefficients  $s(m, v, l)$  are the elements of the spectrum tensor  $\mathcal{S}$ , which is of the same size as  $\mathcal{X}$ . Each coefficient  $s(m, v, l)$  represents the weights of the basic tensor,  $\mathcal{K}_{u,v,l}$ . Each basic tensor is represented as the outer product of three vectors:

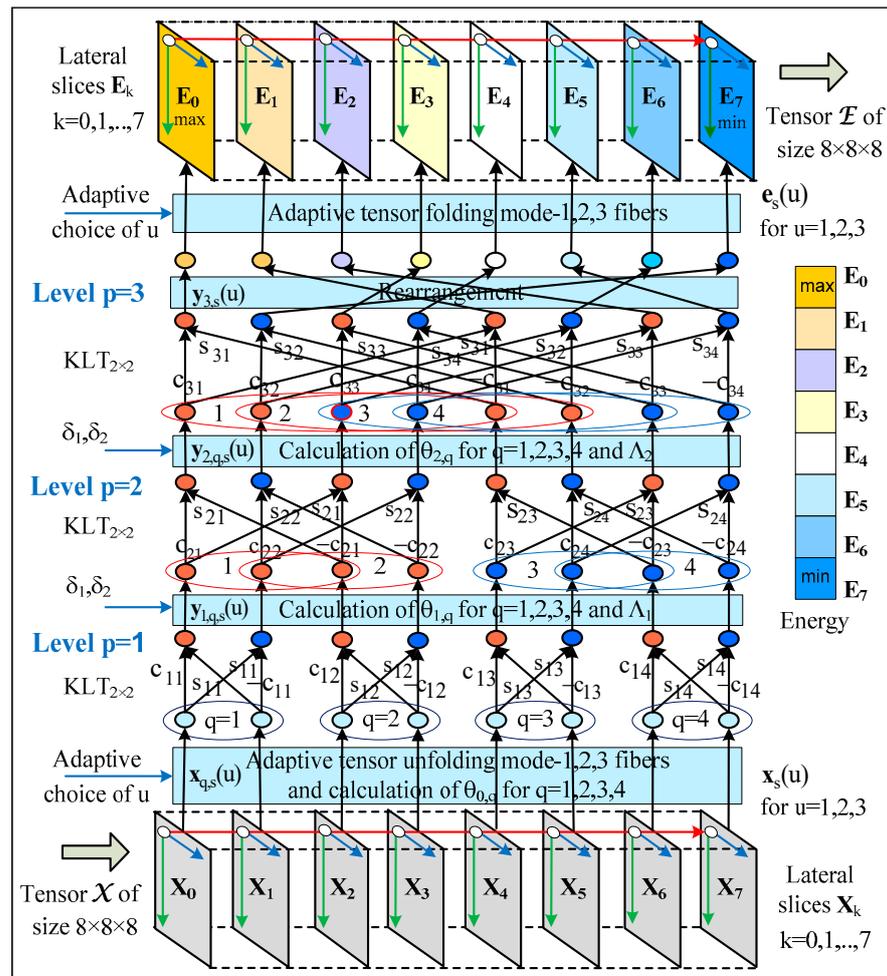
$$\mathcal{K}_{m,v,l} = \mathbf{k}_m(1) \circ \mathbf{k}_v(2) \circ \mathbf{k}_l(3). \quad (2)$$

Here, “ $\circ$ ” denotes the outer product of the two column-vectors ( $\mathbf{x} \circ \mathbf{y} = \mathbf{x} \cdot \mathbf{y}^T$ ), and  $\mathbf{k}_m(1), \mathbf{k}_v(2), \mathbf{k}_l(3)$  are the basic vectors, obtained after execution of the three stages of 3D FO-HKLT. In the first decomposition components  $s(m, v, l) \cdot \mathcal{K}_{m,v,l}$  (as given in Equation (1)) the main part of the tensor  $\mathcal{X}$  power is concentrated, and their high decorrelation is achieved. The kernel of 3D FO-HKLT, defined as  $\text{KLT}_{2 \times 2}$ , is KLT with a transform matrix of size  $2 \times 2$ .

The decomposition comprises three consecutive stages arranged in accordance with the correlation analysis of the tensor  $\mathcal{X}$  elements. One example decomposition for the tensor  $\mathcal{X}$  of size  $N \times N \times N$  (for  $N = 8$ ) is shown in Figure 1. After applying the 3D FO-AHKLT on the input tensor  $\mathcal{X}$ , it is sequentially transformed into the intermediate tensors  $\mathbf{E}, \mathbf{F}$  and the output tensor,  $\mathcal{S}$ . The 3D FO-AHKLT is divisible, and this permits it to be executed by using the FO-AHKLT, whose graph for the case  $N = 8$  is shown in Figure 2. As a result, the tensor  $\mathcal{X}$  is transformed into the first intermediate tensor  $\mathbf{E}$ , of the same size.



**Figure 1.** Sequence of the 3D FO-AHKLT execution stages for a tensor  $\mathcal{X}$  of size  $8 \times 8 \times 8$  for the case,  $\Delta(1) > \Delta(2) > \Delta(3)$ , based on FO-AHKLT. On each 2-component  $x_{q,s}(u)$  for  $q = 1, 2, 3, 4$ ,  $KLT_{2 \times 2}$  is applied.



**Figure 2.** Graphs of the 3-level direct FO-AHKLT of the tensor  $\mathcal{X}$  of size  $8 \times 8 \times 8$ , with ADV.

In the general case, FO-HKLT is executed for the  $N$ -dimensional vectors  $x_s(u)$  of total number  $N^2$ , oriented in horizontal ( $u = 1$ ), vertical ( $u = 2$ ) or lateral ( $u = 3$ ) direction. These vectors are defined as a result of the ADV of the tensor  $\mathcal{X}$ . The choice of the orientation direction of vectors  $x_s(u)$  for  $s = 1, 2, \dots, N^2$  is defined through analysis of their covariance

matrices,  $\mathbf{K}_x(\mathbf{u})$ . Before the calculation of FO-HKLT for the  $N$ -dimensional vector  $\mathbf{x}_s(\mathbf{u})$ , it is divided into  $N/2$  two-component vectors  $\mathbf{x}_{q,s}(\mathbf{u})$ . For the case  $N = 8$ , each vector  $\mathbf{x}_s(\mathbf{u})$  is divided into 4 vectors  $\mathbf{x}_{q,s}(\mathbf{u})$ , for  $q = 1, 2, 3, 4$ . For this case, the total number of vectors  $\mathbf{x}_s(\mathbf{u})$  is  $4N^2 = 256$ , and they are divided into 4 sub-groups of  $N^2 = 64$  vectors  $\mathbf{x}_{q,s}(\mathbf{u})$  each.

