A Spectral Graph Theory Approach for Data Re-mapping

Zihong Fan and Antonio Ortega
* Signal and Image Processing Institute
Ming Hsieh Department of Electrical Engineering
University of Southern California, Los Angeles, CA, USA

***Abstract*—Interactive navigation of large high-dimensional media datasets aims at allowing viewers to freely navigate content, selecting a subset of the high-dimensional visual data of interest for display. An example application would be remote visualization of an arbitrary 2-D planar cut from a large volumetric dataset with random access. In our previous work, we proposed a client-server based data representation and retrieval system using overlapping rotated tiles to represent the dataset, which leads to lower bandwidth required for accessing a random plane from large volume data. This leads to the question of how best to represent these rotated tiles for compression. We have presented a non-interpolated symmetric mapping algorithm, which maps each voxel in the original image to a rotated Cartesian grid point. In this paper, we will present a tool to analyze and quantify the performance and demonstrate the benefits of our proposed re-mapping algorithm. We will show that in general the more symmetric the mapping is, the better RD performance can be achieved. Our analysis, based on spectral graph theory, could be used for measuring the performance of different mapping algorithms on a grid of any dimension.**

I. Introduction

Interactive navigation of large high-dimensional media datasets aims at allowing viewers to freely navigate content, selecting a subset of the high-dimensional visual data of interest for display. Even in 3 dimensional case, volume visualization with random data access poses significant challenges.

We focus on situations where lower dimensional portions of a dataset need to be accessed. Specifically, we consider cases where arbitrary oblique planes of a 3D volume may need to be extracted and rendered, as is required in some medical imaging applications. An example of oblique plane intersection with the 3D volume is shown in Fig. 1. In many techniques proposed for volumetric image coding [1]–[5], including approaches such as JP3D [6]–[8], some form of random access is provided via a non-overlapped, independently encoded, cuboid tiling. These approaches can be inefficient in the scenario we focus on, because the only useful voxels[1] for each retrieved cubic tile are those near the intersection between the cube and the desired 2D plane.

In our previous work [9], [10], we have proposed a random image retrieval system for a 3D dataset and shown that it is more efficient to use overlapping tiles to represent the dataset. This work demonstrates that in exchange for increased server-side storage, significant reductions in average transmission rate can be achieved relative to the conventional cubic tiling techniques. The run-time memory space can also be reduced accordingly. A key element in our proposed system [9]–[11] involves tiling images or 3D datasets using rotated tiles. That is, in some cases we use square or rectangular tile shapes at an angle with respect to the horizontal/vertical directions. Since we use a standard transform, which assumes pixels are located in a regular grid, these tiles require us to introduce a remapping algorithm, so that pixels in the original grid are mapped to a regular rotated grid before transform. In [12] we studied the mapping algorithm for rotated tile encoding and provided experiments for the 3D case. Our results showed that a non-interpolated symmetric mapping approach, which maps each voxel in the original image to a rotated Cartesian grid point, outperforms tile representation methods based on interpolation and non-symmetric mapping. Remapping without interpolation has been shown to lead to overall better RD performance in other settings [13]. In [12] we show experimentally that more symmetric mappings outperform less symmetric ones. In this paper, we will present a tool to both analyze and quantify the performance and benefits of our proposed re-mapping algorithm. The result will explain that in general the more symmetric the mapping is, the better RD performance that can be achieved. This analysis using spectral graph theory can be used for measuring the performance of different mapping algorithms on a grid of any dimension.

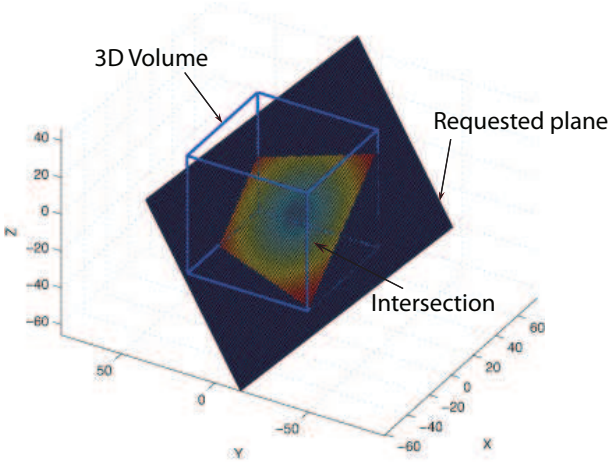


Fig. 1. Random oblique plane acquisition illustration.

