УДК 621.397

SPECTRAL SPACES FOR 3D MESH IMAGES

Osharovskaya E. V.

O.S. Popov Odessa national academy of telecommunications,
1 Kovalska St., Odessa 65029, Ukraine.

osharovskaya@gmail.com

СПЕКТРАЛЬНІ ПРОСТОРИ ДЛЯ СІТКОВИХ 3D ЗОБРАЖЕНЬ

Ошаровська О. В.

Одеська національна академія зв'язку ім. О.С. Попова,
65029, Україна, м. Одеса, вул. Ковальська, 1.

osharovskaya@gmail.com

СПЕКТРАЛЬНЫЕ ПРОСТРАНСТВА ДЛЯ СЕТОЧНЫХ 3D ИЗОБРАЖЕНИЙ

Ошаровская Е.В.

Одесская национальная академия связи им. А.С. Попова,
65029, Украина, г. Одесса, ул. Кузнечная, 1

osharovskaya@gmail.com

Abstract. In a multidimensional space is preserved topological proximity: close events are displayed in the close area. The space thus clustered into regions similar in some respects. .Dynamic 3D meshes of television images are sequences of static meshes with unchanged-term connection having geometric correlation of the connections, both in space on-field as well as in time. Compression methods of such meshes can be divided into two classes: the spectral-transformations and prediction in space. Applying the methods for reducing the dimension of the signal space: Wavelet transform, Tensor-Train Decomposition and principal component analysis (PCA) are now competing with each other among spectral methods.

Key words: 3D meshes, TV image, spectral space, compression, decomposition, tensor train

Анотація. У багатовимірному просторі зберігається топологічна близькість: близькі події відображаються в близькій зоні. Таким чином, простір розбивається на подібні в деяких відносинах кластери. Динамічні 3D сітки телевізійних зображень є послідовності статичних сіток з незмінним з'єднанням, що мають геометричну кореляцію з'єднань, як у просторі, так і в часі. Методи стиснення таких сіток можна розділити на два класи: спектральні перетворення і передбачення в просторі. Застосування методів для зменшення розмірності простору сигналу: вейвлет-перетворення, тензорні поїзд-розкладання й аналіз головних компонентів (PCA) в даний час конкурують один з одним серед спектральних методів.

Ключові слова: 3D сітки, ТВ зображення, спектральний простір, стиснення, декомпозиція, тензорний поїзд

Аннотация. В многомерном пространстве сохраняется топологическая близость: близкие события отображаются в близкие зоны. Таким образом, пространство разбивается на подобные в некоторых отношениях кластеры. Динамические 3D сетки телевизионных изображений являются последовательностями статических сеток с неизменной топологической связностью, имеющей геометрическую корреляцию соединений как в пространстве, так и во времени. Методы сжатия таких сеток можно разделить на два класса: спектральные преобразования и предсказания в пространстве. Применение методов для уменьшения размерности в пространстве сигнала: вейвлет-преобразования, тензорные поезд-разложения и анализ главных компонентов (PCA) в настоящее время конкурируют друг с другом для спектральных методов.

Ключевые слова: 3-D сетки. ТВ изображение, спектральное пространство, сжатие, декомпозиция, тензорный поезд.

Introduction

The purpose of image compression is to eliminate contained there physiological and statistical redundancy through the effective using of communication channels for the transmission of television 3D images. Accelerated development of advanced technology solutions will lead to a significant change in the range of requirements to the functional characteristics of subscriber units in the near future. You will need a variety of devices, provides the ability to high quality 3D images in the presence of significantly different reception conditions, performance and bandwidth of communication channels. Existing international standards for coding video images do not currently provide a fully effective coordination with the specific compression parameters and the characteristics of the telecommunications systems.