Let  $\mathbf{x}_s = [x_{1,s}, x_{2,s}, \dots, x_{n,s}]^T$  and  $\mathbf{e}_s = [e_{1,s}, e_{2,s}, \dots, e_{N,s}]^T$  be the input and the output column-vectors of  $N = 2^n$  components, respectively, and  $n$  is the number of FO-HKLT hierarchical levels. The relation between vectors  $\mathbf{e}_s$  and  $\mathbf{x}_s$ , is [17]:

$$\mathbf{e}_s = \mathbf{P}_n(2^n) \left[ \prod_{p=1}^n \mathbf{G}_p(2^n) \right] \mathbf{x}_s = \mathbf{T}_{\text{FO-HKLT}}(2^n) \mathbf{x}_s \text{ for } s = 1, 2, \dots, N^2, \tag{3}$$

where  $\mathbf{T}_{\text{FO-HKLT}}(2^n) = \mathbf{P}_n(2^n) \left[ \prod_{p=1}^n \mathbf{G}_p(2^n) \right]$  is the FO-HKLT matrix;  $\mathbf{P}_n(2^n)$  of size  $2^n \times 2^n$

is the permutation matrix for the last level  $n$  of FO-HKLT, and  $\prod_{p=1}^n \mathbf{G}_p(2^n)$  is the product of  $n$  sparse transform matrices  $\mathbf{G}_p(2^n)$  for  $p = 1, 2, 3, \dots, n$ . Each matrix  $\mathbf{G}_p(2^n)$  is defined as follows:

$$\mathbf{G}_p(2^n) = \begin{bmatrix} \bigoplus_{j=1}^{2^{p-1}} \mathbf{C}_{p,j} & \bigoplus_{j=1}^{2^{p-1}} \mathbf{S}_{p,j} \\ \bigoplus_{j=1}^{2^{p-1}} \mathbf{S}_{p,j} & \bigoplus_{j=1}^{2^{p-1}} -\mathbf{C}_{p,j} \end{bmatrix} \oplus \begin{bmatrix} \bigoplus_{j=1}^{2^{p-1}} \mathbf{C}_{p,2^{p-1}+j} & \bigoplus_{j=1}^{2^{p-1}} \mathbf{S}_{p,2^{p-1}+j} \\ \bigoplus_{j=1}^{2^{p-1}} \mathbf{S}_{p,2^{p-1}+j} & \bigoplus_{j=1}^{2^{p-1}} -\mathbf{C}_{p,2^{p-1}+j} \end{bmatrix} \oplus \dots \oplus \begin{bmatrix} \bigoplus_{j=1}^{2^{p-1}} \mathbf{C}_{p,2^{p-1}(n-p-1)+j} & \bigoplus_{j=1}^{2^{p-1}} \mathbf{S}_{p,2^{p-1}(n-p-1)+j} \\ \bigoplus_{j=1}^{2^{p-1}} \mathbf{S}_{p,2^{p-1}(n-p-1)+j} & \bigoplus_{j=1}^{2^{p-1}} -\mathbf{C}_{p,2^{p-1}(n-p-1)+j} \end{bmatrix} \tag{4}$$

for  $p = 1, 2, 3, \dots, n$ , where “ $\oplus$ ” denotes the direct sum of matrices.

In the equations, figures and text below, the abbreviations  $c_{p,q} = \cos \theta_{p,q}$  and  $s_{p,q} = \sin \theta_{p,q}$  are used. Here:

$$\theta_{p,q} = \arctan \left\{ 2k_3(p,q) / [k_1(p,q) - k_2(p,q) + \sqrt{(k_1(p,q) - k_2(p,q))^2 + 4k_3^2(p,q)}] \right\}, \tag{5}$$

$$k_1(p,q) = E \{ x_{1,s}^2(p,q) \}; k_2(p,q) = E \{ x_{2,s}^2(p,q) \}; k_3(p,q) = E \{ x_{1,s}(p,q)x_{2,s}(p,q) \}. \tag{6}$$

$E \{ x_{i,s}(p,q) \} = (1/N^2) \sum_{s=1}^{N^2} x_{i,s}(p,q)$  for  $i = 1, 2$ -averaging operator.

If  $\beta_2 = 2k_3(p,q)$  and  $\beta_1 = k_1(p,q) - k_2(p,q) + \sqrt{(k_1(p,q) - k_2(p,q))^2 + 4k_3^2(p,q)}$  are two real numbers, the extended arctan function  $\epsilon_{\arctan}(\beta_1, \beta_2)$  is mapped to  $(-\pi, \pi]$  by adding  $\pi$  [18]:

$$\epsilon_{\arctan}(\beta_1, \beta_2) = \begin{cases} \arctan(\beta_2/\beta_1), & \text{if } \beta_1 > 0; \\ \arctan(\beta_2/\beta_1) + \pi, & \text{if } \beta_1 < 0, \beta_2 \geq 0; \\ \arctan(\beta_2/\beta_1) - \pi, & \text{if } \beta_1 < 0, \beta_2 < 0; \\ \pi/2, & \text{if } \beta_1 = 0, \beta_2 > 0; \\ -\pi/2, & \text{if } \beta_1 = 0, \beta_2 < 0; \\ 0, & \text{if } \beta_2 = \beta_1 = 0. \end{cases} \tag{7}$$

In particular, the direct and inverse  $\text{KLT}_{2 \times 2}$  of the elements  $x_{1s}$  and  $x_{2s}$ , which have the same spatial position in the couple of matrices  $\mathbf{X}_1$  and  $\mathbf{X}_2$  of size  $N \times N$  for  $m_1 = E \{ x_{1s} \} = 0$  and  $m_2 = E \{ x_{2s} \} = 0$  (valid for the Adaptive  $\text{KLT}_{2 \times 2}$ ), are:

$$y_{1s} = x_{1s} \cos \theta_{1,2} + x_{2s} \sin \theta_{1,2}; x_{1s} = y_{1s} \cos \theta_{1,2} + y_{2s} \sin \theta_{1,2}; \tag{8}$$

$$y_{2s} = x_{1s} \sin \theta_{1,2} - x_{2s} \cos \theta_{1,2}; x_{2s} = y_{1s} \sin \theta_{1,2} - y_{2s} \cos \theta_{1,2}. \tag{9}$$

or  $\mathbf{y}_s = \mathbf{T}(2) \cdot \mathbf{x}_s$  and  $\mathbf{x}_s = \mathbf{T}(2) \cdot \mathbf{y}_s$  for  $s = 1, 2, \dots, N^2$ .

Here,  $y_{1s}$  and  $y_{2s}$  are the corresponding elements in the couple of transformed matrices  $\mathbf{Y}_1$  and  $\mathbf{Y}_2$  (each of size  $N \times N$ );  $\mathbf{y}_s = [y_{1s}, y_{2s}]^T$ ,  $\mathbf{x}_s = [x_{1s}, x_{2s}]^T$  and  $[\mathbf{T}(2)] = \begin{bmatrix} \cos \theta_{1,2} & \sin \theta_{1,2} \\ \sin \theta_{1,2} & -\cos \theta_{1,2} \end{bmatrix} = \begin{bmatrix} c_{1,2} & s_{1,2} \\ s_{1,2} & -c_{1,2} \end{bmatrix}$ .

In particular, for  $\theta_{1,2} = \pi/4$ , the relations (8) and (9) become as follows:

$$\begin{aligned} y_{1s} &= (x_{1s} + x_{2s}); & x_{1s} &= (1/2)(y_{1s} + y_{2s}); \\ y_{2s} &= (x_{1s} - x_{2s}); & x_{2s} &= (1/2)(y_{1s} - y_{2s}). \end{aligned} \tag{10}$$

Hence, in this case, the  $\text{KLT}_{2 \times 2}$  coincides with the Walsh–Hadamard Transform (WHT).

In the level  $n$ , the components of the column-vectors  $\mathbf{y}_{n,s} = \mathbf{G}_n(2^n)\mathbf{y}_{n-1,s}$  (respectively, the components of matrices  $\mathbf{Y}_{n,k}$  for  $k = 0, 1, \dots, N - 1$ ) are rearranged. For this, the permutation matrix  $\mathbf{P}_n(2^n)$  is used.

From the components of the column-vectors  $\mathbf{e}_s = \mathbf{P}_n(2^n)\mathbf{y}_{n,s}$  are obtained the frequency-ordered matrices  $\mathbf{E}_r$  (i.e.,  $\mathbf{E}_0, \mathbf{E}_1, \dots, \mathbf{E}_{N-1}$ ), calculated in accordance with the relation between their sequential number  $r$  and the sequential number  $k$  (for matrices  $\mathbf{Y}_{n,k}$ ).