[1]A volume element, representing a value on a regular grid in three dimensional space.

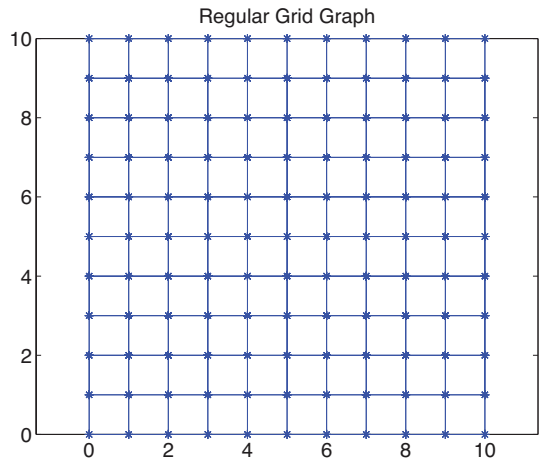


Fig. 2. Regular grid graph. Size: 11×11

In Section II, we define the graphs for the regular grid and the re-mapping grid. In Section III, both an analysis using spectral transforms on graphs as well as experiment results will be shown, followed by the conclusion in Section IV.

II. A GRAPH REPRESENTATION OF REMAPPING PROBLEM

Transforms (e.g., DCT) represent data on a regular grid. When we consider a rotated tile we select a subset of pixels from regular grid. Then the question is how to apply a regular grid transform to this rotated set of pixels. Our solution is to "remap" the pixels in the rotated tile to a regular grid. This is illustrated in Figure 3. The key problem is that when placing these pixels in this new regular grid a geometric distortion is incurred. For example, three pixels that were not aligned in their original location become aligned once they have been placed on the regular grid. We starting by providing a more formal definition of the problem. Then, in Section III, a tool will be introduced to evaluate different mapping algorithms quantify the geometrical distortion.

Let $G = (V, E)$ be a graph, which represents a grid. Vertex set V contains N nodes indexed by $n \in \{1, 2, \ldots, N\}$. The set of edges E represents the connectivity of grid pixels. In Figure 2 for example, except for the pixels on the boundaries, each pixel is connected with its four neighboring pixels (up, down, left and right).

Figures 3 and 4 display rotated regular grids (blue stars) superimposed on top of their corresponding original Cartesian grids (red dots) as well as the mapping relationship (matching pairs) between pixels in the two (blue vectors). When the symmetric mapping algorithm is used, connectivity between pixels in the original Cartesian grid is preserved between the pixels they are matched to, as seen in Figure 3. This leads to a regular pattern to the blue vectors showing the mapping relationships. In contrast to this is the irregular pattern seen in Figure 4, where we see the results of the non-symmetric mapping algorithm.

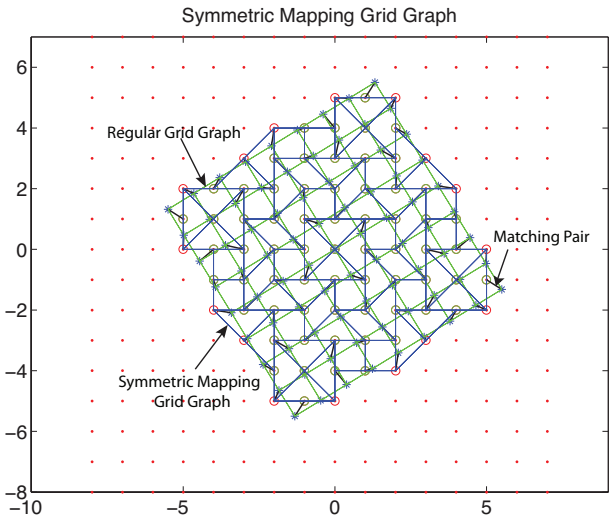


Fig. 3. Mapping of the Cartesian grid pixels onto the rotated grid using the symmetric mapping algorithm. Size: 9×9