There is an urgent needing to carry out a comparative analysis of the promising methods of encoding dynamic images using transformations, ensuring the creation of a multi-level space- scalable structure. To consider efficient coding of numerous parameters, i.e. the transformation of the object shapes to the signals, the set of forms in a signal space, can be understood as an information space that is a direct source of information about the object to the subject, if it is able to decode these signals with the appropriate body feelings. Note that the process of generating the signal field does not necessarily imply the existence of subjects who could perceive this information. So here affirms the objective existence of the information as a set of natural form codes of material objects in the signaling processes of various environments as in simultaneously or sequentially facility is located. Naturally formed laws of nature primary code objects of the material world are filled with the surrounding space. Defining information as a form of code, we can say that the world is filled with information, and evolving in the world develops the subject at the appropriate organs of perception. Such senses should be able to convert the primary codes to another form for one reason or another comfortable for the subject, and on the basis of how the subject creates an internal image of the object. This internal image, in fact, is a subjective model of the object.

1 Analysis of research and publications

There are a variety of scalable coding schemes in the MPEG-4 visual: spatial scalability, temporary bridges and scalable object-oriented spatial scalability. Spatial scalability supports changing the texture quality (SNR and spatial resolution). Object-oriented spatial scalability expands "normal" types of scalability of objects in a direction of arbitrary shape, so that it can be used in conjunction with other object- oriented features. Thus, there may be achieved very flexible scalability. This makes it possible to improve the playback dynamically SNR, spatial resolution, fidelity of shape, etc., for objects of interest, or for a particular region [1].

Displaying information in a multidimensional space is a fruitful idea of information processing a variety of modalities, including image processing. In a multidimensional space is preserved topological proximity: close events are displayed in the close area. The space thus clustered into regions similar in some respects.

Multidimensional signal – is a signal, where each sample is a point in d-dimensional space: $d_x \in P^d$, where x – signal sample, P^d – d-dimensional space.

Under the decreasing of the signal space we understand a linear mapping of the original d-dimensional signal space of dimension k, where $k < d$. Operator effecting this transition should be determined by taking into account the possibility of reverse transition into the space P^d (i.e., approximately at the possibility of recovery of the signal). An example of such a linear operator can serve orthonormal matrix with the columns (S) with size $k \times d$. For such a matrix is true that $S^T S = E^k$, while $S\,S^T \neq E^k$, where S^T – transposed matrix S with size $d \times k$, E – the identity matrix of size $k \times k$. The dimension of vector $\vec{x}$ can be reduced by the help of the S matrix [1].

You can use the Johnson-Lindenstrauss lemma in applying the method for reducing the dimension of the signal space. Then there is a mapping $\boldsymbol{f}$: $P^d \rightarrow P^k$, such that the reduced image by approximating procedure will differ from the original by no more than ε [2].

Dynamic 3-D meshes are sequences of static meshes with unchanged-term connection having geometric correlation of the connections, both in space on-field as well as in time. Compression methods of such meshes can be divided into two classes: the spectral-transformations and prediction in space. Method of a parallelogram was widely used as prediction methods in the space. Wavelet transform and principal component analysis (PCA) are now competing with each other among spectral methods [3].

The principal component analysis is one of the main ways to reduce the dimensionality of the data, loosing the least amount of information. This method is widely distributed in image processing and data compression [4].

Evaluation of principal component analysis reduces to the calculation of the eigenvectors and eigenvalues of the covariance matrix of the original data. Sometimes the principal components method called Karhunen-Loeve transform, or Hotelling transform [5].

2.1 Main part

The task of principal component analysis has at least four basic versions:

– to approximate data linear manifolds of lower dimension;

– to find a subspace of lower dimension in the orthogonal projection on which scatter in the data (i.e., the standard deviation from the mean) is maximal;

– to find a subspace of lower dimension in the orthogonal projection on which mean square distance between the points of maximum;

– -to construct an orthogonal coordinate transformation for the multivariate random variable, which resulted in a correlation between the individual coordinates will turn to zero.

All the tasks of the main components drive to the problem of diagonalization of the covariance matrix or sample covariance matrix. Empirical or sample covariance matrix is

$$C = [c_{ij}],\ \mathrm{c_{ij}} = \frac{1}{m}\sum_{l=1}^{m}(x_{li} - \overline{X}_i)(x_{lj} - \overline{X}_j) \tag{1}$$

Covariance matrix of the multivariate random variable X is

$$\Sigma\Sigma = [\sigma_{ij}], \sigma_{\mathrm{ij}} = \mathrm{cov}(x_i x_j) = E[(x_i - E\overline{X}_i)(x_j - E\overline{X}_j) \tag{2}$$