The relation that defines the matrix  $\mathbf{P}_n(2^n)$  is:

- The binary code  $k_{n-1}, k_{n-2}, \dots, k_0$  of the sequential decimal number  $k = 0, 1, \dots, 2^{n-1}$  of the component  $\mathbf{Y}_{n,k}$  is arranged inversely (i.e.,  $k_0, k_1, \dots, k_{n-1}$ ), as  $g_i = k_{n-i-1}$  for  $0 \leq i \leq n - 1$ ;
- The so-obtained code  $g_{n-1}, g_{n-2}, \dots, g_0$  is transformed from Gray code into the binary code  $r_{n-1}, r_{n-2}, \dots, r_0$ , in accordance with the operations  $r_{n-1} = g_{n-1}, r_i = r_{i-1} \oplus g_i$  for  $0 \leq i \leq n - 2$ . Here, “ $\oplus$ ” denotes the operation “exclusive OR”.

The decimal number  $r = \sum_{i=0}^{n-1} r_i 2^i$  defines the sequential number of  $\mathbf{E}_r$  for  $r = 0, 1, \dots, 2^{n-1}$ , which (before the rearrangement) corresponded to  $\mathbf{Y}_{n,k}$ , with sequential number  $k = \sum_{i=0}^{n-1} k_i 2^i$ .

For  $N = 8$  ( $n = 3$ ),  $p = 1, 2, 3$  and  $q = 1, 2, 3, 4$ , FO-HKLT is executed in three consecutive levels. The transform matrix of size  $8 \times 8$  is decomposed in accordance with the relation:

$$\mathbf{T}_{\text{FO-HKLT}}(8) = \mathbf{P}_3(8)\mathbf{G}_3(8)\mathbf{G}_2(8)\mathbf{G}_1(8) \tag{11}$$

Since  $\mathbf{e}_s = \mathbf{P}_3(8)\mathbf{G}_3(8)\mathbf{G}_2(8)\mathbf{G}_1(8)\mathbf{x}_s$ , the transform is executed as follows:

$$\mathbf{y}_{1,s} = \mathbf{G}_1(8)\mathbf{x}_s, \mathbf{y}_{2,s} = \mathbf{G}_2(8)\mathbf{y}_{1,s}, \mathbf{y}_{3,s} = \mathbf{G}_3(8)\mathbf{y}_{2,s}, \mathbf{e}_s = \mathbf{P}_3(8)\mathbf{y}_{3,s} \tag{12}$$

The operations above correspond to the calculations in hierarchical levels  $p = 1, 2, 3$  of FO-HKLT. In this case, the transform matrices  $\mathbf{G}_1(8)$ ,  $\mathbf{G}_2(8)$ ,  $\mathbf{G}_3(8)$  are defined by the equations:

$$\mathbf{G}_1(8) = \begin{bmatrix} c_{1,1} & s_{1,1} & 0 & 0 & 0 & 0 & 0 & 0 \\ s_{1,1} & -c_{1,1} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & c_{1,2} & s_{1,2} & 0 & 0 & 0 & 0 \\ 0 & 0 & s_{1,2} & -c_{1,2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & c_{1,3} & s_{1,3} & 0 & 0 \\ 0 & 0 & 0 & 0 & s_{1,3} & -c_{1,3} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & c_{1,4} & s_{1,4} \\ 0 & 0 & 0 & 0 & 0 & 0 & s_{1,4} & -c_{1,4} \end{bmatrix} \tag{13}$$

where  $\theta_{1,q} = \arctan \left\{ 2k_3(1,q) / [k_1(1,q) - k_2(1,q) + \sqrt{(k_1(1,q) - k_2(1,q))^2 + 4k_3^2(1,q)}] \right\}$   
 for  $q = 1, 2, 3, 4$  and  $p = 1$ ;  $k_1(1,q) = E\{x_{1,s}^2(1,q)\}$ ;  $k_2(p,q) = E\{x_{2,s}^2(1,q)\}$ ;  $k_3(1,q) = E\{x_{1,s}(1,q) \cdot x_{2,s}(1,q)\}$ .

$$G_2(8) = \begin{bmatrix} c_{2,1} & 0 & s_{2,1} & 0 & 0 & 0 & 0 & 0 \\ 0 & c_{2,2} & 0 & s_{2,2} & 0 & 0 & 0 & 0 \\ s_{2,1} & 0 & -c_{2,1} & 0 & 0 & 0 & 0 & 0 \\ 0 & s_{2,2} & 0 & -c_{2,2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & c_{2,3} & 0 & s_{2,3} & 0 \\ 0 & 0 & 0 & 0 & 0 & c_{2,4} & 0 & s_{2,4} \\ 0 & 0 & 0 & 0 & s_{2,3} & 0 & -c_{2,3} & 0 \\ 0 & 0 & 0 & 0 & 0 & s_{2,4} & 0 & -c_{2,4} \end{bmatrix} \tag{14}$$

where:  $\theta_{2,q} = \arctan \left\{ 2k_3(2,q) / [k_1(2,q) - k_2(2,q) + \sqrt{(k_1(2,q) - k_2(2,q))^2 + 4k_3^2(2,q)}] \right\}$   
 for  $q = 1, 2, 3, 4$  and  $p = 2$ ;  $k_1(2,q) = E\{y_{1,1s}^2(2,q)\}$ ;  $k_2(2,q) = E\{y_{1,2s}^2(2,q)\}$ ;  $k_3(2,q) = E\{y_{1,1s}(2,q) \cdot y_{1,2s}(2,q)\}$ .

$$G_3(8) = \begin{bmatrix} c_{3,1} & 0 & 0 & 0 & s_{3,1} & 0 & 0 & 0 \\ 0 & c_{3,2} & 0 & 0 & 0 & s_{3,2} & 0 & 0 \\ 0 & 0 & c_{3,3} & 0 & 0 & 0 & s_{3,3} & 0 \\ 0 & 0 & 0 & c_{3,4} & 0 & 0 & 0 & s_{3,4} \\ s_{3,1} & 0 & 0 & 0 & -c_{3,1} & 0 & 0 & 0 \\ 0 & s_{3,2} & 0 & 0 & 0 & -c_{3,2} & 0 & 0 \\ 0 & 0 & s_{3,3} & 0 & 0 & 0 & -c_{3,3} & 0 \\ 0 & 0 & 0 & s_{3,4} & 0 & 0 & 0 & -c_{3,4} \end{bmatrix} \tag{15}$$

where  $\theta_{3,q} = \arctan \left\{ 2k_3(3,q) / [k_1(3,q) - k_2(1,q) + \sqrt{(k_1(3,q) - k_2(3,q))^2 + 4k_3^2(3,q)}] \right\}$   
 for  $q = 1, 2, 3, 4$  and  $p = 3$ ;  $k_1(3,q) = E\{y_{2,1s}^2(3,q)\}$ ;  $k_2(3,q) = E\{y_{2,2s}^2(3,q)\}$ ;  $k_3(3,q) = E\{y_{2,1s}(3,q) \cdot y_{2,2s}(3,q)\}$ .

The permutation matrix  $P_3(8)$  in the level  $p = 3$  is:

$$P_3(8) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{16}$$

The vector  $k_m(1)$  for  $m = 0, 1, \dots, 7$  comprises the elements of the  $m$ th row of the matrix  $T_{FO-HKLT}^1(8)$  for the stage  $S_1$  of 3D FO-HKLT. The vectors  $k_v(2)$  and  $k_l(3)$  for  $v, l = 0, 1, \dots, 7$  respectively comprise the elements of rows  $v$  and  $l$  in matrices  $T_{FO-HKLT}^2(8)$  and  $T_{FO-HKLT}^3(8)$ , for 3D FO-HKLT stages  $S_2$  and  $S_3$ .

### 3. Enhancement of the 3D FO-HKLT Efficiency, Based on Correlation Analysis

To achieve higher efficiency of the 3D tensor decomposition based on the 3D FO-HKLT, here, we use the correlation relations between the tensor  $\mathcal{X}$  components in the orthogonal directions  $x, y, z$ . To detect the direction in which these relations are strongest, correlation

analysis should be used. It is based on the analysis of the covariance matrices  $\mathbf{K}_x(\mathbf{u})$  of vectors  $\mathbf{x}_s(\mathbf{u})$ , obtained through unfolding mode  $u = 1, 2, 3$  of the tensor  $\mathcal{X}$ .

