Wavelet-Based Hierarchical Surface Approximation from Height Fields

Sang-Mook Lee and A. Lynn Abbott
Electrical and Computer Engineering
Virginia Tech, Blacksburg, VA 24061, USA

Daniel L. Schmoldt
USDA CSREES
Washington, DC 20024, USA

Abstract

This paper presents a novel hierarchical approach to triangular mesh generation from height fields. A wavelet-based multiresolution analysis technique is used to estimate local shape information at different levels of resolution. Using predefined templates at the coarsest level, the method constructs an initial triangulation in which underlying object shapes are well preserved. Wavelet detail coefficients directly control the selection of appropriate templates, and are then used for subdividing and refining the initial mesh.

1. Introduction

The triangular mesh is a popular means of representing three-dimensional surfaces, and multiresolution analysis is often used to allow surface approximations at varying levels of spatial resolution [1,2]. Multiresolution approaches, particularly those moving from coarse to fine resolution, can often improve the computational efficiency of mesh generation.

This paper concerns the use of wavelet-based multi-resolution methods to construct a triangular-mesh surface approximation from a single range image. The novel approach presented here directly evaluates wavelet coefficients to assess surface shape characteristics bounded in each triangular patch at a given scale, and then subdivides and refines triangles based on the evaluation. As the system moves to finer resolution levels, subdivision and refinement steps are repeated to construct a final triangular mesh with desired approximation accuracy.

Related work includes multiresolution surface recovery [5,6] and representation [7], in which spline wavelets are typically utilized to approximate the surface directly with spline functions. A wavelet-based surface approximation for irregularly sampled points has also been studied [8,18]. While most earlier studies have focused on parametric surface representation, however, a few researchers have considered triangular mesh generation directly from wavelet coefficients. By decomposing range data with wavelets, Gross et al. [3] have directly used wavelet coefficients to assist in pruning vertices from a quadtree structure before triangulation. Yu and Ra [4] have extended the work to include edge information.

For manipulating the constructed triangular surface mesh, many computer graphics and computer-aided geometric design applications have adopted multi-resolution techniques, especially with a wavelet-based framework. As means of multiresolution description of meshes, wavelets have very attractive features from a computational point of view [15], and many researchers have considered decomposition and reconstruction of triangular meshes [10,11,12]. In one case, a specially designed wavelet is used to represent functions on the sphere [16,20]. Multiresolution analysis has also been considered for irregular surfaces meshes and for arbitrary shapes [17,19].

Fast triangulation methods for terrain data have also been developed. Garland and Heckbert [22] developed optimized algorithms for Delaunay and data-dependent triangulation criteria, and other researchers focused on real-time meshing and level-of-detail rendering [23,24].

This paper presents a new approach for constructing triangular meshes for surface approximation relatively quickly. The emphasis is on selecting a small number of triangles subject to a specified error tolerance. The next section of this paper presents an overview of wavelet-based multiresolution approaches to hierarchical mesh generation. In section 3, a set of triangulation templates is constructed, and section 4 describes how wavelet coefficients improve the efficiency of coarse-to-fine mesh refinement. Experimental results and concluding remarks are presented in section 5 and section 6, respectively.

2. Multiresolution mesh generation

A common approach in multiresolution analysis is to convert a given dense dataset to a hierarchy of representations, each associated with a different scale. Let m=0 represent the resolution level of original data.

2.1 Background: wavelet decomposition

A common way to decompose two-dimensional signals is to use tensor product wavelets. Let $\phi_x=\phi(x)$ be a scaling function and $\psi_x=\psi(x)$ be a corresponding wavelet basis of

0-7695-2158-4/04 $20.00 (C) 2004 IEEE

$\mathbf{L}^2(\mathbb{R})$, a set of finite energy functions. Then, tensor product wavelets given by

$\psi^1(x, y) = \psi_x\phi_y$, $\psi^2(x, y) = \phi_x\psi_y$, and $\psi^3(x, y) = \psi_x\psi_y$, (1)

and the wavelet family

$$\left\{\psi^1_{m,i,j}(x,y), \psi^2_{m,i,j}(x,y), \psi^3_{m,i,j}(x,y)\right\}_{(i,j)\in\mathbb{Z}^2} \qquad (2)$$

constructs a two-dimensional orthonormal basis of detail space at resolution level m [9]. Each $\psi^k_{m,i,j}$ is derived by scaling and shifting the corresponding function ψ^k:

$$\psi^k_{m,i,j}(x, y) = 2^{-m}\psi^k(2^{-m}x-i, 2^{-m}y-j) \text{ for } 1\le k\le 3.$$