Vectors of principal components for the problem of best approximation and orthogonal projections about finding with the highest scattering are a set of orthonormal eigenvectors $\{a_1,...,a_n\}$ of the empirical covariance matrix C, arranged in descending order $\lambda : \lambda_1 \geq \lambda_2 \geq ... \geq \lambda_n \geq 0$ of the eigenvalues. These vectors are used to estimate the eigenvectors of the covariance matrix $\mathrm{cov}(X_i, X_j)$. In the basis of the eigenvectors the covariance matrix is, of course, diagonal, and in this base the coefficient covariance between different coordinates is zero. If the spectrum of the covariance matrix is degenerate, then choose an arbitrary orthonormal basis of eigenvectors. It exists always, and the eigenvalues of the covariance matrix is always real and non-negative.

In PCA transformation is dominated image clustered approaches to compress the dynamic 3 -D meshes. The trajectories of vertices, i.e., single- path nodes in a group of frames are grouped into clusters, each cluster and for converting PCA is chosen so as to approximate best the global behavior of the vertices. Individual transformation and approximation errors are coded results. Dynamic mesh is represented as a matrix of large dimension 3VF, where V and F denote the number of vertices and a dynamic mesh frames, respectively. Compression is achieved by decomposing this matrix via RCA and omitting most of the basic vectors, which can then be recovered by interpolation. PCA approach for clusters vertex trajectories allows to obtain even higher degree of compression. PCA coefficients are predicted using the parallelogram predictor, which makes it less entropy balances.

This method works effectively for high pointed meshes, where the number of vertices substantially larger than the number of frames.

Using PCA transformation as a linear transformation is for certain types of data optimal in terms of the size of the data obtained with the same distortions. Also, data compression can be achieved by discarding the last conversion factors.

In recent years, as a new tool in the scientific computing was created multilinear algebra to solve large-scale problems of linear approximation of tensors with lower healing rang, that would be unsolvable by classical methods. This study attempts to provide an overview of the current state of the literature in this area, with an emphasis on functions related to tensors.

According to the number of elements in the chain approximation method for approximating tensors (Tensor-Train Decomposition) (TT) is close to the canonical approximation, while there is a stable algorithm for obtaining such an approximation. By the same method TT can be used in conjunction with a method for Tucker further reduce the number of elements [6].

The idea of the method is to provide high dimension chain tensor by tensor of small dimension (of 3). Approximation is more excessive than the canonical, by introducing an additional index. In this representation, each three-dimensional tensor G_k, associated with subsequent and previous tensors using the index α_k, so this structure is associated with the "chain" or "train" [6]

Application of wavelet filter chain (Wavelet Tensor-Train, WTT) is a modification of the method of TT, allows us to represent the original signal in a diluted form. The idea is to use tensors H_k as filters for the original signal (i.e., H_k is used as a matrix without transformation). Applying filters for an image is reduced a serial multiplication of the filter and the image matrix with a preliminary resizing matrix data to the size of the filter. The signal can be recovered, since the filters are orthogonal.

Further review is devoted tensor in terms of multi-dimensional arrays. In general, the tensor of order d and dimension $n_1 \times n_2 \times ... \times n_d$ for integers will be denoted $X \in R^{n_1 \times n_2 \times ... \times n_d}$

Source tensor X is represented as $X_{i_1,...,i_d}$ where each index $i_\mu \in \{1,...,n_\mu\}$ refers to μ tensor representation for $\mu = 1, ..., d$. For simplicity, we assume that X consists of the real values, but it is possible to determine tensors with complex values or, in general, tensors over arbitrary fields.

A wide variety of applications leads to problems in which the data or the desired solution can be represented by a tensor. In this review we will focus on tensors, derived as a result of sampling a multidimensional function. Discretization of multidimensional functions $F(x_1, x_2, ..., x_d)$ on the grid is the easiest way to come to a tensor in the domain $\Omega = [0,1]^d$ of the tensor product, In this case, each tensor element comprises a multi-dimensional vector value of the function at the corresponding point of the spatial grid.

