Irregular Morphing for Real-Time Rendering of Large Terrain

Sid'Ali Kalem and Assia Kourgli

USTHB, Facuté d'Electronique et d'Informatique, LTIR, BP32 El Alia Bab-Ezzouar, 16111, Algiers, Algeria
{skalem, akourgli}@usthb.dz

Commission III, WG III/5

KEY WORDS: Large Terrain, Rendering, LOD, View-Dependent, Wavelet, QuadTree

ABSTRACT:

The following paper proposes an alternative approach to the real-time adaptive triangulation problem. A new region-based multi-resolution approach for terrain rendering is described which improves on-the-fly the distribution of the density of triangles inside the tile after selecting appropriate Level-Of-Detail by an adaptive sampling. This proposed approach organizes the heightmap into a QuadTree of tiles that are processed independently. This technique combines the benefits of both Triangular Irregular Network approach and region-based multi-resolution approach by improving the distribution of the density of triangles inside the tile. Our technique morphs the initial regular grid of the tile to deformed grid in order to minimize approximation error. The proposed technique strives to combine large tile size and real-time processing while guaranteeing an upper bound on the screen space error. Thus, this approach adapts terrain rendering process to local surface characteristics and enables on-the-fly handling of large amount of terrain data. Morphing is based-on the multi-resolution wavelet analysis. The use of the D2WT multi-resolution analysis of the terrain height-map speeds up processing and permits to satisfy an interactive terrain rendering. Tests and experiments demonstrate that Haar B-Spline wavelet, well known for its properties of localization and its compact support, is suitable for fast and accurate redistribution. Such technique could be exploited in client-server architecture for supporting interactive high-quality remote visualization of very large terrain.

1. INTRODUCTION

Over recent years, terrain rendering has been used in different fields such as movies, virtual environments, cartography, and games. In particular, it has been intensively developed for real-time outdoor games including flight simulators, driving simulators, and massive multiplayer games. The rapid development in acquisition of topographic maps and cartography has led to the generation of large terrain datasets as height-maps that contain billions of samples. Such terrains datasets exceed the rendering capability of available graphics hardware. Consequently, it is not possible to display 3D scenes represented by too many details in real-time. Thus, adaptive Level-Of-Detail (LOD) rendering has been introduced to simplify the geometry of the heightmap-based terrain. Usually, LOD rendering algorithms represent terrains as triangulated meshes which approximate the surface of the terrain. The aim is to efficiently combine quality rendering and real-time navigation.

The goal of this paper is to point out an alternative approach to the adaptive triangulation problem and large-scale data stream analysis. The discrete wavelet transform (DWT) is exploited as a mathematical framework in order to improve the distribution of the density of triangles inside the tile after selecting appropriate Level-Of-Detail by an adaptive sampling. In some sense, the DWT provides a local spectral estimate of the data and describes local variations which can be harvested to govern the distribution of a surface mesh. Thus, the terrain can be studied at different levels of complexity even within the same tile. The geometry of the tile can appear at different levels of roughness at different locations. In practice, such LOD technique can alleviate aliasing.

The organization of the paper is as follows: In the next section, we give a brief review on terrain representation techniques. In section 3, mathematical framework of the discrete 2D wavelet transform is described. Section 4 is dedicated to the description of the terrain rendering algorithm and the use of the proposed morphing technique. Then we introduce our main contribution, the algorithm that morphs the initial regular grid of the tile to deformed grid in order to minimize geometric approximation error. Section 6 introduces the implementation. And finally, the results are presented in section 7.

2. RELATED WORKS

According to the hierarchical structure used, previous work can be broadly classified into dynamic re-meshing strategies, region-based multi-resolution approaches and regular nested grids. All of those approaches permit continuous LOD rendering of the terrain geometry.

First, view-dependent dynamic re-meshing techniques construct a continuous LOD triangulation in every frame with respect to a given world-space deviation and screen-space error tolerance. Early approaches were based on Triangulated Irregular Networks (TINs) as introduced by Fowler (Fowler, 1979). Those approaches are well-known by their approximation quality. Triangulated irregular networks minimize the amount of triangles to be rendered at a given approximation error, but on the other hand they require quite elaborate data structures that necessitate an intense CPU processing. Consequently, more regular triangulation approaches have been used, for instance, bin-tree hierarchies (Lindstrom, 1996, Duchaineau, 1997) and restricted quad-tree meshes (Pajarola, 1998). In addition to regular hierarchical structures, wavelet transform was used first in (Gross, 1996) to build a hierarchical meshing. The algorithm transforms the initially regular data grid into a QuadTree representation by rejection of unimportant mesh vertices. Lindstrom et al. in (Lindstrom, 2002) described a general and modular framework for out-of-core rendering and management

This contribution has been peer-reviewed.
doi:10.5194/isprsarchives-XLI-B3-755-2016