With 2D scaling function $\phi_{m,i,j}(x,y)=2^{-m}\phi(2^{-m}x-i)\phi(2^{-m}y-j)$, therefore, any $f(x,y) \in \mathbf{L}^2(\mathbb{R}^2)$ can be represented by

$$f(x,y)=\sum_{(i,j)} a_{M,i,j}\phi_{M,i,j} + \sum_{m=1}^{M}\sum_{(i,j)}\sum_{k=1}^{3} d^k_{m,i,j}\psi^k_{m,i,j}\,, \qquad (3)$$

where $a_{M,i,j} = \langle f,\phi_{M,i,j}\rangle$ and $d^k_{m,i,j}=\langle f, \psi^k_{m,i,j}\rangle$. The symbol $\langle\cdot,\cdot\rangle$ denotes an inner product, and M is the highest decomposition level. The equation shows that the original function consists of the coarsest approximation and detail information at resolution levels $1\le m\le M$. Hence, a function f can be approximated by

$$\hat{f}(x,y)=\sum_{(i,j)} a_{M,i,j}\phi_{M,i,j} + \sum_{m=1}^{M}\sum_{(i,j)}\sum_{k=1}^{3}\chi_m(d^k_{m,i,j})\psi^k_{m,i,j} \qquad (4)$$

where

$$\chi_m(d^k_{m,i,j})=\begin{cases} d^k_{m,i,j} & \left|d^k_{m,i,j}\right|>\tau_m \\ 0 & otherwise \end{cases}$$

and τ_m is a threshold for detail coefficients at level m.

Since the detail coefficients reflect directional variation of data in a given region, a large magnitude in $d^1_{m,i,j}$, $d^2_{m,i,j}$, and $d^3_{m,i,j}$ tends to be caused by data discontinuities in horizontal, vertical, and diagonal directions, respectively.

2.2 Multilevel hierarchical mesh generation

Consider a two-dimensional data set taken at a grid of sample points V_0 in $\mathbb{R}^2$. A hierarchical mesh is obtained by constructing an initial mesh $\hat{\Gamma}_M$ at the coarsest level, M, with a set of sample points $\hat{V}_M$, and by refining the initial mesh to finer meshes $\hat{\Gamma}_{M-1}$, $\hat{\Gamma}_{M-2}$, …, $\hat{\Gamma}_0$, with sets $\hat{V}_{M-1}$, $\hat{V}_{M-2},\cdots,\hat{V}_0$, respectively. The original dataset is approximated over the entire domain by a piecewise planar function f_m that interpolates all data values at points of $\hat{V}_m$, $0\le m\le M$. Typically, the number of points in $\hat{V}_m$ increases as m decreases, and generally $\hat{V}_0 \ne V_0$. This implies that $|\hat{\Gamma}_m|<|\hat{\Gamma}_{m-1}|$ for $1<m\le M$, where $|\cdot|$ represents mesh size, and $|\hat{\Gamma}_0|\ne|\Gamma_0|$. Consequently, the error $\varepsilon_m=\|f-f_m\|_\infty$, defined in the $\mathbf{L}^\infty(\mathbb{R}^2)$ norm, which means maximum error, decreases as m does.

In most applications, the main goal of mesh generation is to minimize $|\hat{\Gamma}_0|$ subject to ε_0 being kept below a given error criterion. However, the minimization of mesh size for a given accuracy is an NP-hard problem and heuristics are needed for practical implementation.

2.3 Algorithm description

The concept of the new method for generating triangular meshes is illustrated in Figure 1. Wavelet decomposition is first used to construct a multiscale representation of input data. Next, assuming that a rectangular grid at the coarsest level represents the coarsest mesh $\hat{\Gamma}_M$ (Figure 1a), the algorithm examines detail coefficients $d^k_{M,i,j}$, $(i,j)\in\mathbb{Z}^2$ for each rectangular grid in order to estimate underlying shape information. Then, proper triangulation templates are selected based on these coefficients to tessellate the rectangular mesh. This produces a finer mesh $\tilde{\Gamma}_{M-1}$ which needs to be refined afterward in trying to obtain a more accurate and compact representation, that is, $\hat{\Gamma}_{M-1}$. Finally, subdivision and refinement procedures are iterated as the decomposition level decreases until the mesh satisfies a given error criterion or the finest level is reached.