### 3.1. Analysis of the Covariance Matrices

To estimate the correlation of vectors  $\mathbf{x}_s(\mathbf{u}) = [x_{1,s}(\mathbf{u}), x_{2,s}(\mathbf{u}), \dots, x_{N,s}(\mathbf{u})]^T$ , for 3D FO-AHKLT, we use the ratio  $\Delta(\mathbf{u})$  of the sums of the squares of coefficients  $k_{x,ij}(\mathbf{u})$  placed outside the main diagonal of the matrix  $\mathbf{K}_x(\mathbf{u})$ , and those on the diagonal:

$$\Delta(\mathbf{u}) = \left\{ \sum_{i=1}^N \sum_{j=1}^N [k_{x,ij}(\mathbf{u})]_{(i \neq j)}^2 / \sum_{i=1}^N \sum_{j=1}^N [k_{x,ij}(\mathbf{u})]_{(i=j)}^2 \right\} \text{ for } \mathbf{u} = 1, 2, 3 \quad (17)$$

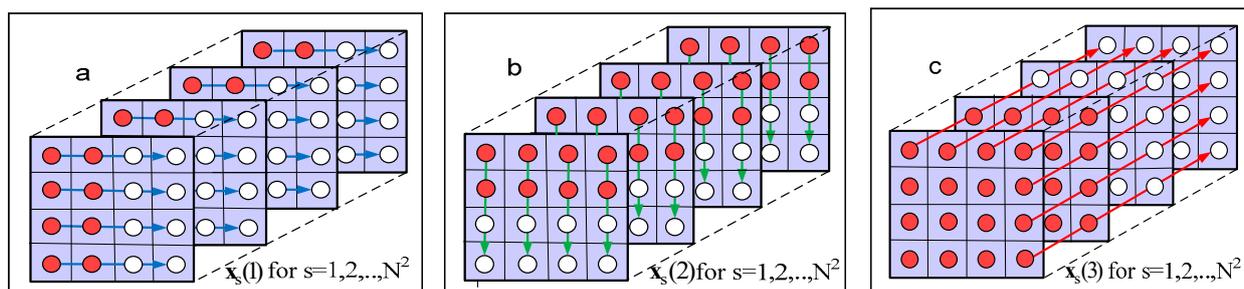
The relation above takes into account the symmetry of coefficients  $k_{x,ij}(\mathbf{u}) = k_{x,ji}(\mathbf{u})$  for  $i \neq j$ , in respect of the main diagonal of the covariance matrix,  $\mathbf{K}_x(\mathbf{u})$ . The value of this ratio is maximum for the highest correlation of vectors  $\mathbf{x}_s(\mathbf{u})$  of same orientation,  $\mathbf{u}$ .

### 3.2. Choice of Vectors' Orientation for Adaptive Directional Tensor Vectorization

The choice of the orientation  $u = 1, 2, 3$  of vectors  $\mathbf{x}_s(\mathbf{u})$  in each stage  $S_u$  of 3D FO-AHKLT which ensures maximum decorrelation for the tensor  $\mathcal{X}$ , is based on the relation between coefficients  $\Delta(\mathbf{u})$ , given in Table 1. Figure 3a–c shows the vectors  $\mathbf{x}_s(1), \mathbf{x}_s(2), \mathbf{x}_s(3)$ , which have horizontal, vertical and lateral orientation, correspondingly, and are obtained through unfolding mode  $u = 1, 2, 3$  of the tensor  $\mathcal{X}$ , of size  $4 \times 4 \times 4$ . The couples of elements of the vectors  $\mathbf{x}_{q,s}(\mathbf{u})$  for  $q = 1, 2$  on which  $\text{KLT}_{2 \times 2}$  is applied in the lowest level of FO-HKLT, are colored in red and white.

**Table 1.** Adaptive choice of the orientation of vectors  $\mathbf{x}_s(\mathbf{u})$  for  $u = 1, 2, 3$ , based on the correlation analysis.

No.	Relations for $\Delta(1), \Delta(2), \Delta(3)$	Sequence of Stages $S_u$ in the 3D FO-AHKLT Execution (for $u = 1, 2, 3$ )
1	$\Delta(1) > \Delta(2) > \Delta(3)$	$S_1 \rightarrow S_2 \rightarrow S_3$
	$\Delta(1) = \Delta(2) = \Delta(3)$	
	$\Delta(1) > \Delta(2) = \Delta(3)$	
	$\Delta(1) = \Delta(2) > \Delta(3)$	
2	$\Delta(1) > \Delta(3) > \Delta(2)$	$S_1 \rightarrow S_3 \rightarrow S_2$
	$\Delta(1) > \Delta(3) = \Delta(2)$	
	$\Delta(1) = \Delta(3) > \Delta(2)$	
	$\Delta(2) > \Delta(1) > \Delta(3)$	
3	$\Delta(2) > \Delta(1) = \Delta(3)$	$S_2 \rightarrow S_1 \rightarrow S_3$
	$\Delta(2) = \Delta(1) > \Delta(3)$	
	$\Delta(2) > \Delta(3) > \Delta(1)$	
	$\Delta(2) > \Delta(3) = \Delta(1)$	
4	$\Delta(2) = \Delta(3) > \Delta(1)$	$S_2 \rightarrow S_3 \rightarrow S_1$
	$\Delta(3) > \Delta(2) > \Delta(1)$	
	$\Delta(3) > \Delta(2) = \Delta(1)$	
	$\Delta(3) = \Delta(2) > \Delta(1)$	
5	$\Delta(3) > \Delta(1) > \Delta(2)$	$S_3 \rightarrow S_2 \rightarrow S_1$
	$\Delta(3) = \Delta(2) > \Delta(1)$	
	$\Delta(3) > \Delta(1) > \Delta(2)$	
	$\Delta(3) > \Delta(1) = \Delta(2)$	
6	$\Delta(3) > \Delta(1) = \Delta(2)$	$S_3 \rightarrow S_1 \rightarrow S_2$
	$\Delta(3) = \Delta(1) > \Delta(2)$	



**Figure 3.** Orientations of vectors  $\mathbf{x}_s(\mathbf{u})$  for  $\mathbf{u} = 1, 2, 3$  and  $N = 4$  after unfolding mode  $\mathbf{u} = 1, 2, 3$  of the tensor  $\mathcal{X}$ : (a) horizontal; (b) vertical; (c) lateral.

### 3.3. Evaluation of the Decorrelation Properties of FO-AHKLT

The qualities of the one-dimensional FO-AHKLT correspond to the decorrelation degree of the output eigen images, which defines the number of needed execution levels,  $n$ . As an example, here is given the evaluation of the decorrelation properties of the transform, for the case  $N = 8$  and  $n = 3$ . In the hierarchical level  $p = 1, 2, 3$  of FO-AHKLT, the covariance matrix  $\mathbf{K}_y^p$  of size  $8 \times 8$ , which represents the transformed 8-component vectors  $\mathbf{y}_{p,s}(\mathbf{u}) = [y_{p,1,s}(\mathbf{u}), y_{p,2,s}(\mathbf{u}), \dots, y_{p,8,s}(\mathbf{u})]^T$ , is:

$$\mathbf{K}_y^p(\mathbf{u}) = E\{\mathbf{y}_{p,s}(\mathbf{u}) \cdot \mathbf{y}_{p,s}^T(\mathbf{u})\} - E\{\mathbf{y}_{p,s}(\mathbf{u})\} \cdot E\{\mathbf{y}_{p,s}(\mathbf{u})\}^T = \begin{bmatrix} \mathbf{K}_{y_1}^p(\mathbf{u}) & \mathbf{K}_{y_1,y_2}^p(\mathbf{u}) & \mathbf{K}_{y_1,y_3}^p(\mathbf{u}) & \mathbf{K}_{y_1,y_4}^p(\mathbf{u}) \\ \mathbf{K}_{y_2,y_1}^p(\mathbf{u}) & \mathbf{K}_{y_2}^p(\mathbf{u}) & \mathbf{K}_{y_2,y_3}^p(\mathbf{u}) & \mathbf{K}_{y_2,y_4}^p(\mathbf{u}) \\ \mathbf{K}_{y_3,y_1}^p(\mathbf{u}) & \mathbf{K}_{y_3,y_2}^p(\mathbf{u}) & \mathbf{K}_{y_3}^p(\mathbf{u}) & \mathbf{K}_{y_3,y_4}^p(\mathbf{u}) \\ \mathbf{K}_{y_4,y_1}^p(\mathbf{u}) & \mathbf{K}_{y_4,y_2}^p(\mathbf{u}) & \mathbf{K}_{y_4,y_3}^p(\mathbf{u}) & \mathbf{K}_{y_4}^p(\mathbf{u}) \end{bmatrix} \quad (18)$$

for  $s = 1, 2, \dots, 64$ .