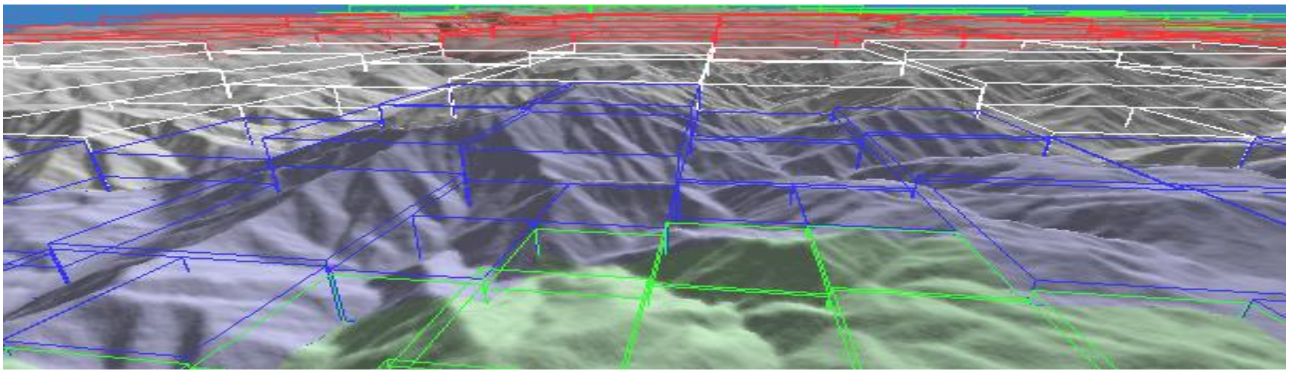

Figure 1. Rendering 3D terrain using region-based multi-resolution approach.

of massive terrain surfaces. It recursively subdivides a triangle mesh defined over regularly gridded data using longest-edge bisection. It organizes the terrain data to improve coherence and reduce the number of paging events from external storage to main memory. In recent works, authors in (Liang, 2010) presented a framework that is implemented entirely in parallel on programmable graphics hardware. The scheme selectively refines and coarsens an irregular hierarchy mesh at the granularity of individual vertices to create meshes that are highly adapted to dynamic view parameters.

Secondly, region-based multi-resolution approaches partition the terrain into tiles that can be processed independently (Ulrich, 2000). To avoid visual artifacts like popping, either geometry-morphing is used or the maximum screen-space error is restricted to one pixel. BDAM (Cignoni, 2003a) and P-BDAM (Cignoni, 2003b) methods exploit bin-tree hierarchies of pre-computed triangulations or batches instead of individual triangles.

Finally, Losasso and Hoppe (Losasso, 2004) even showed that re-meshing can completely be avoided by using a set of nested regular grids centered about the viewer. As the grid resolution decreases with increasing distance to the viewer, approximately uniform screen-space resolution is achieved. This technique caches the terrain in a set of nested regular grids centered about the viewer. Asirvatham and Hoppe further improved this technique in (Asirvatham, 2005) to handle most of computations on the GPU.

Recent advances in graphics hardware processing power have driven to promote region-based multi-resolution approaches (Yacine, 2009, Levny, 2009, Schneider, 2006, Junfeng, 2006, Đurđević, 2011). Usually, they combine LOD representation with near-lossless compression. C-BDAM method, an extension of BDAM and P-BDAM algorithms, was presented by Gobbetti et al. in (Gobbetti, 2006). The method exploits a wavelet-based two stages near-lossless compression technique to efficiently encode the height map data. This method has been optimized later for remote visualization (Bettio, 2007). Hwa et al. in (Hwa, 2005) used GPU-based geometry morphing to render terrain patches of different resolutions from out-of-core multi-resolution data structures. In (Lindstrom, 2010), lossless compression for GPU-based decompression is considered. It uses linear prediction and compresses the residuals into a variable-rate bit stream using an adaptation of the RBUC code. Each terrain triangular patch is decoded sequentially. The parallelism is achieved by simultaneous decompression of multiple patches in the geometry Shader unit. In (Bösch, 2009) a GPU based LOD technique based on a paired multi-resolution tree structure is presented. It follows the idea of batch-based multi-resolution triangulation and rendering. It is based on the concept of LOD triangle K-Patches and M-Blocks of elevation data. In (Đorđe, 2013) the authors introduce an approach suitable for modern GPU architectures. Lossy compression is achieved by approximating the height field with a set of quadratic Bezier surfaces. In addition, lossless compression is achieved by superimposing the residuals over the lossy approximation. Dick et al. (Dick, 2009) propose a method for tile triangulations encoding that enables efficient GPU-based decoding. This allows reducing disk access and CPU-GPU data transfer in terrain rendering. Refer to a nice survey by R. Pajarola and E. Gobbetti (Pajarola, 2007).

Real-time rendering techniques proposed handle either large tile sizes, and do not take fully into account local surface characteristics, or small tile sizes, nevertheless they consider local characteristics of the surface. In fact, the larger the tile is, the more time-consuming surface characteristics handling becomes. On the other hand, local surface characteristics are an important component of 3D terrain rendering process, since different datasets have different characteristics that should be automatically taken into account to guarantee the quality of the rendering.

The proposed technique combines the benefits of both Triangular Irregular Network approach and region-based multi-resolution approach by improving the distribution of the density of triangles inside the tile. Our technique morphs the initial regular grid of the tile to deformed grid in order to minimize approximation error. Thus, this technique strives to combine large tile size and real-time processing. Such technique could be exploited in client-server architecture for supporting interactive high-quality remote visualization of very large terrain.

3. DWT DECOMPOSITION

The DWT decomposition is employed to expand the data and the amplitude of the detail signals is taken as a measure of the local frequency. Thus, applying discrete wavelet transform allows a fast evaluation of the local Level-of-Detail.