With increasing order of d, the number of entries in the X increases exponentially with constant $n = n_1 = ... = n_d$

This so-called curse of dimensionality does not allow a clear record keeping except for very small values d, Even for $n = 2$, storage order tensor $d = 50$ will require 9 terabytes! Therefore, it is important to find ways of approximation of tensors of higher order tensor decomposition scheme in the lower ranks. Various such expansions were developed. As a rule, the tensor X, containing the observable data does not directly induced by function, but only as a solution to some of algebraic equations, for example, the linear system or the eigenvalue problem, it requires the development of solvers for such equations, working in the scheme of compressed storage. Such algorithms are described in [7].

Scope of low rank tensor methods is expanding rapidly. For example, they have been used to solve problems of approximation parameters of definite integrals, multidimensional integration and multi-dimensional convolution; computational problems in electronic structure calculation, for example, based on the Hartree-Fock or DFT models; problems of the rational approximation; solving multidimensional Schrödinger equations.

Despite the fact that the area of low-rank approximation of tensors is relatively young, it is already a challenge for a full review of all developments in this area. This study concerns the work associated with CT and hierarchical Tucker decomposition.

Rarely keep all records of a higher order tensor explicitly. Different compression schemes have been developed to reduce the memory requirements. For $d = 2$, all these schemes are reduced to the well- known algorithm singular value decomposition (SVD) of matrices.

Peer components of the tensor X can be represented in the form of (3):

$$X_{i_1,i_2,...,i_d} = u^{(1)}_{i_1} u^{(2)}_{i_2} ... u^{(d)}_{i_d}, \quad 1 \le i_\mu \le n_\mu, \quad \mu = 1,...,d \tag{3}$$

We define the vectors $u^{(\mu)} := (u^{(\mu)}_1,...,u^{(\mu)}_{n_\mu})^T$, in a more compact form

$$vec(\mathrm{X}) = u^{(d)} \otimes u^{(d-1)} \otimes ... \otimes u^{(1)},$$

where the sign ⊗ specifies the common Kronekovsk product. and *vec* is a stack for tensor components in the form of a long column vector with indices arranged in reverse order.

Using an external vector product *o*, this ratio can be written as $\mathrm{X} = u^{(1)} \circ u^{(2)} \circ ... \circ u^{(d)}$). In the case when the components of the tensor *X* are separable functions samples $f(x_1,x_2,..., x_d) = f_1(x_1)f_2(x_2)... f_d(x_d)$, then each vector $u^{(\mu)}$, corresponding samples f_μ., is a peer of the original tensor *X*.

CP decomposition described by a sum peer tensors:

$$\mathrm{vec}(X) = \mathrm{u(d)} \; vec(X) = u^{(d)}_1 \otimes u^{(d-1)}_1 \otimes ... \otimes u^{(1)}_1 + ... + u^{(d)}_R \otimes u^{(d-1)}_R \otimes ... \otimes u^{(1)}_R \tag{4}$$

Rank tensor X can be defined as a minimum value of R, where the tensor X is decomposed by CP decomposition on R components.

CP- decomposition requires storage components, $(n_1 + n_2 + ... + n_d)R$, which is very attractive for small *R*. In order to use the CP expansion for the approximation of functions associated with con- tensors, reliable and efficient compression techniques are essential.

In particular, it is often necessary to truncate rank tensor *R* to lower one. Almost all of the existing algorithms are based on carefully adapting existing optimization algorithms.

2.2 Taker Decomposition

Expansion in Tucker tensor *X* takes the form (5)

$$vec(X) = (U_d \otimes U_{d-1} \otimes ... \otimes U_1) vec(C) \tag{5}$$

where $U_1,U_2,...,U_d$, if $U_\mu \in R^{n_\mu \times r_\mu}$ called the factor matrices or basic matrices, and $\mathrm{C}= R^{r_1 \times r_2 \times ... \times r_d}$ called core tensor decomposition.

As a CP, Tucker decomposition has a long history and we refer to the survey [8] for a more detailed story. In the following, we briefly review the basic techniques needed to motivate TT and HT expansions described below.