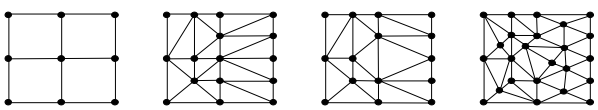
The novelty of the work includes the template construction and the direct use of wavelet coefficients on the subdivision and refinement processes. The use of templates boosts the triangulation speed and reduces the number of triangles by setting up a data-dependent initial mesh at the very beginning. In addition, wavelet coefficients guide the subdivision and refinement for faster processing and more compact and accurate representation. Refinement techniques used here are known as edge swapping and vertex removal, and have often been used for polygonal surface simplification [10,11,12].

3. Initial triangulation

A good initial representation could improve efficiency of all subsequent procedures. The proposed method uses 47 predefined templates, which are defined and examined in this section, to obtain initial triangular meshes.

3.1 Template construction

Templates constructed here are designed to preserve the shape information reflected in detail coefficients. By the dyadic property of wavelet decomposition, wavelet detail coefficient $d^k_{M,i,j}$ represents shape information contained primarily in each square patch with opposite corners given by $(i2^M, j2^M)$ and $((i+1)2^M, (j+1)2^M)$. The set of corner points is $\hat{V}_M = \{(i2^M, j2^M) \mid 0\le i\le C/2^M, 0\le j\le R/2^M\}$, where (C, R) is the size of dataset. If $|d^1_{M,i,j}|=\max_{1\le k\le 3}(|d^k_{M,i,j}|)$ and $|d^1_{M,i,j}|>\tau_M$, for example, then there is a strong evidence of a depth discontinuity along the horizontal direction within the corresponding square patch, and the $d^1_{M,i,j}$ is considered as a dominant coefficient in the patch. A simple example is shown in Figure 2 to illustrate how the templates correspond to underlying shapes. Proper templates are chosen based on investigating wavelet detail coefficients. Note that the templates tend to place triangle edges along depth discontinuities.



(a) $\hat{\Gamma}_2$ (b) $\tilde{\Gamma}_1$ (c) $\hat{\Gamma}_1$ (d) $\hat{\Gamma}_0$

Figure 1. Illustration of mesh generation using M=2.

Figure 2a shows dominant coefficients inside each square. The square on the left top corner does not have a dominant coefficient because the region is relatively flat.

3.1.1. Basic templates. There are four basic templates to which each dominant coefficient corresponds. Figure 3 shows the templates and their construction, denoting as π_f, π_v, π_h and π_d for flat, vertical, horizontal, and diagonal templates, respectively. The template π_f is for relatively flat (planar) portions of data, and there is no additional vertex. The other three templates introduce vertical, horizontal, and diagonal edges into square regions, respectively, with additional vertices. The edges are intended to obtain better approximation of discontinuities contained in the dataset. All other templates are constructed by varying these basic templates.

3.1.2. Variants. One of the four basic templates in Figure 3 could be chosen independently for each square of $\tilde{\Gamma}_M$. However, this would not result in a true triangulation, as illustrated in Figure 4a where two templates, π_h and π_v, are placed side by side. The vertex v from template π_h is not matched with any other vertices of π_v. In order to avoid this kind of discrepancy, an additional set of templates is inevitable so that vertex compatibility can be retained. Figures 4b and 4c show two possible variations of templates π_v, and the one in Figure 4b has better mesh regularity and is chosen as a *variant* for this case. In total, 3 variants for each π_v and π_h, and 15 variants for each π_f and π_d are added to the basic templates.

3.1.3 Duals. A better fit can be achieved, with only a modest additional amount of computational cost, if more templates are considered. The additional templates, called *duals*, could reduce a substantial amount of error for some cases. Figure 5 illustrates the usage of a dual vertical template. It shows that the dual template $\hat{\pi}_v$ obviously fits the given region better than its original template π_v. Proper use of dual templates speeds up the triangulation process by avoiding unnecessary operations, and results in a better approximation with fewer triangles.