The 2D version of the WT expands any finite energy function $f(x,y) \in L^2(R^2)$ using a set of similar basis function $\psi_{(a,b)}(x,y)$. Its generic continuous form description is provided as the following inner product:

$$WT_{f,\psi}(a_x, a_y, b_x, b_y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \psi_{a,b}(x,y)\ f^*(x,y)\, dx\, dy \quad (1)$$

Most discrete formulations of the 2D WT comprise a tensor product extension along with a dyadic scaling of the bases with $a_x = a_y = 2$ and a unit shift $b_x = b_y = 1$ by which the respective bases are derived as follows:

$$\varphi^{2}_{mpq}(x,y) = 2^{-m}\,\varphi(2^{-m}\,x-p)\;\varphi(2^{-m}\,y-q)$$
$$\psi^{2,1}_{mpq}(x,y) = 2^{-m}\,\psi(2^{-m}\,x-p)\;\varphi(2^{-m}\,y-q)$$
$$\psi^{2,2}_{mpq}(x,y) = 2^{-m}\,\varphi(2^{-m}\,x-p)\;\psi(2^{-m}\,y-q) \qquad (2)$$
$$\psi^{2,3}_{mpq}(x,y) = 2^{-m}\,\psi(2^{-m}\,x-p)\;\psi(2^{-m}\,y-q)$$

Where m is the iteration step and φ stands for the so-called scaling function. The upper index 2 denotes the dimension of the bases.

$$f(x,y) = \sum_{p}\sum_{q}\left(c^{M}_{pq}\,\varphi^{2}_{Mpq} + \sum_{m=1}^{M}\left(c^{m,1}_{pq}\,\psi^{2,1}_{mpq} + c^{m,2}_{pq}\,\psi^{2,2}_{mpq} + c^{m3}_{pq}\,\psi^{2,3}_{mpq}\right)\right) \qquad (3)$$

Consequently, any finite energy function f(x,y) ∈ $L^2(R^2)$ can be approximated by the previous bases using the formula (3). $c^{m,i}_{pq}$ denotes the coordinate of f in functional space with respect to the wavelet $\psi^{m,i}_{mpq}$:

$$c^{m,i}_{pq} = \langle f, \psi^{m,i}_{mpq} \rangle \qquad (4)$$

So that, c^{m}_{pq} represents the LL (Low-Low) sub-band containing a low resolution band, and $c^{m,1}_{pq}$, $c^{m,2}_{pq}$, and $c^{m,3}_{pq}$ represent horizontal HL (High-Low), vertical LH (Low-High), and diagonal HH (High-High) sub-bands, respectively. In the decomposition stage, the band c^{m}_{pq} is obtained from c^{m-1}_{pq}, which is derived from c^{m-2}_{pq}. Hence, the decomposition of a two-dimensional multi-scale DWT can be produced by a cascade of decomposition stages (Figure 2).

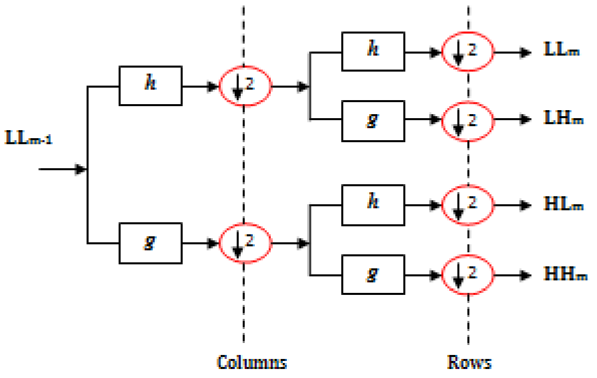

Figure 2. Decomposition filters bank for separable 2D-biorthogonal wavelet transform

In two-dimensional, scaling functions and wavelets can be obtained by extending separable one dimensional base functions as shown in formula 2. This means that two-dimensional wavelet coefficients can be obtained by performing one-dimensional WT successively along the row and column directions (Figure 2).

In figure 2, decomposition filters, h and g, denote one-dimensional low-pass and high-pass filters. The filters bank decomposes this representation into a high-frequency part (HL, LH and HH), based on wavelets, and a low-frequency part (LL) based on coarser scaling functions.

Each scale is composed of the three high-frequency sub-bands (LH, HL and HH), and wavelet coefficients in each sub-band indicate the local variations along one direction (Vertical, horizontal or diagonal). Wavelet coefficient in a more complex region has a high magnitude and, conversely, a low magnitude in a less complex region.

In most computer graphics applications, strict local support along with an appropriate smooth shape, symmetry and fast decay in frequency domain are required. Unfortunately, these properties cannot be satisfied with orthonormal wavelets. Hopefully, the B-Spline wavelets meet the upper requirements, even if their bases are not orthonormal to each other. Compactly supported B-Spline wavelets were introduced by Charles K.Chui and Jian-Zhong Wang (Charles, 1992a, Charles, 1992).
Note that, the B-spline of order 0 is the constant B-spline, and is commonly known as the Haar wavelet:

$$\varphi_0(x) = \begin{cases} 1 & 0 \le x < 1 \\ 0 & otherwise \end{cases} \qquad (5)$$

Recall that, Haar bases are the only symmetric orthonormal biorthogonal wavelets. Haar wavelet is suitable because of its simplicity of implementation, efficiency of localization and its rapidity.

