

# WAVELET-BASED HYBRID MULTILINEAR MODELS FOR MULTIDIMENSIONAL IMAGE APPROXIMATION

Qing Wu<sup>†</sup>

Chun Chen<sup>‡</sup>

Yizhou Yu<sup>†</sup>

<sup>‡</sup> College of Computer Science, Zhejiang University

<sup>†</sup> University of Illinois at Urbana-Champaign

## ABSTRACT

The wavelet transform hierarchically decomposes images with prescribed bases, while multilinear models search for optimal bases to adapt visual data. In this paper, we integrate these two approaches to compactly represent 2D images and 3D volume data. Once a wavelet (packet) decomposition has been performed, the coefficients are subdivided into small blocks most of which have small energy and are pruned. Surviving blocks usually exhibit strong redundancy among different channels and subbands. To exploit this property, we organize the surviving blocks into small tensors, group the tensors into clusters using an EM algorithm, and compactly approximate each cluster using tensor ensemble approximation. Experimental results on images and medical volume data indicate that our approach achieves better approximation quality than wavelet (packet) transforms.

**Index Terms**— Hybrid multilinear models, multiscale analysis, wavelet transform, adaptive bases, tensor ensemble approximation

## 1. INTRODUCTION

With modern digital imaging technologies, visual data such as images, videos and multidimensional medical images have been generated, processed and transmitted with tremendous rate and scale. Much research has hence been performed to efficiently and compactly represent these data. As the first step, most methods transform the input dataset so that energy is concentrated in the coefficients of a relatively small number of bases to facilitate subsequent encoding/decoding or analysis.

The wavelet transform has been extensively investigated for image analysis and compression [1, 2]. Using carefully designed bases, a.k.a. scaling functions and wavelets, the input signal is decomposed into a low-frequency subband and a hierarchy of higher-frequency subbands. Wavelets are capable of capturing discontinuities so that energy in those high-frequency subbands are highly concentrated. As a generalization of the wavelet transform, a wavelet packet transform decomposes both low-frequency and high-frequency subbands recursively and searches for the best combination of different scales to improve compactness [3, 4]. Even though the wavelet (packet) transforms are highly effective, encoding the remaining sparse high-energy coefficients remains an active research topic.

Multilinear models based on tensor approximation have achieved impressive compression ratios for high-dimensional visual data [5, 6]. Instead of using prescribed bases, they search for optimal basis matrices specifically tailored for the input data. Hierarchical tensor approximation that recursively subdivides residual tensors and approximates them as an ensemble has been proposed in [6], further improving PSNR under the same compression ratio. Nevertheless, adaptive basis matrices need to be stored for each individual

dataset, which could impose a significant overhead. The compression performance of tensor approximation becomes worse than the wavelet transform on low-dimensional datasets such as 2D images. An additional problem exhibited by tensor approximation is that although the basis matrices being optimal in terms of MSE, they cannot effectively preserve sharp discontinuities and, thus, make compressed results less visually appealing.

In this paper, we propose hybrid multilinear models in the wavelet domain to harness the power of both wavelet (packet) transforms and tensor approximation. Our method first performs a wavelet (packet) transform. The coefficients in the high-frequency subbands exhibit strong correlation across different subbands and color channels. Multilinear models are very effective in simultaneously removing redundancies across multiple modes, including spatial locations, subbands and color channels. Therefore, our method spatially subdivides the high-frequency subbands, and forms small tensors with the aforementioned modes. Most of the formed tensors have negligible energy and thus pruned. Non-negligible tensors are grouped into clusters using an EM algorithm, so that tensors in each cluster can be approximated efficiently as an ensemble.

We have tested our new hybrid method on both natural and medical images, achieving more compact approximation than wavelet transforms, wavelet packet transforms or tensor approximations alone. Meanwhile, our compressed data retains high visual fidelity because the initial wavelet (packet) transform can well preserve discontinuities. In addition, our method significantly reduces basis overhead because we only need to approximate a small number of surviving tensors after pruning and the tensors in the same cluster share basis matrices.