Introducing duals increases complexity in selecting a template for a given situation. Directional information embedded in detail coefficients, however, simplifies the selection process. When spline wavelets are used, it is appropriate to interpret d_m^1 and d_m^2 as approximations of two partial derivatives of input data at scale m [9]. That is, the approximation is given by

$$d_m^1 \approx \partial s^{m-1}/\partial x,\ d_m^2 \approx \partial s^{m-1}/\partial y, \qquad (5)$$

where s^{m-1} is average information at level $m-1$. Thus, the orientation of the discontinuity can be estimated by

$$\theta^m(x,y) \approx \tan^{-1}\left(d_m^2/d_m^1\right). \qquad (6)$$

All templates are collected in Figure 6. Variants of π_f and π_d are not displayed here because of space limitations. Template π_f has only one dual, and no duals exist for π_d and variants of π_f due to symmetry.

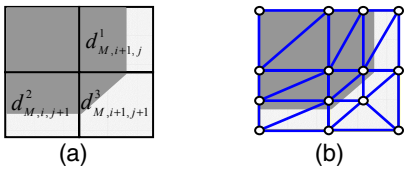


Figure 2. Illustration of (a) dominant coefficients and (b) corresponding templates.

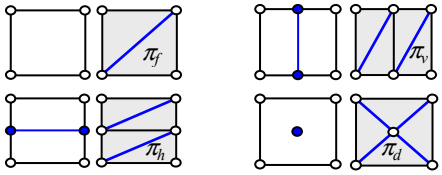


Figure 3. Construction of basic templates.

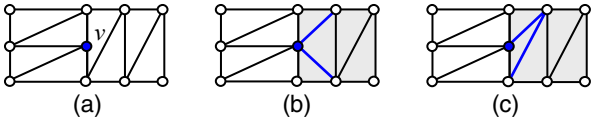


Figure 4. Variants. (a) Basic templates lying side-by-side may be incompatible in vertex location and need a change to the templates as shown in (b) and (c).

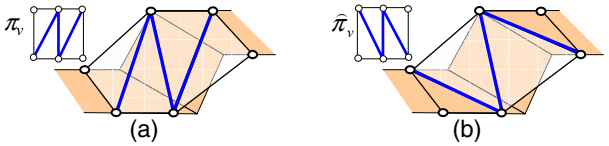


Figure 5. Data fitting with (a) π_v and (b) its dual $\hat{\pi}_v$.

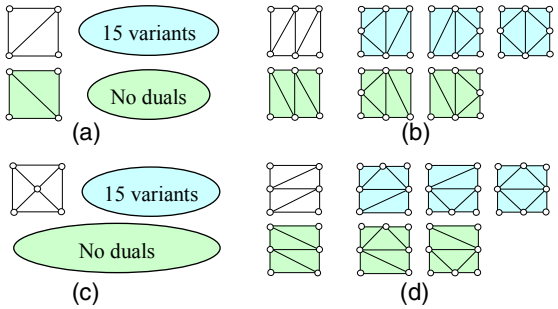
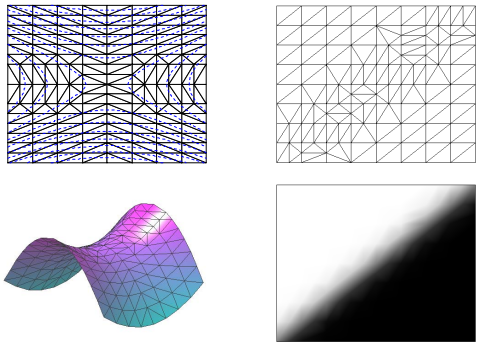


Figure 6. Complete set of templates. Variants are on the right side of each basic template, and duals are located underneath.

3.2. Examples of initial triangulation

Two synthetic datasets are approximated to evaluate the feasibility of the templates defined in the previous section. The first dataset is a hyperbolic quadratic surface sampled at 128×128 grid points, ranging from -20 to 52.2. The second dataset represents a hill shape having gradually varying values from 0 to 128 in a diagonal direction. The datasets are decomposed up to level M=4 with a spline wavelet, yielding 8×8 rectangular grids. Initial triangular meshes are shown on top of Figure 7. The mesh size, $|\tilde{\Gamma}_3|$, is 272 for the hyperbolic surface and 158 for the hill. At the bottom are approximations depicted with errors ε_3 = 1.177 and 0.241, respectively. The error is measured as



(a) Quadratic surface (b) Diagonal hill

Figure 7. Initial triangulation examples.

normal distance in the $\mathbf{L}_\infty$ norm.