Here, the sub-matrices  $\mathbf{K}_{y_q}^p(\mathbf{u})$  of size  $2 \times 2$  (for  $q = 1, 2, 3, 4$ ) are defined as:

$$\mathbf{K}_{y_q}^p(\mathbf{u}) = E\{\mathbf{y}_{q,s}^p(\mathbf{u}) \cdot \mathbf{y}_{q,s}^{p,T}(\mathbf{u})\} - E\{\mathbf{y}_{q,s}^p(\mathbf{u})\} \cdot E\{\mathbf{y}_{q,s}^p(\mathbf{u})\}^T = \begin{bmatrix} \lambda_{1,q}^p(\mathbf{u}) & 0 \\ 0 & \lambda_{2,q}^p(\mathbf{u}) \end{bmatrix} \quad (19)$$

These are the covariance matrices of the transformed vectors  $\mathbf{y}_{q,s}^p(\mathbf{u}) = [y_{q,1,s}^p(\mathbf{u}), y_{q,2,s}^p(\mathbf{u})]^T$  in the group  $q$  for the level  $p$ , and  $\lambda_{1,q}^p(\mathbf{u}), \lambda_{2,q}^p(\mathbf{u})$  are the eigen values of the covariance matrices  $\mathbf{K}_{y_q}^p(\mathbf{u})$ . Respectively,

$$\mathbf{K}_{y_q,y_k}^p(\mathbf{u}) = E\{\mathbf{y}_{q,s}^p(\mathbf{u}) \cdot \mathbf{y}_{k,s}^{p,T}(\mathbf{u})\} - E\{\mathbf{y}_{q,s}^p(\mathbf{u})\} \cdot E\{\mathbf{y}_{k,s}^p(\mathbf{u})\}^T \text{ for } q, k = 1, 2, 3, 4 (q \neq k) \quad (20)$$

are the mutual covariance matrices of size  $2 \times 2$  of the two-component vectors  $\mathbf{y}_{q,s}^p(\mathbf{u})$  and  $\mathbf{y}_{k,s}^p(\mathbf{u})$  for the couples of groups  $q$  and  $k$  from the level  $p$ .

After the rearrangement that follows the decomposition level  $p = 3$ , the vectors  $\mathbf{y}_{3,s}(\mathbf{u}) = [y_{3,1,s}(\mathbf{u}), y_{3,2,s}(\mathbf{u}), \dots, y_{3,8,s}(\mathbf{u})]^T$  are transformed into vectors  $\mathbf{e}_s(\mathbf{u}) = [e_{1,s}(\mathbf{u}), e_{2,s}(\mathbf{u}), \dots, e_{8,s}(\mathbf{u})]^T$ . To evaluate the result of FO-AHKLT, the covariance matrix  $\mathbf{K}_e^3(\mathbf{u})$  of the vectors  $\mathbf{e}_s(\mathbf{u})$  is analyzed, from which the achieved decorrelation degree in the level is defined.

### 4. Adaptive Control for Each Level of FO-AHKLT

After the execution of FO-AHKLT in the current level  $p$ , from the transformed two-component vectors  $\mathbf{y}_{q,s}^p(\mathbf{u}) = [y_{q,1,s}^p(\mathbf{u}), y_{q,2,s}^p(\mathbf{u})]^T$  for each group  $q = 1, 2, \dots, (N/2)$ , through concatenation the  $N$ -dimensional vectors  $\mathbf{y}_{p,s}(\mathbf{u}) = [y_{p,1,s}(\mathbf{u}), y_{p,2,s}(\mathbf{u}), \dots, y_{p,n,s}(\mathbf{u})]^T$  for  $p = 1, 2, \dots, n (n = \lg_2 N)$  and  $s = 1, 2, \dots, N^2$  are obtained. At this moment, the decision

to continue with the next level of FO-AHKLT, or to stop, must be taken. For this, the covariance matrix  $\mathbf{K}_y^p(\mathbf{u})$  of the vectors  $\mathbf{y}_{p,s}(\mathbf{u})$  is analyzed for  $s = 1, 2, \dots, S$ , which defines the achieved decorrelation. In the case that the decorrelation is full, their matrix  $\mathbf{K}_y^p(\mathbf{u})$  is diagonal, and the algorithm is stopped. The proposed adaptive control of FO-AHKLT permits the process to stop earlier, despite that full decorrelation is not achieved, if the result is satisfactory. The decision to stop the FO-AHKLT in the current level  $p$  is defined by the relation:

$$\Lambda_p = \sum_{i=1}^N \sum_{j=1}^N [k_{y_p,ij}(\mathbf{u})]_{(i \neq j)}^2 \leq \delta_1 \tag{21}$$

Here,  $k_{y_p,ij}(\mathbf{u})$  is the  $(i, j)$ th element of the matrix  $\mathbf{K}_y^p(\mathbf{u})$ , and  $\delta_1$  is a threshold of small value, set in advance. In the case that the condition is satisfied, the calculations stop. Otherwise, the processing continues with the next FO-AHKLT level,  $p + 1$ . When the calculations for the second level are finished, the result is checked again, but in this case  $k_{y_{p+1},ij}(\mathbf{u})$  are the elements of the matrix  $\mathbf{K}_y^{p+1}(\mathbf{u})$  of the vectors  $\mathbf{y}_{p+1,s}(\mathbf{u})$ , etc.

Taking into account the condition (21), the FO-AHKLT matrix defined in Equation (3) turns into:

$$\mathbf{T}_{\text{FO-AHKLT}}(2^n) = \begin{cases} \mathbf{P}_{p_0}(2) \left[ \prod_{p=1}^{p_0} \mathbf{G}_p(2^n) \right] \text{ for } p_0 < n & \text{if } \Lambda_{p_0} \leq \delta_1; \\ \mathbf{P}_n(2^n) \left[ \prod_{p=1}^n \mathbf{G}_p(2^n) \right] \text{ for } p = 1, 2, \dots, n - 1 & \text{if } \Lambda_p > \delta_1, \end{cases} \tag{22}$$

$$y_{p,1,s}^q(\mathbf{u}) = \begin{cases} y_{p-1,1,s}^q(\mathbf{u}) \cos \theta_{p,q}(\mathbf{u}) + y_{p-1,2,s}^q(\mathbf{u}) \sin \theta_{p,q}(\mathbf{u}) & \text{if } [k_{p,3}^q(\mathbf{u})]^2 > \delta_2; \\ y_{p-1,1,s}^q(\mathbf{u}) & \text{if } [k_{p,3}^q(\mathbf{u})]^2 \leq \delta_2, \end{cases} \tag{23}$$

$$y_{p,2,s}^q(\mathbf{u}) = \begin{cases} y_{p-1,1,s}^q(\mathbf{u}) \sin \theta_{p,q}(\mathbf{u}) - y_{p-1,2,s}^q(\mathbf{u}) \cos \theta_{p,q}(\mathbf{u}) & \text{if } [k_{p,3}^q(\mathbf{u})]^2 > \delta_2; \\ -y_{p-1,2,s}^q(\mathbf{u}) & \text{if } [k_{p,3}^q(\mathbf{u})]^2 \leq \delta_2 \end{cases} \tag{24}$$

for  $p = 2, 3, \dots, n, q = 1, 2, \dots, (N/2)$  and  $k_{p,3}^q(\mathbf{u}) = E\{y_{p-1,1,s}^q(\mathbf{u}), y_{p-1,2,s}^q(\mathbf{u})\}$ .