## 2. BACKGROUND

As in [7], a real  $N$ th-order tensor  $\mathcal{A} \in \mathfrak{R}^{n_1 \times n_2 \times \dots \times n_N}$ , can be considered as an element of a composite vector space,  $R^{n_1} \otimes R^{n_2} \otimes \dots \otimes R^{n_N}$ , where we call each  $R^{n_i}$  an elementary vector space, and  $\otimes$  denotes the Kronecker product of vector spaces. The dimensionality of the  $i$ -th elementary vector space is  $n_i$ .

A rank- $(r_1, r_2, \dots, r_N)$  approximation of  $\mathcal{A}$  is formulated as

$$\tilde{\mathcal{A}} = \mathcal{B} \times_1 \mathbf{U}^{(1)} \times_2 \mathbf{U}^{(2)} \times \dots \times_N \mathbf{U}^{(N)}, \quad (1)$$

where each basis matrix  $\mathbf{U}^{(i)} \in \mathfrak{R}^{n_i \times r_i}$ , the core tensor  $\mathcal{B} \in \mathfrak{R}^{r_1 \times r_2 \times \dots \times r_N}$ , and  $\times_k$  represents  $k$ -mode product of a tensor by a matrix<sup>1</sup>. The column vectors of each  $\mathbf{U}^{(i)}$  are orthonormal to each other. Once the basis matrices are known,

<sup>1</sup>The  $k$ -mode product of a tensor  $\mathcal{A}$  by a matrix  $\mathbf{U} \in J_k \times n_k$ , denoted by  $\mathcal{A} \times_k \mathbf{U}$ , is defined as a tensor with entries:  $(\mathcal{A} \times_k \mathbf{U})_{i_1 \dots i_{k-1} j_k i_{k+1} \dots i_N} = \sum_{i_k} a_{i_1 \dots i_N} u_{j_k i_k}$ .

$\mathcal{B} = \mathcal{A} \times_1 \mathbf{U}^{(1)T} \times_2 \mathbf{U}^{(2)T} \times \dots \times_N \mathbf{U}^{(N)T}$ . When  $r_1, r_2, \dots, r_N$  are sufficiently small, the core tensor and the basis matrices together give rise to a compact representation. The Alternative Least Square (ALS) algorithm was used in [8, 7] to solve the optimal basis matrices given their reduced ranks.

Note that existing tensor approximation techniques [7, 5, 6] directly operate on the original input data while our new method in this paper applies tensor approximation to the coefficients resulting from a multi-level wavelet (packet) transform. As discussed earlier, such an approach can harness the power of both wavelet (packet) transforms and tensor approximation.

### 3. SUBBAND DATA ORGANIZATION

The wavelet transform decomposes a  $N$ -dimensional image into one low-frequency subband and a hierarchy of levels consisting of  $2^N - 1$  high-frequency subbands each. We subdivide the subbands along spatial axes into small blocks and approximate surviving blocks after pruning. Since corresponding blocks within different subbands or color channels have strong correlation, they usually exhibit similar patterns. Therefore, we form third-order tensors within each level of a wavelet transform by placing together corresponding blocks from all subbands and color channels at that level. Note that the tensors we form in this paper are always third-order: the first mode represents different spatial locations within a block, the second one represents different subbands, and the third one represents different color channels. The third mode could be degenerate for single-channel inputs.

For example, if the input is a  $256 \times 256 \times 256$  RGB volume dataset and we first decompose it using a 3-level wavelet transform, for each channel we obtain one  $32 \times 32 \times 32$  low-frequency subband, seven  $32 \times 32 \times 32$  third-level high-frequency subbands, seven  $64 \times 64 \times 64$  second-level high-frequency subbands and seven  $128 \times 128 \times 128$  first-level high-frequency subbands. If we subdivide each high-frequency subband into  $2 \times 2 \times 2$  blocks and flatten these blocks into 8-dimensional vectors, we obtain 4096  $8 \times 7 \times 3$  tensors for the third level before pruning, and 32768 and 262144 tensors of the same size for the second and first levels, respectively.