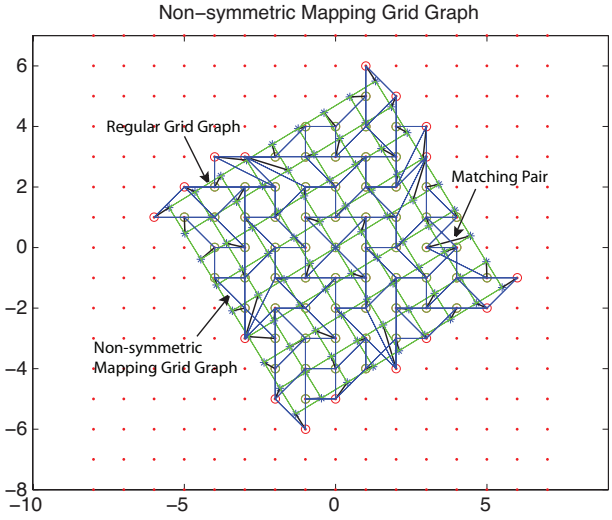


Fig. 4. Mapping the Cartesian grid pixels onto the rotated grid using non-symmetric mapping algorithm. Size: 9×9

Let A_G denote the adjacency matrix of the grid graph, with entries $a_{i,j}$ given by

$$a_{i,j} = \begin{cases} A_G(i,j) = \frac{1}{w(i,j)} & \text{if } (i,j) \in E \\ 0 & \text{otherwise,} \end{cases} \quad (1)$$

where $w(i, j)$ is the Euclidean distance between nodes i and j. $a_{i,j}$ can be seen as the relationship between the two nodes, which is smaller when the distance between nodes i and j is larger and vice versa. The degree matrix of a weighted graph G will be denoted D_G with diagonal elements

$$D_G(i,i) = \sum_j A_G(i,j). \quad (2)$$

The Laplacian matrix of a weighted graph G will be denoted by L_G and is defined as

$$L_G = D_G - A_G. \quad (3)$$

Since L_G for undirected graphs is symmetric and positive semidefinite, the eigenvectors of L_G form an orthonormal basis in $\mathbb{R}^N$. Let $\{\lambda_i, \nu_i\}_{i=1}^N$ be the eigenvalues and vectors of L_G arranged in non-decreasing order with respect to the eigenvalues. Similarly to L_G, we let L_{RG}, L_{SG}, L_{NSG} and $\{\lambda_{ri}, \nu_{ri}\}_{i=1}^n$, $\{\lambda_{si}, \nu_{si}\}_{i=1}^N$, $\{\lambda_{nsi}, \nu_{nsi}\}_{i=1}^N$ denote the Laplacian matrices and the eigenvalues and vectors for the regular grid graph, symmetric mapping graph, and the non-symmetric mapping graph, respectively. We also denote $< v_1, v_2 >$ as inner-product between vectors v_1 and v_2.

III. Analysis and Results

As mentioned earlier, the transform operates on a regular grid, producing transform coefficients that are encoded. Compression is achieved when the data is smooth (or sparse) in the bases of the transform. Note, however, that the geometrical distortion introduced by remapping may mean that even though the original data was smooth (in a traditional 2D or 3D separable transform), the information in the block obtained after remapping may not be as smooth and, thus, overall coding efficiency may suffer. We next propose a metric that will allow us to compare mapping algorithms with different degrees of symmetry, e.g., Figures 3 and 4, and predict which approach will achieve better performance.

L_{RG} is symmetric and positive semidefinite, the eigenvectors of L_{RG} form an orthonormal basis in $\mathbb{R}^N$ and the eigenvectors can be used to characterize various properties of the graph. The transform is based on the regular graph L_{RG}, but the mapping graph grids are not regular any more, which can be seen as geometric distorted from the regular graph. In order to measure the geometric distortion, here we measure the difference of the eigenvectors between L_{RG} and L_{SG}, L_{NSG}.