For the level  $p = 1$  only, the relations (22) and (23) are transformed as follows:

$$y_{1,1,s}^q(\mathbf{u}) = \begin{cases} x_{1,1,s}^q(\mathbf{u}) \cos \theta_{1,q}(\mathbf{u}) + x_{1,2,s}^q(\mathbf{u}) \sin \theta_{1,q}(\mathbf{u}) & \text{if } [k_{1,3}^q(\mathbf{u})]^2 > \delta_2; \\ x_{1,1,s}^q(\mathbf{u}) & \text{if } [k_{1,3}^q(\mathbf{u})]^2 \leq \delta_2. \end{cases} \tag{25}$$

$$y_{1,2,s}^q(\mathbf{u}) = \begin{cases} x_{1,1,s}^q(\mathbf{u}) \sin \theta_{1,q}(\mathbf{u}) - x_{1,2,s}^q(\mathbf{u}) \cos \theta_{1,q}(\mathbf{u}) & \text{if } [k_{1,3}^q(\mathbf{u})]^2 > \delta_2; \\ -x_{1,2,s}^q(\mathbf{u}) & \text{if } [k_{1,3}^q(\mathbf{u})]^2 \leq \delta_2, \end{cases} \tag{26}$$

where  $[k_{1,3}^q(\mathbf{u})]^2 = E\{x_{1,1,s}^q(\mathbf{u}), x_{1,2,s}^q(\mathbf{u})\}$ .

In these relations,  $\delta_2$  is a threshold of small value, set in advance. The condition (21), together with the adaptive  $\text{KLT}_{2 \times 2}$  performed in accordance with relations (23)–(26), permits reducing the number of calculations without worsening the 3D FO-AHKLT decorrelation properties.

### 5. Algorithm 3D FO-AHKLT

On the basis of the 3D FO-HKLT tensor decomposition, together with the adaptive control of the directional vectorization on the input, the intermediate and the output tensor, and the correlation analysis, the algorithm called 3D FO-AHKLT is developed as Algorithm 1:

---

**Algorithm 1**

---

**Input:** Third-order tensor  $\mathcal{X}$  of size  $N \times N \times N$  ( $N = 2^n$ ) with elements  $x(i, j, k)$ , and threshold values,  $\delta_1, \delta_2$ ;

The steps of the algorithm are given below:

1. Unfolding of the tensor  $\mathcal{X}$  (mode  $u = 1, 2, 3$ ), thereby obtaining the corresponding groups of vectors,  $\mathbf{x}_s(u)$ .
2. Calculation of the covariance matrices  $\mathbf{K}_y^p(u)$  of the vectors  $\mathbf{x}_s(u)$ , for  $u = 1, 2, 3$ .
3. Calculation of the coefficients  $\Delta(u)$  for  $u = 1, 2, 3$ , using the matrices  $\mathbf{K}_y^p(u)$ .
4. Setting the sequence of stages  $S_{u1}, S_{u2}, S_{u3}$  for the execution of 3D FO-AHKLT, chosen in accordance with the relations between coefficients  $\Delta(1), \Delta(2), \Delta(3)$ .
5. Start of stage  $S_{u1}$ , which comprises:
  - 5.1. Execution of FO-AHKLT for the vectors  $\mathbf{x}_s(u_1)$  with orientation  $u_1$  (for  $\Delta(u_1) = \max$ ), in correspondence with the calculation graph of  $n$  hierarchical levels and with a kernel—the adaptive  $KLT2 \times 2$ , thereby obtaining the vectors  $\mathbf{e}_s(u_1)$ . In each level  $p = 1, 2, \dots, n$  is checked if the condition to stop the FO-AHKLT calculations is satisfied;
  - 5.2. Folding of the tensor  $E$  of size  $N \times N \times N$ , by using the vectors  $\mathbf{e}_s(u_1)$ ;
  - 5.3. Unfolding of the tensor  $E$  (mode  $u_2$ ), thereby obtaining the vectors  $\mathbf{x}_s(u_2)$  with orientation  $u_2$ , for  $\Delta(u_2) \leq \Delta(u_1)$ .
6. Start of stage  $S_{u2}$ , which comprises:
  - 6.1. Execution of FO-AHKLT for the vectors  $\mathbf{x}_s(u_2)$  in correspondence with the computational graph of  $n$  hierarchical levels and with a kernel—the adaptive  $KLT2 \times 2$ , thereby obtaining the vectors  $\mathbf{y}_s(u_2)$ . In each level  $p = 1, 2, \dots, n$  is checked if the condition to stop the FO-AHKLT calculations is satisfied;
  - 6.2. Folding the tensor  $F$  of size  $N \times N \times N$ , by using the vectors  $\mathbf{y}_s(u_2)$ ;
  - 6.3. Unfolding the tensor  $F$  (mode  $u_3$ ), thereby obtaining the vectors  $\mathbf{x}_s(u_3)$ , with orientation  $u_3$ , for  $\Delta(u_3) \leq \Delta(u_2)$ ;
7. Start of stage  $S_{u3}$ , which comprises:
  - 7.1. Execution of FO-AHKLT for the vectors  $\mathbf{x}_s(u_3)$  in correspondence with the computational graph of  $n$  hierarchical levels and with a kernel—the adaptive  $KLT2 \times 2$ , thereby obtaining the vectors  $\mathbf{s}_s(u_3)$ . In each level  $p = 1, 2, \dots, n$  is checked if the condition to stop the FO-AHKLT calculations is satisfied;
  - 7.2. Folding of the tensor  $S$  of size  $N \times N \times N$  ( $N = 2n$ ) by using the vectors  $\mathbf{s}_s(u_3)$ . The elements of the tensor  $S$  are the spectrum coefficients  $s(m, v, l)$ .
8. Calculation of the transform matrices  $\mathbf{T}_{FO-AHKLT}^t(2^n)$  for each stage  $t = 1, 2, 3$  of the 3D FO-AHKLT in accordance with the relations below:

$$\mathbf{e}_s(u_1) = \mathbf{T}_{FO-AHKLT}^1(2^n)\mathbf{x}_s(u_1) \text{ for } s = 1, 2, \dots, N^2; \tag{27}$$

$$\mathbf{y}_s(u_2) = \mathbf{T}_{FO-AHKLT}^2(2^n)\mathbf{x}_s(u_2) \text{ for } s = 1, 2, \dots, N^2; \tag{28}$$

$$\mathbf{s}_s(u_3) = \mathbf{T}_{FO-AHKLT}^3(2^n)\mathbf{x}_s(u_3) \text{ for } s = 1, 2, \dots, N^2. \tag{29}$$

9. Determination of the basic vectors  $\mathbf{k}_m(1), \mathbf{k}_v(2), \mathbf{k}_l(3)$  for  $m, v, l = 0, 1, 2, \dots, N - 1$ , which are the rows of the matrices  $\mathbf{T}_{FO-AHKLT}^1(2^n), \mathbf{T}_{FO-AHKLT}^2(2^n), \mathbf{T}_{FO-AHKLT}^3(2^n)$ . With this, the decomposition of the tensor  $\mathcal{X}$  in correspondence with Equation (1), is finished.

**Output:** Spectral tensor  $S$ , whose elements  $s(m, v, l)$  are the coefficients of the 3D FO-AHKLT.

---

As a result of the algorithm execution:

- The main part of the tensor energy is concentrated into a small number of coefficients  $s(m, v, l)$  of the spectrum tensor  $S$ , for  $m, v, l = 0, 1, 2$ ;
- The decomposition components of the tensor are uncorrelated.