In terms of wavelet packet transforms, since subbands are adaptively selected from a full subband tree, each level contains a variable number of subbands which come from the same level of the tree and have the same size. The rest of the data organization steps are similar to those for wavelet transforms. Note that one level could be entirely missing in a wavelet packet hierarchy. For example, if we decompose the input dataset into a 3-level subband tree and the adaptive algorithm chooses all (and only) subbands on the third level, then the wavelet packet hierarchy contains no subbands on the first and second level.

## 4. HYBRID MULTILINEAR MODELS

### 4.1. Tensor Approximation

After pruning, tensors with significant energy are approximated using rank- $(r_1, r_2, \dots, r_N)$  approximation. Determining optimal reduced ranks in tensor approximation is in general nontrivial and has not been well addressed in previous work. In this paper, we use an error threshold  $\epsilon$  to control the reduced ranks. Suppose we have a tensor  $\mathcal{A}_{n_b \times n_s \times n_c}$ , where  $n_b$  represents the number of pixels/voxels in a block,  $n_s$  denotes the number of high-frequency subbands on each level, and  $n_c$  indicates the number of channels. The tensor is first decomposed using an  $N$ -mode SVD [9]:

$$\mathcal{A}_{n_b \times n_s \times n_c} = \mathcal{B}_{n_b \times n_s \times n_c} \times_1 \mathbf{U}_{n_b \times n_b}^{(b)} \times_2 \mathbf{U}_{n_s \times n_s}^{(s)} \times_3 \mathbf{U}_{n_c \times n_c}^{(c)} \quad (2)$$

In 2D cases, the core tensor is a diagonal matrix with large singular values at the top left corner. In higher-dimensional cases, the core tensor is not sparse, but most significant coefficients still concentrate near the top left corner. As a result, we choose the ranks  $(r_b, r_s, r_c)$  by performing the following minimization of the total number of bits in the reduced data so that approximation MSE is controlled by  $\epsilon^2$  and basis overhead is minimized:

$$\begin{aligned} \min_{r_b, r_s, r_c} (n_b r_b + n_s r_s + n_c r_c) q_b + r_b r_s r_c q_c, \quad \text{s.t.} \\ \sum_{i=1}^{r_b} \sum_{j=1}^{r_s} \sum_{k=1}^{r_c} \mathcal{B}_{ijk}^2 \geq \sum_{i=1}^{n_b} \sum_{j=1}^{n_s} \sum_{k=1}^{n_c} \mathcal{B}_{ijk}^2 - n_b n_s n_c \epsilon^2, \\ 1 \leq r_b \leq n_b, 1 \leq r_s \leq n_s, 1 \leq r_c \leq n_c, \end{aligned} \quad (3)$$

where  $q_b$  and  $q_c$  represent the number of quantization bits for basis matrix entries and core tensor coefficients, respectively. In the above optimization, we need to choose an optimal rank for each mode, and for the  $i$ -th mode we have  $n_i$  possible ranks  $(1, 2, \dots, n_i)$  to choose from. So the complexity of an exhaustive search is  $\prod_{i=1}^N n_i$ . Fortunately, no matter what dimensionalities the input dataset has, we always subdivide the subbands to form small third-order tensors. Thus, we find optimal reduced ranks by an exhaustive search of all possible combinations.

With the optimal reduced ranks, we can truncate the basis matrices and the core tensor accordingly to obtain the optimal approximation:

$$\mathcal{A}_{n_b \times n_s \times n_c} \approx \mathcal{B}_{r_b \times r_s \times r_c} \times_1 \mathbf{U}_{n_b \times r_b}^{(b)} \times_2 \mathbf{U}_{n_s \times r_s}^{(s)} \times_3 \mathbf{U}_{n_c \times r_c}^{(c)} \quad (4)$$