Tucker decomposition similar to the representation of the tensor *X* by matrices. Array in μ-th approximation $\mathrm{X}^{(\mu)}$ formed dimension $n_\mu \times (n_1...n_{\mu-1}n_{\mu+1}n_d)$ in a special way from the original components of the tensor *X*:

$$X^{(\mu)}_{i_\mu,l} := X_{i_1,...i_d}, \qquad l = 1 + \sum_{\nu<\mu} (i_\nu - 1) \prod_{\eta<\nu} n_\mu + \sum_{\nu>\eta} (i_\nu - 1) \prod_{\substack{\eta<\nu \\ \eta\ne\mu}} n_\eta \,. \tag{6}$$

In particular, it followed from the relation (6),

$$X^\mu = U_\mu \cdot C^{(\mu)} \cdot (U_d \otimes ... \otimes U_{\mu+1} \otimes U_{\mu-1} \otimes ... \otimes U_1)^{T+1}, \; \mu=1,...,d \tag{7}$$

It follows that rank $(X^{(\mu)}) \le r_\mu$, as the first factor $U_\mu \in R^{n_\varpi \times r_\mu}$ obviously has rank at most r_μ. This motivates to define the multilinear rank (also called μ-rank) of a tensor X as the tuple (r_1, r_2,….,r_d) with $r_\mu = rank(X^\mu)$.

In contrast to the tensor rank related to the CP decomposition, the s*et* $T(r_1,...,r_d)$ *o*f tensors of μ -rank at most r_μ is closed.

Another consequence of the relation (4) is the higher-order SVD (HOSVD) introduced in [55, 56] for approximating a tensor by a Tucker decomposition (3) of lower multilinear rank. In HOSVD, the columns of each factor matrix U_μ are computed as the k_μ dominant left singular vectors of $X^{(\mu)}$. The core tensor is then obtained by forming $vec(C) := (U_d \otimes ... \otimes U_1)^T vec(X)$. Eventually, this yields

$$vec(\widetilde{X}) := (U_d k... \otimes U_1) \cdot vec(C) \in T(k_1,...,k_d).$$

In contrast to the matrix case, where the SVD yields a best low-rank approximation for all unitarily invariant norms [8], the truncated tensor $\widetilde{X}$ resulting from the HOSVD is usually not optimal. However, we have

$$\left\| X - \widetilde{X} \right\| \le \sqrt{d} \quad \min_{Y \in T(k_1,...,k_d)} \left\| X - Y \right\|$$

This quasi-optimality condition is usually sufficient for the purpose of obtaining an accurate approximation to a function-related tensor.

Various alternatives to improve on the approximation provided by the HOSVD have been developed, see [9] and the references therein. Recent developments include Newton-type methods on manifolds [6,8], a Jacobi algorithm for symmetric tensors [6], generalizations of Krylov subspace methods [10], and modifications of the HOSVD [11].

2.3 Tensor train decomposition

The need for storing the $r_1 \times ... \times r_d$ core tensor C renders the Tucker decomposition increasingly unattractive as d gets larger. This has motivated the search for decompositions which potentially avoid these exponentially growing memory requirements, while still featuring the two most important advantages of the Tucker decomposition: closed ness and SVD-based compression.

One well established candidate for such a decomposition is the so called TT (tensor train) decomposition, which takes the form

$$X_{i_1,...,i_d} = G_1(i_1) \cdot G_2(i_2) \cdots G_d(i_d), \quad G_\mu(i_\mu) \in R^{r_{\mu-1} \times r_\mu}, \tag{8}$$

where $r_0 = r_d = 1$. For every mode μ and every index i_μ the coefficients $G_\mu(i_\mu)$ are matrices. In the context of numerical analysis, a decomposition of the form (5) was first proposed in [10]. However, such a decomposition has been proposed earlier in the density-matrix renormalization group method (DMRG) for simulating quantum systems [11]. In this area, the term matrix product state (MPS) representation for the decomposition (5) has been established [219]. Suitable conditions that imply a unique MPS representation can be found in [6]. The connection between TT and MPS has been explained in [12].