4. RENDERING ALGORITHM

A new region-based multi-resolution approach for terrain rendering is described which improves on-the-fly the distribution of the density of triangles inside the tile after selecting appropriate Level-Of-Detail by an adaptive sampling. This proposed approach organizes the heightmap into a QuadTree of tiles that are processed independently. This technique combines the benefits of both Triangular Irregular Network approach and region-based multi-resolution approach by improving the distribution of the density of triangles inside the tile. Our technique morphs the initial regular grid of the tile to deformed grid in order to minimize approximation error. The proposed technique strives to combine large tile size and real-time processing while guaranteeing an upper bound on the screen space error.

As the terrain height-map is subdivided into large rectangular tiles our technique comes to enhance geometric approximation quality by redistributing the density of triangles inside the large tile. In fact, local surface characteristics are an important component of 3D terrain rendering process, since most terrain data-sets exhibit in the same height-map rough areas and smooth areas that should be taken into account to guarantee the quality of the rendering. Thus, this approach adapts terrain rendering process to local surface characteristics, and enables on-the-fly handling of large amount of terrain data.

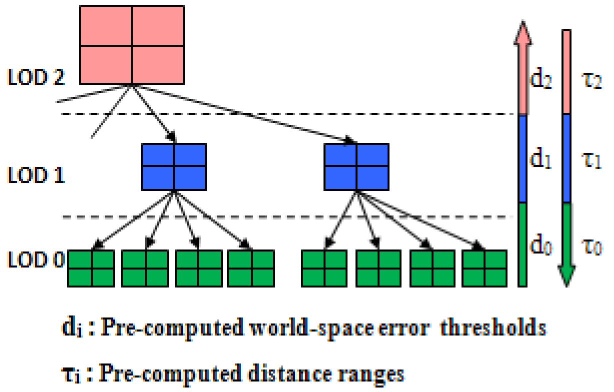
LOD 2
LOD 1
LOD 0
d2
d1
d0
τ2
τ1
τ0
di : Pre-computed world-space error thresholds
τi : Pre-computed distance ranges

Figure 3. Height-map organized as a Quad-Tree of nodes, and corresponding arrays of distance ranges and elevation tolerable errors.

First, the algorithm organizes the height-map into a QuadTree of tiles, which is used to select appropriate tiles from different

LOD layers at run time, as illustrated in figure 1. This structure is commonly used in real-time terrain rendering systems. The QuadTree hierarchy does not store any geometry; instead it stores the position and dimension of each tile with respect to the terrain. The QuadTree structure is generated from the input height-map. It is of constant depth, predetermined by memory and granularity requirements. Once created, the QuadTree does not change unless the source height-map changes. In the QuadTree structure every tile has four child tiles and it covers four times more area than one of its children.

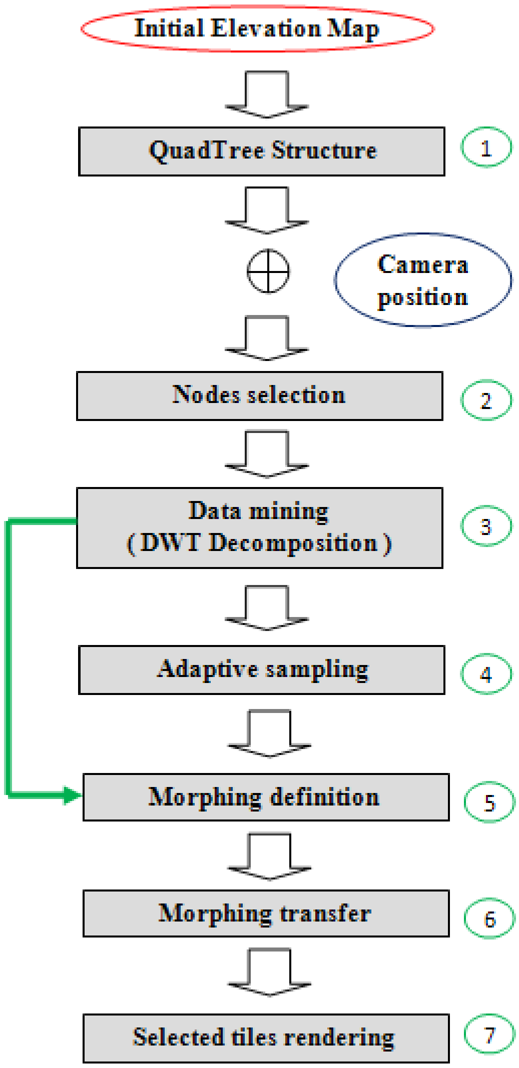

Figure 4. Terrain rendering algorithm.

At run time, the second step of the rendering process is the QuadTree tiles selection. It is performed every time the observer moves, which usually means during every frame. The selection is performed according to the distance between the tile and the camera and the maximum world-space error. At this stage we perform view-frustum culling that eliminates the rendering of non-visible tiles. In order to know which nodes to select, distances covered by each QuadTree layer are pre-calculated before the node selection process is performed. Thus, an array of distance ranges is created which is also used to create an array of world-space error tolerable at each layer of the QuadTree (Figure 3). The array representing the world-space error tolerances in each range of distance will help us to choose the resolution of the regular grid for tiles of the QuadTree.

In the third step (Data mining/training stage), the tile's height-map is first extracted at full resolution. As the datasets of realistic terrains are so big, we use the multi-resolution wavelet decomposition of the full-resolution height-map as training stage to localize the position in the tile where the geometry approximation error could be minimized and to focus control where the maximum of world-space error could be computed. In the following, we will show that the wavelet B-Spline 0(Haar wavelet) is suitable because of its simplicity of implementation, efficiency of localization and their rapidity.