### 4.2. Tensor Ensemble Approximation

Although most tensors are pruned before approximation, it is still inefficient to approximate the surviving tensors individually, which requires three basis matrices for each tensor. An efficient way is to pack all tensors together and approximate them as an ensemble [6], so that all tensors share the same basis matrices, and an additional basis matrix can be introduced to further remove the redundancies among the tensors. Specifically, if we have  $n_t$  tensors  $\mathcal{A}_1, \mathcal{A}_2, \dots, \mathcal{A}_{n_t}$ , we build a new 4D tensor  $\mathcal{A}_{n_b \times n_s \times n_c \times n_t}$  so that  $\mathcal{A}_{:, :, :, i} = \mathcal{A}_i, 1 \leq i \leq n_t$ , and perform a rank- $(r_b, r_s, r_c, r_t)$  approximation:

$$\mathcal{A} \approx \mathcal{B} \times_1 \mathbf{U}_{n_b \times r_b}^{(b)} \times_2 \mathbf{U}_{n_s \times r_s}^{(s)} \times_3 \mathbf{U}_{n_c \times r_c}^{(c)} \times_4 \mathbf{U}_{n_t \times r_t}^{(t)} \quad (5)$$

Similarly,  $r_b, r_s, r_c, r_t$  are determined by:

$$\begin{aligned} \min_{r_b, r_s, r_c, r_t} (n_b r_b + n_s r_s + n_c r_c) q_b + n_t r_t + r_b r_s r_c r_t q_c, \quad \text{s.t.} \\ \sum_{i=1}^{r_b} \sum_{j=1}^{r_s} \sum_{k=1}^{r_c} \sum_{h=1}^{r_t} \mathcal{B}_{ijkh}^2 \geq \sum_{i=1}^{n_b} \sum_{j=1}^{n_s} \sum_{k=1}^{n_c} \sum_{h=1}^{n_t} \mathcal{B}_{ijkh}^2 \\ - n_b n_s n_c n_t \epsilon^2 \\ 1 \leq r_b \leq n_b, 1 \leq r_s \leq n_s, 1 \leq r_c \leq n_c, 1 \leq r_t \leq n_t. \end{aligned} \quad (6)$$

Since  $n_b, n_s$  and  $n_c$  are all small, it is again feasible to perform an exhaustive search for the reduced ranks.

### 4.3. Clustering

The above multilinear model works well if the tensors follow a unimodal distribution. In a general case, it is more reasonable to group them into clusters and approximate every cluster as an ensemble. Tensors in the same cluster must have strong correlations so that they can share basis matrices.

Clustering is a typical "chicken-and-egg" problem. If we know the membership of the tensors, the basis matrices could be found using  $N$ -mode SVD and exhaustive rank search; and conversely, if we know the basis matrices for each cluster, the tensors could easily be assigned to the cluster where the approximation error is minimal. For example, if  $\mathbf{U}_i^{(b)}$ ,  $\mathbf{U}_i^{(s)}$ ,  $\mathbf{U}_i^{(c)}$ ,  $\mathbf{U}_i^{(t)}$  are basis matrices for cluster  $i$ , the approximation MSE of tensor  $\mathcal{A}_j$  is computed using the first three basis matrices:

$$MSE = \|\mathcal{A}_j - \mathcal{B}_j \times_1 \mathbf{U}_i^{(b)} \times_2 \mathbf{U}_i^{(s)} \times_3 \mathbf{U}_i^{(c)}\|^2 \quad (7)$$

where  $\mathcal{B}_j = \mathcal{A}_j \times_1 \mathbf{U}_i^{(b)T} \times_2 \mathbf{U}_i^{(s)T} \times_3 \mathbf{U}_i^{(c)T}$ .

This type of problems are typically solved with Expectation Maximization [10, 11]. In our scenario, an initial classification is specified and basis matrices are computed for each initial cluster with respect to Eq. (5). Then we iteratively optimize the classification and the basis matrices. In each iteration, all tensors are re-classified into the cluster with the minimum approximation MSE (Eq. (7)), and the basis matrices are updated using the new classification. The EM algorithm is described in Algorithm 1.