Similar to the Tucker decomposition, the TT decomposition is closely related to certain matricizations of X. Let $X^{(1,...,\mu)}$ denote the matrix obtained by reshaping the entries of X into an $(n_1 \ldots n_\mu) \times (n_{\mu+1} \quad n_d)$ array, such that (5) implies rank $(X^{(1,...,\mu)}) \le r_\mu$ for $\mu = 1,...,d$. Consequently, the tuple containing the ranks of these matricizations is called the TT-rank of X. As explained, e.g., in [20] a quasi-best approximation in a TT decomposition for a given TT-rank can be obtained from the SVDs of $X^{(1,...,\mu)}$, similarly to the HOSVD. It is important to avoid the explicit construction of these matrices and the SVDs when truncating a tensor in TT decomposition to lower TT-rank. Such truncation algorithms are described in [12]. On the theoretical side, it turns out that the set TT

(r1,...,r_{d-1}) of tensors with TT-ranks bounded by r_μ is closed, Actually, the set of tensors with TT-rank equal to r_μ forms a smooth manifold [6, 12]. The Kahler manifold structure for complex MPS with open and periodic boundary conditions has been studied in [6].

Tensor network diagrams, which have been attributed to Penrose [13], are helpful in visualizing tensor decompositions and their manipulation. Figure 1 gives a few basic examples, see, e.g., [125, 128, 174] for more details. In particular, Figure 1 (v) gives an illustration of the contraction (5) representing a TT decomposition. In view of this diagram, the TT decomposition is also sometimes called linear tensor network [14].

In applications related to quantum spin systems, the tensor X often exhibits symmetries inherited from underlying physical properties. There are variants of MPS/TT that reflect such symmetries in the low-rank decomposition, see [15] and the references therein.

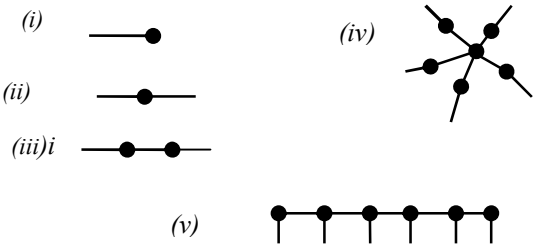

Figure 1 Tensor network diagrams representing (i) a vector, (ii) a matrix, (iii) a matrix-matrix multiplication, (iv) a tensor in Tucker decomposition, and (v) a tensor in TT decomposition.

2.4 Hierarchical Tucker decomposition

An alternative way to reduce the complexity of the Tucker decomposition is given by the hierarchical Tucker (HT) decomposition [16] (also called hierarchical tensor representation). This decomposition is based on the idea of recursively splitting the modes of the tensor, which results in a binary tree T containing a subset $t \subset \{1,...,d\}$ at each node. An example of such a binary tree is given in the left plot of Figure 2. The matricization $X^{(t)}$ of a tensor X corresponding to such a subset t merges all modes contained in t into row indices of the matrix, and the other modes into column indices. We then consider a hierarchy of matrices U^t whose columns span the image of $X^{(t)}$ for each $t \subset T$. Hence, U_t has exactly $r_t = rank(X^{(t)})$ columns. The rank tuple $(r_t)_{t \in T}$ is called the HT-rank of X.

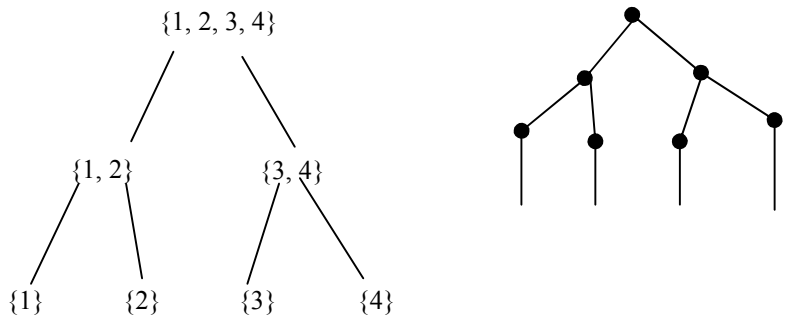

Figure 2 – Left: Binary tree representing mode splitting for HT decomposition. Right: Tensor network diagram representing a tensor in HT decomposition.