In the fourth step, the rendering algorithm performs an adaptive sampling of the initial full-resolution height-map while respecting the world-space error so that the resolution of the regular grid representing the new height-map guarantees the quality of the triangle approximation of the tile. So, the resolution of the grid to be used for each node of the terrain at run time is stocked in the structure QuadTree for future uses.

After that, the morphing is applied in two steps. First, morphing settings are adjusted for each tile according to local roughness in the CPU. They are based-on the multi-resolution wavelet analysis. Secondly, the morphing is transferred to the initial regular grid in the GPU. The latter is deformed in the GPU in such a way that dense triangular meshes are distributed where the terrain is most rough. In contrast, less dense mesh is distributed where the topography is fairly smooth and only minor differences in the change in elevation exist. The use of the D2WT multi-resolution analysis of the terrain height-map speeds up processing and permits to satisfy an interactive terrain rendering. Tests and experiments demonstrate that Haar B-Spline wavelet, well known for its properties of localization and its compact support, is suitable for fast and accurate redistribution.

The actual rendering is performed in step seven by rendering areas covering selected tiles using grid-meshes representing different resolution from sub-sampled height-map of tiles and deformed in such a way that it improves geometry approximation quality, reading the heightmap in the Vertex Shader, and displacing the mesh vertices accordingly, thus forming the representation of the particular terrain tile.

The quality of the approximation is guaranteed by the use of a maximum screen-space error. Screen-space error is derived at run time from a node bounding volume and its world-space error. To approximate screen space error of the node we use the following equation:

$$\varepsilon_{scr} = \frac{S}{tg(\gamma/2)}\frac{\varepsilon}{d} \qquad (6)$$

Where S is the screen resolution (maximum of horizontal and vertical resolutions), γ is the field of view angle, ε is the node geometric world space error calculated at the pre-processing stage, and d is the distance from the camera to the bounding box.

Since the proposed algorithm provides an accurate estimation of the world-space error bound, the given formula provides guaranteed screen-space error bound of the node. During the rendering we increase the grid resolution of the regions where the screen-space error is greater than defined threshold and decrease it where it does not introduce intolerable error. This simple down-sampling algorithm generates adaptive approximation that satisfies user-defines screen space error threshold. The proposed morphing technique allows to decrease further screen-space error, and thus, improves the geometric approximation quality.

5. MORPHING ALGORITHM

The main contribution of this paper is the idea of using the wavelet coefficients in order to define a geometric morphing that improves the distribution of the density of triangles inside each tile, and thus, enhances the geometric approximation quality of the terrain. So, we present an approach for adaptive surface meshing which enhances the geometric approximation of the terrain's surface inside each tile according to its own local characteristics. The latter are determined by a wavelet representation of the surface data.

The thought is to decompose the initial data set by wavelet transform WT and to analyze the resulting coefficients. In surface regions, where the partial energy of the resulting coefficients is high, fine grain details are localized, and consequently, the density of triangle could be increased to improve the geometric approximation quality. In contrast, where resulting coefficients are smooth surface is localized and density of triangle could be decreased in favor of regions of fine grain details. We must specify that the number of triangle in the grid will not change. The aim is to move triangles inside the tile in such a way that the distribution of triangles much the repartition of LODs inside the tile (Figure 5). Therefore, this technique allows a large-scale data processing.

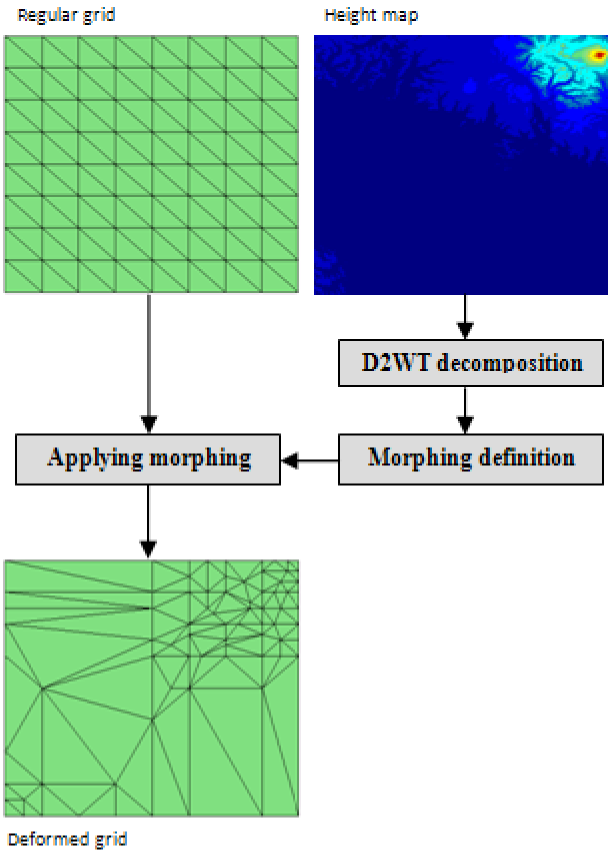

Figure 5. Illustration of the proposed morphing algorithm.