**6. Comparative Evaluation of the Computational Complexity**

For the computation of the covariance matrix  $\mathbf{K}_y^p(u) = E\{\mathbf{y}_{p,s}(u) \cdot \mathbf{y}_{p,s}^T(u)\} - E\{\mathbf{y}_{p,s}(u)\} \cdot E\{\mathbf{y}_{p,s}(u)\}^T$  of size  $N \times N$  for  $N = 2^n$ , in correspondence with Equation (17),  $A_{\mathbf{K}_y^p}(n) =$

$3 \times 2^{2n}$  additions and  $M_{\mathbf{K}_y^p}(n) = 2^{2n}$  multiplications are needed. Then, the total number of the operations needed for the calculation of  $\mathbf{K}_y^p(u)$  is:

$$O_{\mathbf{K}_y^p}(n) = A_{\mathbf{K}_y^p}(n) + M_{\mathbf{K}_y^p}(n) = 4 \times 2^{2n} \text{ for } p = 1, 2, \dots, n \tag{30}$$

For the  $n$  levels of the algorithm FO-HKLT is obtained:

$$\sum_{p=1}^n O_{\mathbf{K}_y^p}(n) = 4 \times 2^{2n}n \tag{31}$$

For the calculation of  $\Delta_u(u)$  in correspondence with Equation (17),  $A_{\Delta_u}(n) = 2^{2n} - 1$  additions and  $M_{\Delta_u}(n) = 2^{2n} + 1$  multiplications are needed. Then,

$$O_{\Delta_u}(n) = A_{\Delta_u}(n) + M_{\Delta_u}(n) = 2 \times 2^{2n} \text{ for } u = 1, 2, 3 \tag{32}$$

Hence,

$$\sum_{u=1}^3 O_{\Delta_u}(n) = 6 \times 2^{2n}. \tag{33}$$

In accordance with [17], the computational complexity (or Complexity) of FO-HKLT is defined by taking into account the number of needed additions  $A_{\text{HKLT}}^{\text{FO}}(n) = 2^{n-1}(5 \times 2^{2n} + 1)n \approx 2.5 \times 2^{3n}n$ , and multiplications  $M_{\text{HKLT}}^{\text{FO}}(n) = 2^{n-1}(7 \times 2^{2n} + 5)n \approx 3.5 \times 2^{3n}n$ . Hence, the total number of operations needed for the 3D FO-HKLT execution is:

$$O_{\text{HKLT}}^{\text{3DFO}}(n) = A_{\text{HKLT}}^{\text{3DFO}}(n) + M_{\text{HKLT}}^{\text{3DFO}}(n) \approx 18 \times 2^{3n}n \tag{34}$$

The total number of operations needed for the execution of the algorithm 3D FO-AHKLT (without taking into consideration the possibility to stop the processing earlier than the last level) is:

$$O_{\text{AHKLT}}^{\text{3DFO}}(n) = A_{\text{AHKLT}}^{\text{3DFO}}(n) + M_{\text{AHKLT}}^{\text{3DFO}}(n) \approx 2^{2n+1}[2n(9 \times 2^{n-1} + 1) + 3] \tag{35}$$

For the H-Tucker and TT decomposition of a cubical tensor of size  $N = 2^n$ , in accordance with [17], the number of needed operations is:

$$O_{\text{HT}}(n) = 2^{3n}(2^{n+1} + 3), O_{\text{TT}}(n) = 3 \times 2^{4n} \tag{36}$$

Compared to H-Tucker and TT, the Complexity of the new decomposition decreases together with the growth of  $n$ . In the general case, the relative Complexity of 3D FO-AHKLT with respect to decompositions H-Tucker and TT is evaluated in accordance with the relations below:

$$\psi_1(n) = \frac{O_{\text{HT}}(n)}{O_{\text{AHKLT}}^{\text{3DFO}}(n)} = 2^{n-1} \frac{2^{n+1} + 3}{2n(9 \times 2^{n-1} + 1) + 3} \tag{37}$$

$$\psi_2(n) = \frac{O_{\text{TT}}(n)}{O_{\text{AHKLT}}^{\text{3DFO}}(n)} = 2^{2n-1} \frac{3}{2n(9 \times 2^{n-1} + 1) + 3} \tag{38}$$

For example, for  $n = 8$  the results are:  $\psi_1(8) = 3.57$  and  $\psi_2(8) = 5.33$ .

For the case when the new decomposition must have minimum Complexity, the transform kernel  $\text{KLT}_{2 \times 2}$  could be replaced by WHT of the same size ( $2 \times 2$ ), which in correspondence with Equation (10), is the particular case of  $\text{KLT}_{2 \times 2}$  for the fixed value of  $\theta_{1,2} = \pi/4$ . According to [19], the Complexity of the 3D Fast Walsh-Hadamard Transform (3D-FWHT) evaluated through the total number of operations  $O_{\text{3D-FWHT}}(n)$  is defined by the relation:

$$O_{\text{3D-FWHT}}(n) = A_{\text{3D-FWHT}}(n) + M_{\text{3D-FWHT}}(n) = 3 \times 2^{3n}n \tag{39}$$

Then, for the Complexity of 3D-FWHT with ADV (3D-AFWHT), we obtain:

$$O_{3D-AFWHT}(n) = A_{3D-AFWHT}(n) + M_{3D-AFWHT}(n) \approx 2^{2n}[3 \times 2^n n + 2(2n + 3)] \quad (40)$$

The relative Complexity of 3D FO-AHKLT with respect to 3D-AFWHT is defined by the relation:

$$\psi_3(n) = \frac{O_{AHKLT}^{3DFO}(n)}{O_{3D-AFWHT}(n)} = 2 \frac{2n(9 \times 2^{n-1} + 1) + 3}{3 \times 2^n n + 2(2n + 3)} \quad (41)$$

Hence, for the example value  $n = 8$ , it is  $\psi_3(n) = 5.97$ , and the relative Complexities of H-Tucker and TT vs. 3D-AFWHT are  $O_{HT}(8)/O_{3D-AFWHT}(8) = 21.33$  and  $O_{TT}(8)/O_{3D-AFWHT}(8) = 31.84$ , correspondingly.

In Table 2 are given the values of coefficients  $\psi_1(n)$ ,  $\psi_2(n)$  and  $\psi_3(n)$ , calculated in accordance with Equations (36), (37) and (40), for  $n = 2, 3, \dots, 10$ . From the table, it is seen that for  $n > 5$ , the values of coefficients  $\psi_1(n)$  and  $\psi_2(n)$  are higher than “1” and grow fast together with  $n$ , while for the same values of  $n$ , the coefficient  $\psi_3(n)$  increases a little in the range from 4 up to 6.

**Table 2.** Relative Complexity of H-Tucker and TT vs. 3D FO-AHKLT and of 3D FO-AHKLT vs. 3D-AFWHT.

n	2	3	4	5	6	7	8	9	10
$\psi_1(n)$	0.28	0.34	0.48	0.74	1.21	2.05	3.57	6.34	11.39
$\psi_2(n)$	0.30	0.42	0.66	1.06	1.77	3.04	5.33	9.47	17.06
$\psi_3(n)$	4.16	5.00	5.46	5.74	5.87	5.94	5.97	5.98	5.99

From the comparison follow the conclusions below:

- The new hierarchical decomposition has low Complexity, which decreases together with the growth of  $n$  faster than those of the H-Tucker and TT decompositions;
- Significant reduction in the decomposition Complexity could be achieved through replacement of the kernel  $KLT_{2 \times 2}$  by  $WHT_{2 \times 2}$ . In this particular case, the decrease in the decomposition Complexity results in a lower decorrelation degree;
- For the case  $n = 8$ , the Complexity of the algorithm 3D-AFWHT is approximately 6 times lower than that of the 3D FO-AHKLT;
- The Complexity of algorithms 3D FO-AHKLT and 3D-AFWHT was evaluated without taking into consideration the use of the adaptive  $KLT_{2 \times 2}$  and the possibility to stop the execution prior to the maximum level  $n$ . Equations (35) and (40) give the maximum values of Complexity used for the evaluation of the compared algorithms.