The following nestedness property allows for the implicit storage of $(U_t)_{t \supset T}$, and thus of the tensor X:

For $t = t_l \cup t_r, t_l \cap t_r$, there exists a matrix Bt such that

$$Ut = (Utr \cdot Utl)\, Bt\,, \tag{9}$$

For simplicity, we have assumed that the ordering of the modes in the tree T is such that all modes contained in t_l are smaller than the modes contained in t_r. The relation (9) implies that it suffices to store the basis matrices U_t only for the leaf nodes t = {1}; {2},…{d}, and B_t for all other nodes in T . The resulting storage requirements are $O(dnr + dr^3)$, when assuming $r \equiv r_t$ and $n \equiv n_\mu$.

Similarly to the Tucker and TT decompositions, a quasi-best approximation in the HT decomposition for a given HT-rank can be obtained from the SVDs of $X^{(t)}$. Algorithms that avoid the explicit computation of these SVDs when truncating a tensor that is already in HT decomposition are discussed in [12, 16]. As for the TT decomposition, the set of tensors having fixed HT-rank forms a smooth manifold [17].

The tensor network corresponding to the HT decomposition is always a binary tree, see also the right plot of Figure 2. Such tensor tree networks had already been discussed in [18] (without the basis matrices at the leafs). Moreover, the so called multilayer multi-configuration time-dependent Hartree method (MLMCTDH) introduced in [19] makes use of a decomposition based on general trees instead of binary trees.

When allowing for general trees, tensor tree networks include the Tucker decomposition from Section 2.2 as a (quite particular) special case. In the case of a degenerate tree, where at each level, one mode is split from the remaining modes, the HT decomposition becomes equivalent to a variant of the TT decomposition discussed in [12]. In contrast to the TT decomposition defined in (5), this variant features additional basis matrices, which may reduce the storage cost for large n_. A discussion on the difference between the ranks for the HT and TT decompositions can be found in [6].

To summarize, the TT and HT decompositions have similar properties and serve similar purposes. While the HT decomposition offers greater flexibility, the simpler structure of the TT decomposition may lead to simplifications in an implementation. However, it seems premature to give an authorative comparison of the two decompositions. Unless there is an underlying topological structure, as in strongly correlated quantum mechanical systems, it appears to be difficult to decide a priori which decomposition should be preferred for approximating a given function-related tensor.

2.5 More general tensor network formats

Motivated by an underlying topology describing interactions, tensor networks beyond trees have been considered in the context of renormalization group methods for simulating strongly correlated quantum spin systems. Well-known examples include the so called projected entangled-pair states (PEPS) [20] and the multiscale entanglement renormalization ansatz (MERA) [6]. Both, PEPS and MERA contain cycles in the tensor network. Tree-structured tensor networks, as the hierarchical Tucker and the TT format, are closed [17] in the sense that tensors with ranks at most r_μ form a closed set in R^{nxtd}. In general, this statement does not hold for tensor networks containing cycles [6].

Possibly for this reason, more general networks have not yet been considered to a large extend in the numerical analysis community for, e.g., the solution of high-dimensional PDEs, but see [21] for some recent mathematically oriented work.

2.6 Hybrid formats

Adding to the diversity of the formats discussed above, it is possible and sometimes useful to combine different low-rank formats. One popular combination is the Tucker format combined with the CP format for the approximation of the core tensor [22], see [24] for other variations of Tucker and CP. In [23], combinations of low-rank tensor formats with hierarchical matrices are investigated.

Conclusion

Presentation of television images by the 3D meshes is the most acceptable result. However, for the transmission of the vertices coordinates, which are called geometrical data, requires too broadband channels. The analysis of geometric data conversion 3D meshes in spectral space has shown the advantages of using tensor decomposition. Competing spectral transforming method can

be called the principal components analyses. However, using Tucker tensor decomposition we come to the hierarchical scheme. Then we can talk about nested spectral space with lower dimension. In terms of television images such hierarchical space allows to implement various sharpness and resolution, depending on the bandwidth and performance of computing devices.

REFERENCES:

1. S. Gumbold, W. Strasser, “Real Time Compression of Triangle Mesh Connectivity”, *Computer Graphics Proceedings, Annual Conference Series, (Proc. of ACM SIGGRAPH 98)*, pp. 133-140, 1998.