The multi-resolution wavelet decomposition of the full-resolution height-map is used as training stage to localize the position of fine grain details and the positions of smooth surfaces. Wavelet coefficients in each sub-band indicate the local variations. To effectively associate a wavelet coefficient with its corresponding local region, we adopt a rearrangement scheme. The relocated position of a wavelet coefficient is determined by considering its location and the coverage of the corresponding decomposition filter. Since there are three details sub-bands per 2D-DWT decomposition level as explained in section 3, the morphing to be applied to each tile is defined starting for the average of three details sub-bands. Wavelet coefficient in a more complex surface region has a high magnitude and, conversely, a low magnitude in a less complex region. Thus, applying discrete wavelet transform allows a fast evaluation of the local Level-of-Detail. Each level of decomposition ties in one grid mesh of the same resolution as the sub-band.

6. IMPLEMENTATION

The implementation of terrain rendering is written in C++, using DirectX9 as the graphical API and HLSL. It should work on most GPUs that support vertex Shader texture sampling, ones supporting Shader Model 3.0.

To best exploit power of modern GPUs we cache data of terrain elevations in the fast GPU video memory and use it across successive frames. CPU performs QuadTree traversal and selection of appropriate LOD for different areas of the terrain based on node geometric world-space error and distance to camera. CPU also performs view-frustum culling. Thus slow data transfer between CPU and GPU occurs very rarely.

The rendering of each tile is provided by a deformed grid of appropriate resolution generated by the CPU for each tile and transferred to the GPU. The grid is then transformed in the GPU to cover the area of the tile. The GPU samples the elevation map, already down-sampled and morphed by our algorithm and stocked as texture in video memory, to determine the elevation of each vertex in world space. Thus, terrain rendering is provided by render a set of deformed grids at different resolutions that cover the area of the terrain.

The vertex and fragment processors assign elevation and colour for each vertex using cached textures. The LOD in each node is determined without querying adjacent nodes. Furthermore, it reduces communication overhead as a result of transmitting only the layout of nodes to the GPU at each frame to generate the selected LOD representation.

7. RESULTS AND DISCUSSION

For the following, we investigate the efficiency of the morphing technique to improve the distribution of the density of triangles inside the tile. Geometry approximation error is examined for several resolutions of grids (8×8, 16×16, 32×32) before and after applying the proposed morphing technique. The results are conclusive in all resolutions tested and geometry approximation errors are decreased while guaranteeing an upper bound on the screen space error.

In this paper and as mentioned above, B-Spline 0 wavelets (Haar wavelet), well known for its rapidity and its properties of localization, is used for D2WT multi-resolution analysis of the terrain height-map in training stage.

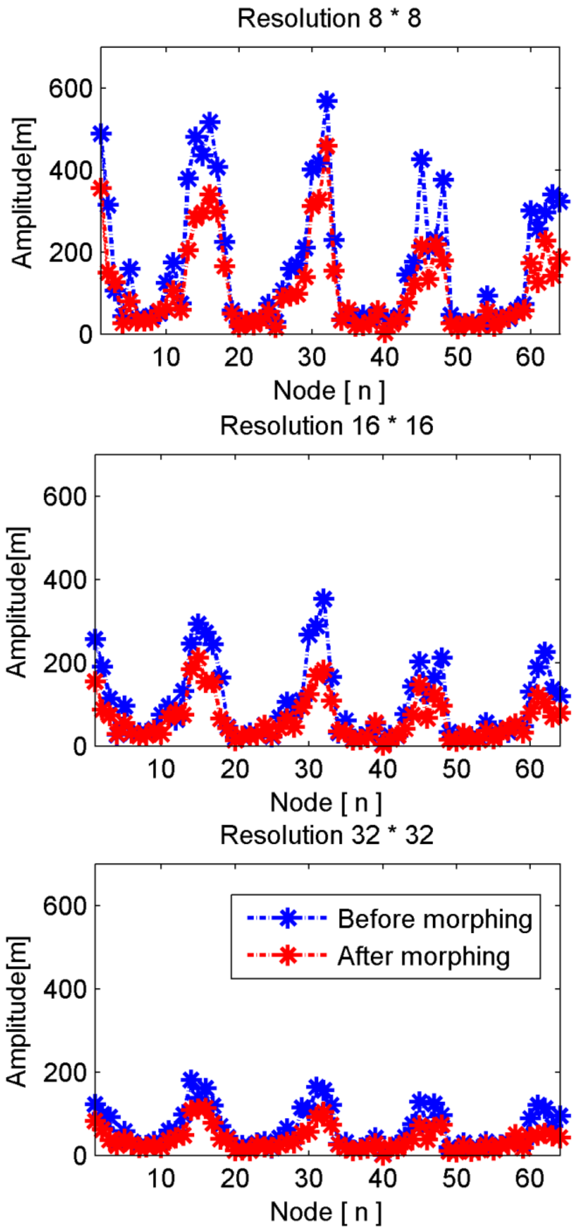

Figure 6. Comparison between the maximum world-space error before and after applying the algorithm.

We use the dataset “Puget Sound” 16385×16385 height map sampled at 10 meters spacing, which is used as the common benchmark, with no normal map, no dynamic lighting, no detail map, and an overlay colour-map with embedded lighting. The dataset is large enough for realistic profiling of performance. Nodes used for measures are nodes of low level of the QuadTree that have 1025 × 1025 size.

In figure 6 maximum approximation error is recorded in meters at several sub-sampling resolutions. According to results depicted and comparing between results before and after applying the morphing algorithm, the proposed technique decreases the maximum world-space errors at all resolutions. We note that the approximation improvement is not constant and depends on the characteristics of each tile.