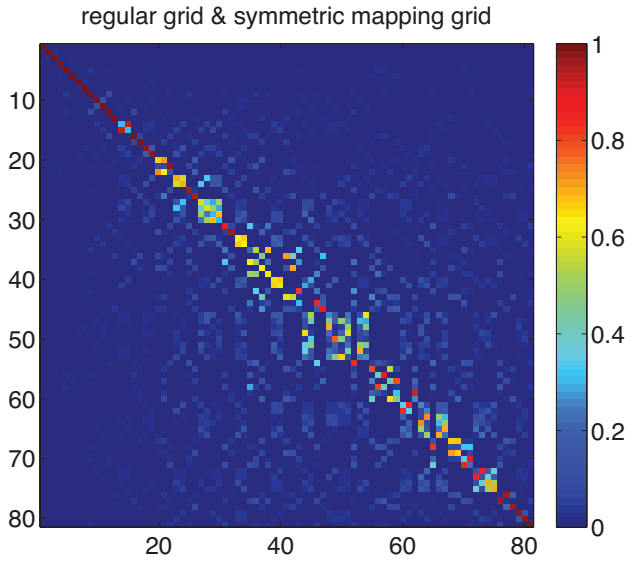
Given that both the symmetric grid graph and the non-symmetric grid graph share the same topology, but different weights, we calculate the inner product between all respective eigenvectors (i.e. regular grid graph vs. symmetric grid graph and regular grid graph vs. non-symmetric grid graph). We let R_S denote the correlation matrix between the eigenvectors of L_{RG} and L_{SG} and let R_{NS} denote the correlation matrix between the eigenvectors of L_{RG} and L_{NSG}. The entries of the correlation matrix R_S are given by

$$R_S\{i,j\} =< \nu_{ri}, \nu_{sj} > \qquad i,j \in \{1,2,\ldots,N\}, \tag{4}$$

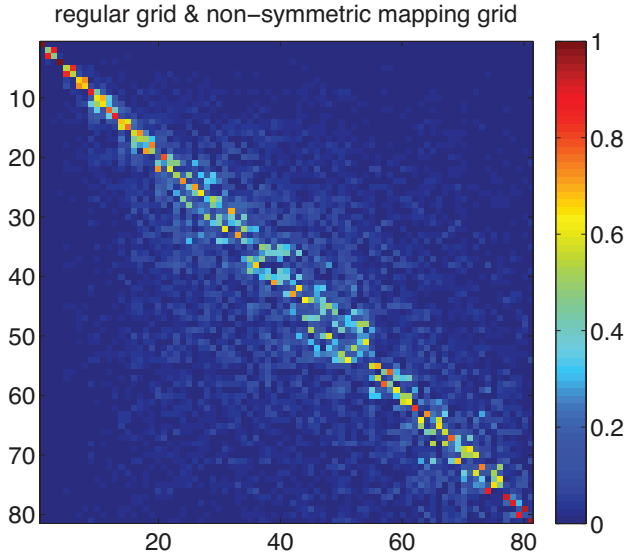
and for the correlation matrix R_{NS} they are given by

$$R_{NS}\{i,j\} =< \nu_{ri}, \nu_{nsj} > \qquad i,j \in \{1,2,\ldots,N\}. \tag{5}$$

Figure 5 shows the absolute values of the correlation matrices R_S and R_{NS}. Figure 5 (a) indicates that the energy of the correlation matrix R_S is more concentrated along the diagonal terms, while (b) indicates that the energy of the correlation matrix R_{NS} is more spread out along the diagonal terms. Assume that a smooth signal is such that it can be described using a small number of eigenvectors of the original graph Laplacian. Then, the intuition is that the effect of remapping is to "disperse" the energy, since eigenvectors in the new graph have non-zero correlation with all eigenvectors corresponding to the original graph. Thus, it would be desirable for this dispersion to be minimal, and hence for the off-diagonal energy to be as small as possible. From this observation, we can conclude that the spectrum of the weighted grid graph does not change as much when using the symmetric mapping algorithm, as compared to the non-symmetric mapping case.