---

**Algorithm 1** Cluster( $\mathcal{A}_1, \mathcal{A}_2, \dots, \mathcal{A}_{n_t}, m$ )

---

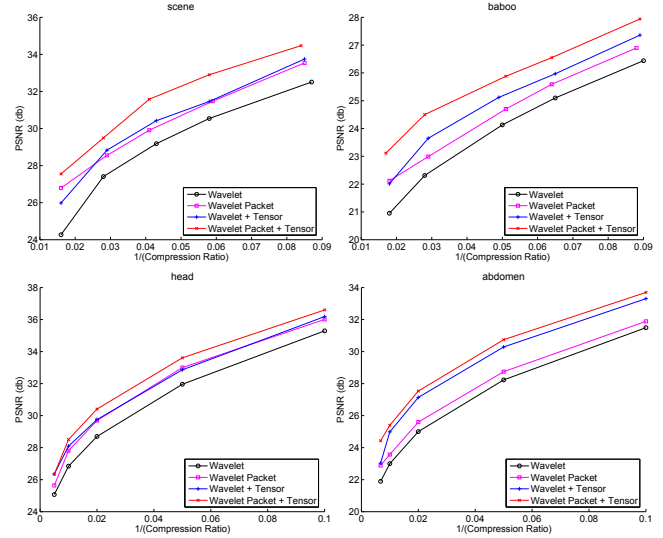
- 1: Initially classify  $\mathcal{A}_1, \mathcal{A}_2, \dots, \mathcal{A}_{n_t}$  into  $m$  clusters  $C_1, C_2, \dots, C_m$ ;
  - 2: **repeat**
  - 3:   **for**  $i = 1$  to  $m$  **do**
  - 4:     Compute ranks for each dimension using Eq. (6);
  - 5:     Compute basis matrices  $\mathbf{U}_i^{(b)}$ ,  $\mathbf{U}_i^{(s)}$ ,  $\mathbf{U}_i^{(c)}$ ,  $\mathbf{U}_i^{(t)}$  for  $C_i$  using ALS;
  - 6:   **end for**
  - 7:   **for**  $i = 1$  to  $n_t$  **do**
  - 8:     **for**  $j = 1$  to  $m$  **do**
  - 9:       Compute approximation error of  $\mathcal{A}_i$  in  $C_j$  using Eq. (7);
  - 10:     **end for**
  - 11:     Move  $\mathcal{A}_i$  into the cluster corresponding to the minimum MSE;
  - 12:   **end for**
  - 13: **until** classification is unchanged
  - 14: **return**  $C_1, C_2, \dots, C_m$ ;
- 

Note that  $m$  should not be large, otherwise the basis overhead could still be significant because we need a set of basis matrices for each cluster. In our experiments we try  $m = 1, 2, \dots, 8$  and choose the one having highest compactness. For initialization, we vectorize all the tensors and use GPCA [12] to obtain a more reasonable classification than a random scheme.

## 5. EXPERIMENTAL RESULTS

We have implemented the hybrid models with both wavelet transforms and wavelet packet transforms. For wavelet packet transforms, high-frequency subbands with the same size are collected to form one level in the decomposed hierarchy. In our experiments, we have tested our method on 2D images and 3D medical datasets, and compared PSNR with wavelet transform, wavelet packet transform and hybrid multiscale linear models, at the same representation compactness. Compression ratio is defined as the ratio between the total number of bits in the original input data and the number of bits for representing coefficients plus bases in the compressed data:

$$\text{Compression Ratio} = \frac{N_{element} \cdot N_{channel} \cdot q_e}{N_{coef} \cdot q_c + N_{basis} \cdot q_b}, \quad (8)$$