2. K. Müller, A. Smolic, M. Kautzner, P. Eisert, and T. Wiegand Predictive Compression of Dynamic 3D Meshes Image Processing, 2005. ICIP 2005. IEEE International Conference, I-621-4

3. Ming Zhong, Hong Qin “Sparse approximation of 3D shapes via spectral graph wavelets” The Visual Computer June 2014, Volume 30, Issue 6-8, pp 751-761 2014

4. M. Alexa and W. Müller, “Representing Animations by Principal Components”, *Proc. of EUROGRAPHICS 2000*, vol. 19, no. 3, pp. 411-418, 2000.

5. Jung-Shiong Chang, Arthur Chun-Chieh Shih, Hsiao-Rong Tyan And Wen-Hsien Fang “Principal Component Analysis-based Mesh Decomposition” Journal of Information Science and Engineering 25, 971-987 (2009)

6. Lars Grasedyck, Daniel Kressner, and Christine Tobler “A literature survey of low-rank tensor approximation techniques” Cornell University Library Feb 2013 20p http://arxiv.org/abs/1302.7121

7. R. Andreev and C. Tobler, “Multilevel preconditioning and low rank tensor iteration for space-time simultaneous discretizations of parabolic PDEs”, Tech. Rep. 2012–16, Seminar for Applied Mathematics, ETH Z¨urich, 2012, Available at http://www.sam.math.ethz.ch/reports/2012/16.

8. A.Arnold and T. Jahnke, “On the approximation of high-dimensional differential equations in the hierarchical Tucker format”, Tech. rep., KIT, Karlsruhe, Germany, 2012.

9. I.V. Oseledets, “Approximation of 2dx 2d matrices using tensor decomposition”, SIAM J. Matrix Anal. Appl. 31(4), 2130–2145 (2010).

10. I.V. Oseledets, “Constructive representation of functions in low-rank tensor formats”, Constr. Appr. 37(1), 1–18 (2013).

11. I.V. Oseledets and E. E. Tyrtyshnikov, “TT-cross approximation for multidimensional arrays”, Linear Algebra Appl. 432(1), 70–88 (2010).

12. Oseledets, Tensor-train decomposition, SIAM J. Sci. Comput. 33(5), 2295–2317 (2011).

13. R. Penrose, “Applications of negative dimensional tensors”, in: Combinatorial Mathematics and its Applications(Proc. Conf., Oxford, 1969), (Academic Press, London, 1971), pp. 221–244

14. F. Fr¨owis, V. Nebendahl, and W. D¨ur, “Tensor operators: Constructions and applications for long-range interactionsystems”, Phys. Rev. A 81(Jun), 062337 (2010).

15. D. Perez-Garcia, F. Verstraete, M. M. Wolf, and J. I. Cirac, “Matrix product state representations”, Quantum Info. Comput. 7(5), 401–430 (2007).

16. W. Hackbusch and S. K¨uhn, “A new scheme for the tensor representation”, J. Fourier Anal. Appl. 15(5), 706–722(2009).

17. A. Falco,W. Hackbusch, and A. Nouy, Geometric structures in tensor representations, Tech. Rep. 9, MPI MISLeipzig, 2013.

18. Y.Y. Shi, L. M. Duan, and G. Vidal, “Classical simulation of quantum many-body systems with a tree tensor network”, Physical Review A 74(2), 022320 (2006).

19. H. Wang and M. Thoss, “Multilayer formulation of the multiconfiguration time-dependent Hartree theory”, J.Chem. Phys 119(3), 1289–1299 (2003).

20. F. Verstraete and J. I. Cirac, “Renormalization algorithms for quantum-many body systems in two and higher dimensions”, Tech. Rep. arXiv:cond-mat/0407066, 2004.

21. S. Handschuh, “Changing the topology of tensor networks”, Tech. Rep. 15, MPI MIS Leipzig, 2012.

22. B. N. Khoromskij and V. Khoromskaia, “Low rank Tucker-type tensor approximation to classical potentials”, Central European Journal of Mathematics 5, 523–550 (2007).

23. W. Hackbusch, B. N. Khoromskij, and E. E. Tyrtyshnikov, “Hierarchical Kronecker tensor-product approximations”, J. Numer. Math. 13(2), 119–156 (2005).

24. T. G. Kolda and B.W. Bader, “Tensor decompositions and applications”, SIAM Review 51(3), 455–500 (2009).