(a) R_S: Absolute value of the Correlation matrices for the symmetric grid graph.



(b) R_{NS}: Absolute value of the Correlation matrices for the non-symmetric grid graph.

Fig. 5. Absolute values of the correlation matrices R_S and R_{NS}.

We notice from Figure 5 that the largest correlation terms are not always exactly on the diagonal of the matrices. For each row i of the correlation matrix R (R can be R_S or R_{NS}),

the maximum correlation term is found and denoted $R_{max}(i)$, i.e.

$$R_{max}(i) = \arg \max_{j \in \{1,2,\ldots,N\}} \{R(i,j)\}. \quad (6)$$

The modified diagonal terms of the correlation matrix are thus the set $\{R_{max}(i)\}_{i=1}^{N}$, the energy of which can be calculated as

$$E_{diag} = \sum_{i=1}^{N} R_{max}(i)^2. \quad (7)$$

Denoting the energy of the modified off-diagonal terms as $E_{\text{off-diag}}$, we have

$$E_{\text{off-diag}} = N - E_{diag}, \quad (8)$$

where the total energy of the correlation matrix is N, since the energy of each row is 1. Hence the percentage of the total energy attributed to the modified off-diagonal terms of the correlation matrix is $\frac{E_{\text{off-diag}}}{N}$. The off-diagonal energy resulting from using either symmetric or non-symmetric mapping is shown in Figure 6. In this figure, different sizes of the grid graphs are used, which are 9×9, 11×11, 13×13, 15×15, 17×17, and 19×19. For all grid graph sizes considered, there is consistently more off-diagonal energy using the non-symmetric mapping algorithm than using the symmetric mapping algorithm.

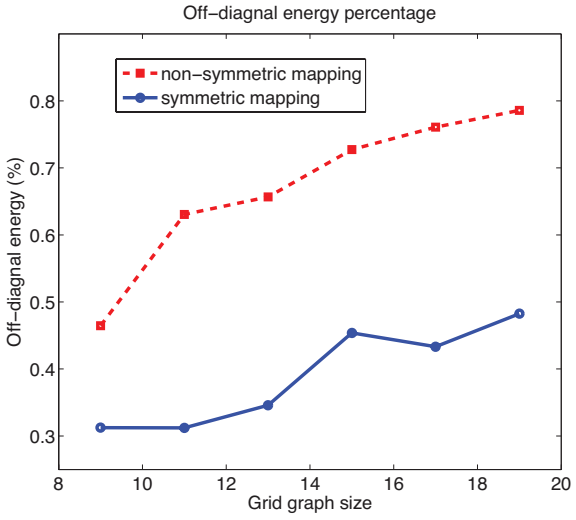


Fig. 6. Percentage of the off-diagonal energy

The experimental results in Figure 7 shows that different non-symmetric mapping algorithms lead to different RD performance in terms of coding, and the symmetric mapping algorithm has better performance than the non-symmetric mapping algorithms. The distortion is measured by using the Cartesian voxel values, which are used for reconstructing the 2D oblique plane from a 3D volume dataset and the distortion only comes from the compression process. For the non-symmetric mapping here, we do select remapping to the nearest neighbor at each point, so that the overall remapping distortion incurred is not very high and the main difference is the lack of symmetry. The experiment uses the same 30 random oblique planes for all different cases. We measure the distortion between the 2D oblique planes, with and without passing through our system. The rate distortion curves in Figure 7 are generated using the quantization step values from 10 to 80, with increment 10. The mean squared error is calculated to measure the distortion and the rate is measured in terms of the average bits per voxel of the transmitted tiles for all random oblique planes.