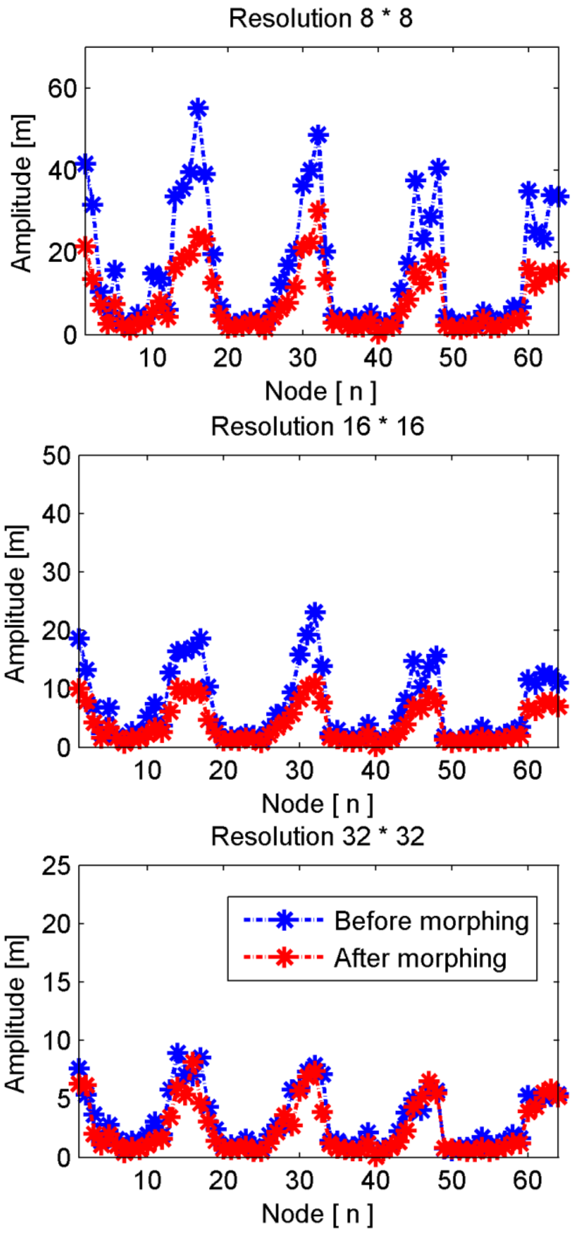

Figure 7. Comparison between the mean square error before and after applying the algorithm.

One important aspect, when dealing with surface approximations, is to quantify the overall approximation error of the method. In our approach, this error quantification is figured out by the following mean-square measure. Let f(x,y) be the original surface and g(x, y) be an approximation. Mean-square error is defined as:

$$\bar{\Delta}^2 = \frac{1}{k}\sum_{i=1}^{k}\Delta(x_i,y_i)^2 \qquad (7)$$

Where: Δ (xi,yi) = f(xi,yi) - g(xi,yi)

The respective reference values for the surface g(xi,yi) are obtained by bilinear interpolation for all the k samples of the regular grid at full resolution.

Figure 7 depicts the mean square errors at several sub-sampling resolutions. According to the curves in figure 7, the algorithm improves the quality of geometric approximation of the entire area and not just parts of the maximum error. Here again we notice that the quality enhancement is not regular and depends on the processed tile.

Considering the results depicted in figure 6 and figure 7, although the algorithm provides an enhancement, the improvement in terms of approximation quality depends on the tile. Its efficiency relies on the initial distribution of LODs inside the tile. The algorithm is more useful when there is an important unbalance of LODs distribution inside the same tile.

8. CONCLUSIONS AND FUTURE WORKS

In this paper, we have presented a new region-based multi-resolution approach for terrain rendering which improves the density of triangles inside the tile after selecting appropriate Level-Of-Detail by an adaptive sampling. Since typical terrain field consists of regions exhibiting high-resolution geometric details and others showing rather coarse geometric structures, this approach employs a geometry morphing and generates for each tile an error-controlled adaptive tessellation of the terrain.
This technique combines the benefits of both Triangular Irregular Network approach and region-based multi-resolution approach by improving the distribution of the density of triangles even inside the tile. Our technique morphs the initial regular grid of the tile to deformed grid in order to minimize approximation error. Thus, this approach uses the wavelet transform as mathematical framework to localize fine grain details region where the density of triangles should be increased to improve the terrain geometry approximation quality. We investigated the efficiency of the proposed method in terms of enhancements of the terrain geometry quality. Geometric approximation error is examined for several resolutions of grids (8×8, 16×16, 32×32), and results before and after applying the proposed morphing technique are compared. The results show that the morphing algorithm is effective to improve the quality of the geometry approximation. However, its efficiency depends on the initial distribution of LODs inside the tile. The algorithm is more effective when there is an important unbalance of LODs distribution inside the same tile. The technique proposed in this paper is simple and tuned for real-time rendering. Such technique could be exploited in client-server architecture for supporting interactive high-quality remote visualization of very large terrain.

In a future work, we will investigate new parameter which determines whether the algorithm processing cost worth it or not. This parameter should estimate the unbalance of LODs distribution inside the same tile. Moreover, future issue of our research is providing an overall mathematical framework based on B-Spline wavelet transform improving the representation of the 3D height-field geometry in terms of memory cost, time performance and rendering quality.