### 7. Conclusions

This work presented the new algorithm 3D FO-AHKLT, aimed at the decorrelation of the elements of a cubical tensor of size  $N = 2^n$ . The Complexity of the algorithm was evaluated and compared with that of other, similar algorithms, and its efficiency was shown. The main qualities of the tensor decomposition 3D FO-AHKLT are:

- Efficient decorrelation of the calculated components;
- Concentration of the tensor energy into a small number of decomposition components;
- Lack of iterations;
- Low Complexity;
- The capacity for parallel recursive implementation, which reduces the needed memory volume;
- The capacity for additional significant Complexity reduction through the use of the algorithm 3D-AFWHT, depending on the needs of the implemented application.

The presented algorithms 3D FO-AHKLT and 3D-AFWHT could be generalized for tensors with three different dimensions  $2^{n_1} \times 2^{n_2} \times 2^{n_3}$  (i.e., for  $n_1 \neq n_2 \neq n_3$ ). The choice

of the offered hierarchical 3D decompositions depends on the requirements and limitations of their Complexity imposed by the application area.

The future investigations of 3D FO-AHKL and 3D-AFWHT will be aimed at the evaluation of their characteristics compared to famous tensor decompositions, in order to define the best settings and to define the most efficient applications for tensor image compression, feature space reduction, filtration, analysis, search and recognition of multidimensional visual information, deep learning, etc. The future development of the presented algorithm will be aimed at applications related to tree tensor network [20], multiway array (tensor) data analysis [21,22], tensor decompositions in neural networks for tree-structured data, etc.

**Author Contributions:** Conceptualization, R.K.K., R.P.M. and R.A.K.; methodology, formal analysis, R.A.K.; investigation, R.P.M.; writing—original draft preparation, R.K.K.; writing—review and editing, R.A.K.; visualization, R.K.K. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was funded by the Bulgarian National Science Fund: Project No. KP-06-H27/16: “Development of efficient methods and algorithms for tensor-based processing and analysis of multidimensional images with application in interdisciplinary areas”.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

## References

- Liu, Y.; Liu, J.; Long, Z.; Zhu, C. Tensor Decomposition. In *Tensor Computation for Data Analysis*; Springer: Cham, Switzerland, 2021; pp. 19–57.
- Ji, Y.; Wang, Q.; Li, X.; Liu, J. A Survey on Tensor Techniques and Applications in Machine Learning. *IEEE Access* **2019**, *7*, 162950–162990. [[CrossRef](#)]
- Cichocki, A.; Mandic, D.; Phan, A.; Caiafa, C.; Zhou, G.; Zhao, Q.; De Lathauwer, L. Tensor Decompositions for Signal Processing Applications: From Two-Way to Multiway Component Analysis. *IEEE Signal Process. Mag.* **2015**, *32*, 145–163. [[CrossRef](#)]
- Sidiropoulos, N.; De Lathauwer, L.; Fu, X.; Huang, K.; Papalexakis, E.; Faloutsos, C. Tensor Decomposition for Signal Processing and Machine Learning. *IEEE Trans. Signal Proc.* **2017**, *65*, 3551–3582. [[CrossRef](#)]
- Oseledets, I. Tensor-train decomposition. *Siam J. Sci. Comput.* **2011**, *33*, 2295–2317. [[CrossRef](#)]
- Grasedyck, L. Hierarchical Singular Value Decomposition of Tensors. *Siam J. Matrix Anal. Appl.* **2010**, *31*, 2029–2054. [[CrossRef](#)]
- Ozdemir, A.; Zare, A.; Iwen, M.; Aviyente, S. Extension of PCA to Higher Order Data Structures: An Introduction to Tensors, Tensor Decompositions, and Tensor PCA. *Proc. IEEE* **2018**, *106*, 1341–1358.
- Vasilescu, M.; Kim, E. Compositional Hierarchical Tensor Factorization: Representing Hierarchical Intrinsic and Extrinsic Causal Factors. In Proceedings of the 25th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD): Tensor Methods for Emerging Data Science Challenges, Anchorage, NY, USA, 2–4 August 2019; p. 9.
- Wang, P.; Lu, C. Tensor Decomposition via Simultaneous Power Iteration. In Proceedings of the 34th International Conference on Machine Learning (PMLR), Sydney, Australia, 6–9 August 2017; Volume 70, pp. 3665–3673.
- Ishteva, M.; Absil, P.; Van Dooker, P. Jacobi Algorithm for the Best Low Multi-linear Rank Approximation of Symmetric Tensors. *SIAM J. Matrix Anal. Appl.* **2013**, *34*, 651–672. [[CrossRef](#)]
- Zniyed, Y.; Boyer, R.; Almeida, A.; Favier, G. A TT-based Hierarchical Framework for Decomposing High-Order Tensors. *SIAM J. Sci. Comput.* **2020**, *42*, A822–A848. [[CrossRef](#)]
- Kernfeld, E.; Kilmer, M.; Aeron, S. Tensor–tensor products with invertible linear transforms. *Linear Algebra Appl.* **2015**, *485*, 545–570.
- Keegany, K.; Vishwanathz, T.; Xu, Y. A Tensor SVD-based Classification Algorithm Applied to fMRI Data. *arXiv* **2021**, arXiv:2111.00587v1.
- Rao, K.; Kim, D.; Hwang, J. *Fast Fourier Transform: Algorithms and Applications*; Springer: Dordrecht, The Netherlands, 2010; pp. 166–170.
- Ahmed, N.; Rao, K. *Orthogonal Transforms for Digital Signal Processing*; Springer: Berlin/Heidelberg, Germany, 1975; pp. 86–91.

16. Kountchev, R.; Kountcheva, R. Adaptive Hierarchical KL-based Transform: Algorithms and Applications. In *Computer Vision in Advanced Control Systems: Mathematical Theory*; Favorskaya, M., Jain, L., Eds.; Springer: Berlin/Heidelberg, Germany, 2015; Volume 1, pp. 91–136.
17. Kountchev, R.; Mironov, R.; Kountcheva, R. Complexity Estimation of Cubical Tensor Represented through 3D Frequency-Ordered Hierarchical KLT. *Symmetry* **2020**, *12*, 1605. [[CrossRef](#)]
18. Lu, H.; Kpalma, K.; Ronsin, J. Motion descriptors for micro-expression recognition. In *Signal Processing: Image Communication*; Elsevier: Amsterdam, The Netherlands, 2018; Volume 67, pp. 108–117.
19. Kountchev, R.; Mironov, R.; Kountcheva, R. Hierarchical Cubical Tensor Decomposition through Low Complexity Orthogonal Transforms. *Symmetry* **2020**, *12*, 864. [[CrossRef](#)]
20. Cichocki, A.; Phan, H.; Zhao, Q.; Lee, N.; Oseledets, I.; Sugiyama, M.; Mandic, D. Tensor networks for dimensionality reduction and large-scale optimizations: Part 2 Applications and future perspectives. *Found. Trends Mach. Learn.* **2017**, *9*, 431–673. [[CrossRef](#)]
21. Abhyankar, S.S.; Christensen, C.; Sundaram, G.; Sathaye, A.M. *Algebra, Arithmetic and Geometry with Applications: Papers from Shreeram S. Abhyankar's 70th Birthday Conference*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2004; pp. 457–472.
22. Peters, J. *Computational Geometry, Topology and Physics of Digital Images with Applications: Shape Complexes, Optical Vortex Nerves and Proximities*; Springer Nature: Cham, Switzerland, 2019; pp. 1–115.