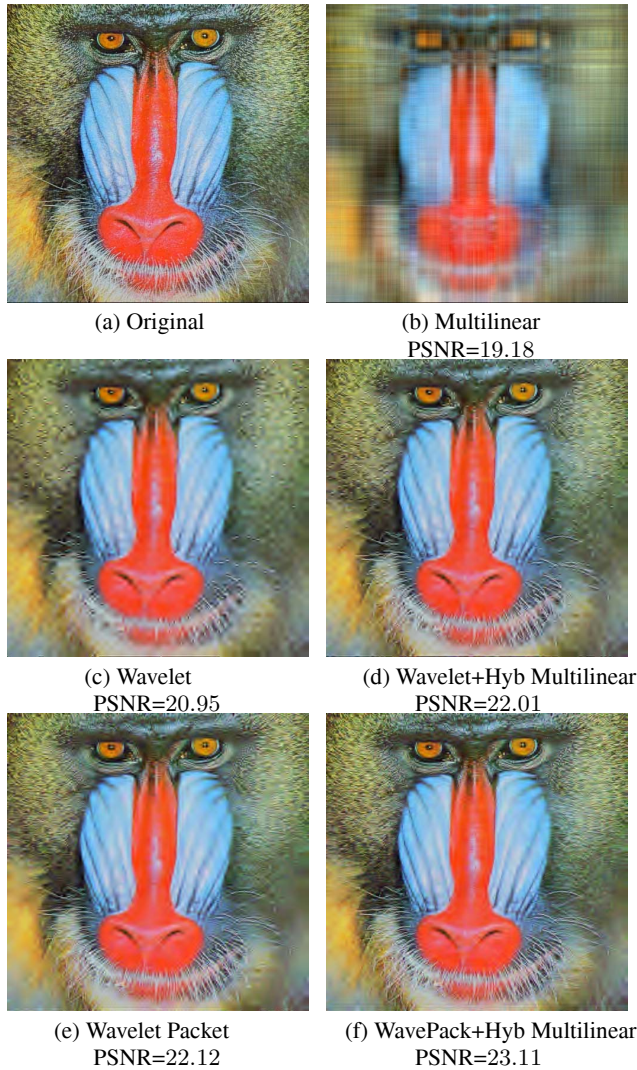
**Fig. 1.** PSNR versus compression ratio curves for wavelet transform, wavelet packet transform, and our method coupled with wavelet transform and wavelet packet transform, respectively. The four diagrams correspond to images, SCENE and BABOO, and Visible Human datasets, HEAD and ABDOMEN.

where  $N_{coef}$ ,  $N_{basis}$ ,  $N_{element}$  and  $N_{channel}$  denote the number of coefficients we retain, the number of entries in basis matrices which is zero for wavelet (packet) transforms, the number of elements in the input data and the number of channels associated with each element, respectively,  $q_c$ ,  $q_b$  and  $q_e$  denote the number of quantization bits used for coefficients, matrix entries and input elements, respectively. In the experimental results shown in this paper, we use uniform quantization with  $q_c = 8$ ,  $q_b = 12$  and  $q_e = 8$ .

Our method has achieved significantly better approximation in terms of PSNR in most of these datasets. Fig. 1 shows the curves of PSNR versus compression ratio for two images, SCENE and BABOO, and two 3D datasets, HEAD and ABDOMEN. In most scenarios, our hybrid multilinear models coupled with wavelet transforms or wavelet packet transforms achieve higher PSNR than existing methods. Performance measurements for pure multilinear tensor approximation are not shown in Fig. 1 because the PSNR they can achieve are typically 3DB lower than the PSNR of our method. Fig. 2 shows the reconstructed images using different approximation methods. In our experiments, hybrid multilinear models coupled with wavelet packet transform improve PSNR by at least 1DB under the same compression ratio over wavelet transforms and wavelet packet transforms. The Visible Human datasets are created and maintained by the United States National Library of Medicine. They are complete, anatomically detailed, three-dimensional representations of the normal male and female human bodies. The images in Fig. 3 show corresponding cross sections extracted from reconstructed datasets. Our wavelet packet based hybrid multilinear models have clear visual and numerical advantages over existing methods.

## 6. CONCLUSIONS AND FUTURE WORK

In this paper, we have developed hybrid multilinear models based on wavelet (packet) transforms. By exploiting redundancies across multiple modes using hybrid multilinear models, our approach can compactly represent sparse high-energy coefficients resulting from wavelet (packet) transforms. Furthermore, our approximation re-



**Fig. 2.** A comparison on the BABOO image. Hierarchical methods decompose the image into 3 levels of subbands. (b)-(f) share the same compression ratio, which is 50.