REFERENCES

Asirvatham, A., Hoppe, H., 2005. Terrain Rendering Using GPU-Based Geometry Clipmaps. In: GPU Gems 2. Addison-Wesley, 27–46.

Bettio, F., Gobbetti, E., Marton, F., Pintore, G., 2007. High-quality networked terrain rendering from compressed bitstreams. In: Proceedings of the 12-th int. conf. on 3D web technology, 37–44.

Bösch, J., Goswami, P., Pajarola, R., 2009: Raster: Simple and efficient terrain rendering on the GPU. In EG 2009 - Areas Papers, Ebert D., Krueger J., (Eds.), Eurographics Association, 35-42.

Charles K.C., 1992. An Introduction to Wavelets. In: Academic Press. 249.

Charles, K.C. and Jian-Zhong W., 1992a. On Compactly Supported Spline Wavelets and a Duality Principle. In: Transactions of the American Mathematical Society, 330(2) 903–915.

Cignoni, P., Ganovelli, F., Gobbetti, E., Marton, F., Ponchio, F., Scopigno, R., 2003a. BDAM – batched dynamic adaptive meshes for high performance terrain visualization. In: Computer Graphics Forum, 22(3), 505–514.

Cignoni, P., Ganovelli, F., Gobbetti, E., Marton, F., Ponchio, F., Scopigno, R., 2003b. Planet-sized batched dynamic adaptive meshes (P-BDAM). In: Proceedings IEEE Visualization, 147–155.

Dick, C., Schneider, J., and Westermann, R., 2009. Efficient Geometry Compression for GPUbased Decoding in Realtime Terrain Rendering. In: Computer Graphics Forum. 28(1), 67–83.

Đorđe, M., Đurđević, D., Igor, I., Tartalja, I., 2013. HFPaC: GPU friendly height field parallel compression. GeoInformatica, 17(1), 207-233.

Duchaineau, M., Wolinsky, M., Sigeti, D. E., Miller, M. C., Aldrich, C., Mineev-Weinstein, M. B., 1997. ROAMing terrain: Real-time optimally adapting meshes. In: Proceedings IEEE Visualization, 81–88.

Đurđević, Đ., Tartalja, I., 2011. Domino tiling: a new method of real-time conforming mesh construction for rendering changeable height fields. J Comput Sci Technol 26(6), 971–987.

Fowler, R. J., Little J. J., 1979. Automatic extraction of irregular network digital terrain models. In: Proceedings ACM SIGGRAPH, 199–207.

Gobbetti, E., Marton, F., Cignoni, P., Di Benedetto, M., and Ganovelli, F., 2006. C-BDAM – compressed batched dynamic adaptive meshes for terrain rendering. In: Computer Graphics Forum, 25(3), 333–342.

Gross, M.H., Staadt, O.G., Gatti, R, 1996. Efficient Triangular Surface Approximations using Wavelet and Quadtree Data Structures. IEEE Transaction on Visualization and Computer Graphics, 2(2), 1-13.

Hwa, L.M., Duchaineau, M.A., JOY, K.I., 2005. Real-time optimal adaptation for planetary geometry and texture: 4-8 tile hierarchies. IEEE Transactions on Visualization and Computer Graphics 11(4), 355–368.

Junfeng, J., Enhua, W., Sheng, L., Xuehui, L., 2006. View-dependent refinement of multiresolution meshes using programmable graphics hardware. Vis.Comput, 22(6), 424-433.

Levny, Y., Kogan, Z., El-Sana, J., 2009. Seamless patches for GPU-based terrain rendering. Vis.Comput, 25(3), 197-208.

Liang, H., Sander, P.V.; Hoppe, H., 2010. Parallel View-Dependent Level-of-Detail Control. Visualization and Computer Graphics, IEEE Transactions, 16(5), 718–728.

Lindstrom, P., Cohen, J.D., 2010: On-the-fly decompression and rendering of multiresolution terrain. In: Proceedings ACM SIGGRAPH, 65–73.

Lindstrom, P., Koller, D., Ribarsky, W., Hodges, L. F., Faust, N., Turner, G. A., 1996. Real-time, continuous level of detail rendering of height fields. In: Proceedings ACM SIGGRAPH, 109–118.

Lindstrom, P., Pascucci, V., 2002. Terrain simplification simplified: a general framework for view-dependent out-of-core visualization. Visualization and Computer Graphics, IEEE Transactions, 8(3), 239 – 254.

Losasso, F., Hoppe, H., 2004. Geometry clipmaps: terrain rendering using nested regular grids. In: SIGGRAPH, 23(3), ACM Press, 769–776.

Pajarola, R., 1998. Large scale terrain visualization using the restricted quadtree triangulation. In: Proceedings IEEE Visualization, 19–26.

Pajarola, R., and Gobbetti, E., 2007. Survey on semi-regular multiresolution models for interactive terrain rendering. Vis.Comput. 23(8), 583–605.

Schneider, J., and Westermann, R., 2006. GPU Friendly High-Quality Terrain Rendering. Journal of WSCG, 14, 49–56.

Ulrich, T., 2000. Rendering massive terrains using chunked level of detail control. In: ACM SIGGRAPH Course “Super-size it! Scaling up to Massive Virtual Worlds”.

Yacine, A., Xavier, M., 2009. A GPU Tile-Load-Map architecture for terrain rendering: theory and applications. Vis.Comput, 25(8), 805-824.

This contribution has been peer-reviewed.
doi:10.5194/isprsarchives-XLI-B3-755-2016