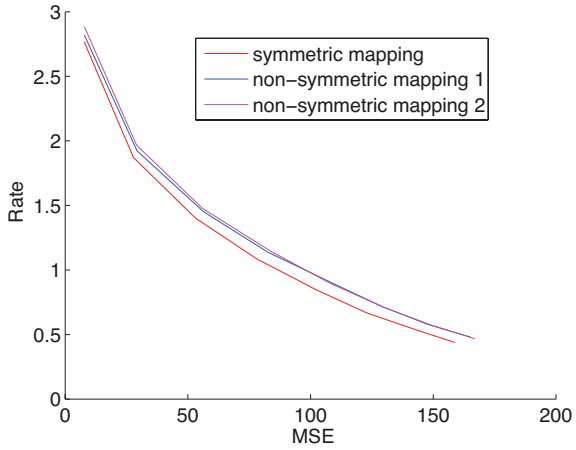


Fig. 7. Symmetric and non-symmetric mapping algorithms' RD curves.

IV. Conclusion

In our previous work, we presented a non-interpolated symmetric mapping algorithm, which maps each voxel in the original image to a rotated Cartesian grid point. Previously, we have shown experimentally that this approach outperforms tile representation methods based on interpolation and non-symmetric mapping. Moreover, especially at high rates, remapping without interpolation was shown to lead to overall better RD performance, with more symmetry leading to better RD performance.

In this paper we have presented a tool to both analyze and quantify the performance and benefits of our proposed re-mapping algorithm. Across a range of grid graph sizes, there is always more off-diagonal energy using non-symmetric mapping algorithm, as compared to the symmetric mapping algorithm. Intuitively, when using the symmetric mapping algorithm the weighted grid graph's spectrum does not change as much as when the non-symmetric mapping algorithm is used. Moreover, this result explains the experiments where it was found that in general the more symmetric the mapping is, the better the RD performance that can be achieved. Our analysis based on spectral graph theory can be used for measuring the performance of different mapping algorithms on a grid of any dimension.

References

[1] I. Ihm and S. Park, "Wavelet-based 3D compression scheme for interactive visualization of very large volume data," *Computer Graphics Forum*, March 1999.

[2] A. Said and W. A. Pearlman, "A new, fast, and efficient image codec using set partitioning in hierarchical trees," *IEEE Trans. Circuits and Systems for Video Technology*, pp. 243–250, June 1996.
[3] Y. Cho, A. Said, and W. A. Pearlman, "Low complexity resolution progressive image coding algorithm: PROGRES(PROGressive RESolution decompression)," in *IEEE ICIP*, 2005.
[4] S. Muraki, "Volume data and wavelet transforms," in *IEEE Trans. Computer Graphics and Application*, July 1993.
[5] Y. Cho and W. A. Pearlman, "Hierarchical dynamic range coding of wavelet subbands for fast and efficient image compression," *IEEE Trans. Image Processing*, vol. 16, no. 2005-2015, Aug 2007.
[6] "Information technology - JPEG 2000 image coding system: Part 10 - extensions for three-dimensional data (jp3d) - fcd v1.0," *ISO/IEC JTC1/SC29/WG1 N4101*, 2006.
[7] T. Bruylants, A. Munteanu, A. Alecu, R. Deklerck, and P. Schelkens, "Volumetric image compression with JPEG2000," in *SPIE The International Society for Optical Engineering*, 2007.
[8] J. P. W. Pluim and J. M. Reinhardt, "Compression of medical volumetric datasets: physical and psychovisual performance comparison of the emerging JP3D standard and JPEG2000," in *SPIE, Medical Imaging 2007: Image Processing.*, 2007, vol. 6512.
[9] Z. Fan and A. Ortega, "Overlapped tiling for fast random access of 3-d datasets," in *Proc of Data Compression Conference (DCC)*, 2009.
[10] Z. Fan and A. Ortega, "Optimization of overlapped tiling for efficient 3d image retrieval," in *Proc of Data Compression Conference (DCC)*, 2010.
[11] Zihong Fan and A. Ortega, "Wavelet-based redundant representation for efficient random access of volumetric images," in *IEEE ICIP*, 2010.
[12] Zihong Fan and A. Ortega, "Mapping data on a rotated grid in high-dimensions for lossless compression," in *IEEE ICIP*, 2011.
[13] S. Lee and A. Ortega, "A novel approach for image compression in digital cameras with bayer color filter array," in *IEEE ICIP*, 2001.