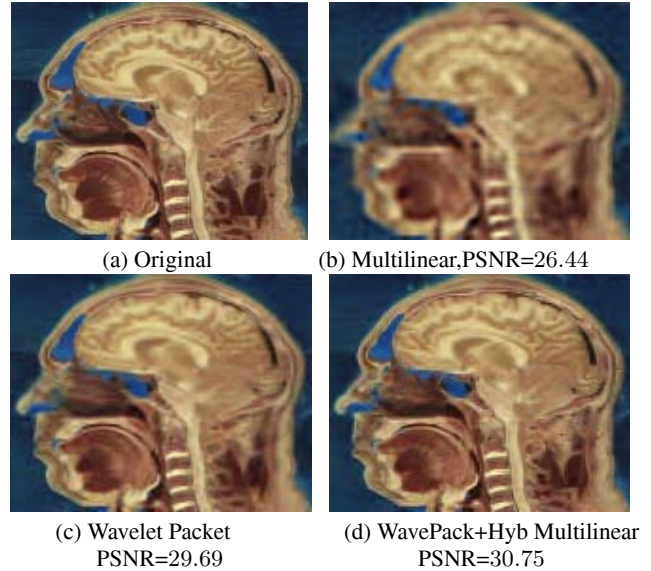
tains high visual fidelity because the initial wavelet (packet) transform can well preserve discontinuities. In addition, our method significantly reduces basis overhead by collectively approximate surviving tensors after pruning. In future, we are interested in improving the performance and scalability of our clustering algorithm.

## Acknowledgments

We wish to thank the National Library of Medicine for licensing the Visible Human dataset, and Wei Hong for sharing his code for multi-scale hybrid linear models. This work was partially supported by National Natural Science Foundation of China (60728204/F020404).

## References

[1] R.A. DeVore, B. Jawerth, and B.J. Lucier, "Image compression through wavelet transform coding," *IEEE Transactions on Information Theory*, vol. 38, no. 2, pp. 719–746, 1992.



**Fig. 3.** A comparison of a reconstructed HEAD cross section from multilinear tensor approximation, wavelet packet transform, and hybrid multilinear models based on wavelet packet transform. Hierarchical methods decompose the image into 3 levels of subbands. (b)-(d) share the same compression ratio, which is 50.

[2] M. Vetterli and J. Kovacevic, *Wavelets and Subband Coding*, Prentice-Hall, 1995.

[3] F. Meyer, A. Averbuch, and J.-O. Stromberg, "Fast adaptive wavelet packet image compression," *IEEE Transactions on Image Processing*, vol. 9, no. 5, pp. 792–800, 2000.

[4] A. Jensen and A. la Cour-Harbo, *Ripples in Mathematics: The Discrete Wavelet Transform*, Springer, 2001.

[5] H. Wang, Q. Wu, L. Shi, Y. Yu, and N. Ahuja, "Out-of-core tensor approximation of multi-dimensional matrices of visual data," *ACM Transactions on Graphics*, vol. 24, no. 3, 2005.

[6] Q. Wu, T. Xia, and Y. Yu, "Hierarchical tensor approximation of multidimensional images," in *Proceedings of IEEE International Conference on Image Processing*, 2007, vol. 4, pp. 49–52.

[7] L. De Lathauwer, B. de Moor, and J. Vandewalle, "On the best rank-1 and rank- $(R_1, R_2, \dots, R_n)$  approximation of higher-order tensors," *SIAM J. Matrix Analysis and Applications*, vol. 21, no. 4, pp. 1324–1342, 2000.

[8] P. Kroonenberg and J. de Leeuw, "Principal component analysis of three-mode data by means of alternating least squares algorithms," *Psychometrika*, vol. 45, pp. 324–342, 1980.

[9] L. De Lathauwer, B. de Moor, and J. Vandewalle, "A multilinear singular value decomposition," *SIAM J. Matrix Analysis and Applications*, vol. 21, no. 4, pp. 1253–1278, 2000.

[10] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the em algorithm (with discussion)," *Journal of the Royal Statistical Society: Series B*, vol. 39, pp. 1–38, 1977.

[11] R. Neal and G. Hinton, "A view of the em algorithm that justifies incremental, sparse, and other variants," in *Learning in Graphical Models*, M. I. Jordan, Ed. 1998, Kluwer.

[12] R. Vidal, Y. Ma, and S. Sastry, "Generalized principal component analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 12, 2005.