The dotted curves in the triangulation of Figure 7a indicate level contours of the quadratic surface. Note that the triangle edges tend to follow the contour curves; that is, our initial triangulation method places triangle edges along contour curves of given surfaces. The same tendency is shown for the second dataset.

4. Local subdivision and refinement

The goal of a coarse-to-fine mesh construction scheme is to build progressively toward a more accurate approximation by increasing the number of triangles. For economy of representation, however, new triangles should be introduced only where warranted by the underlying data. In contrast to most refinement methods, our approach uses wavelet detail coefficients directly for finding triangles to be subdivided. These triangles are subdivided through a process called *edge split*. The resulting mesh is then refined through *edge swap* and *vertex removal* steps.

4.1 Candidate regions, valid pairs and edge split

For the sake of computational efficiency, candidate regions are identified before subdivisions take place. First, triangles whose interior detail energy, defined as

$$e_{m,i,j} = |d^1_{m,i,j}|^2 + |d^2_{m,i,j}|^2 + |d^3_{m,i,j}|^2, \qquad (7)$$

is greater than a threshold δ_d are identified as "active" regions, and their neighbors are called "passive" regions (Figure 8). Triangle T is a candidate when detail coefficients at (i, j)=*floor*(b_x, b_y) have a large amount of energy, where (b_x, b_y) is the center of mass of the triangle. Triangles in active regions must be subdivided since they may produce high approximation error, and triangles in passive regions take part in subdivision process only if necessary. More discussion on regions is found in [21].

The algorithm next searches for a *valid pair*, which is a pair of two triangles that share an edge and have consistent data discontinuities through the common edge. Wavelet detail coefficients are used to determine valid pairs that could lead to best subdivisions. Figure 9 illustrates the concept. Triangles T_1 and T_2 in Figure 9a are adjacent with

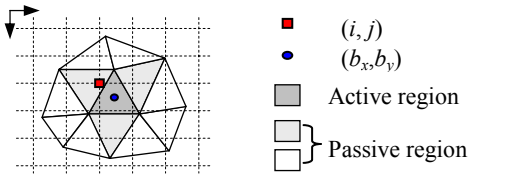


Figure 8. An active region and its neighbors.

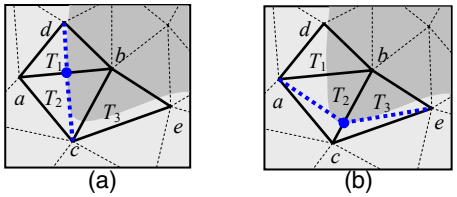


Figure 9. (a) Valid pair. (b) Invalid pair.

an edge E_{ab} and they are associated with vertical discontinuities. Dominant coefficients for both triangles are highly likely to be $|d^1_{m,i,j}| = \max_{1 \le k \le 3}(|d^k_{m,i,j}|)$. Accordingly, the pair T_1–T_2 is said to be *valid*. On the other hand, the triangles T_2 and T_3 in Figure 9b contain vertical and horizontal discontinuities, respectively. The discontinuity is not "consistent" across E_{bc}, and T_2 and T_3 are not considered as a valid pair. Consequently, splitting E_{ab} creates a better approximation than splitting E_{bc}. The split position can be chosen at the midpoint of E_{ab}, but a more sophisticated method is possible, in which edge direction and estimated discontinuity orientation are utilized [21]. When a triangle is involved in more than one pair, the algorithm chooses the pair that has the longest common edge.

4.2 Mesh refinement

4.2.1. Edge swap. The edge-split operation alone often produces very narrow triangles called slivers. Sometimes these slivers are inevitable and represent good fits to the input data. In most cases, however, eliminating slivers does not reduce the quality of approximation appreciably. A well-known approach to eliminating slivers is the *edge swap* operation. This is implemented by altering diagonals of the quadrilateral formed with two adjacent triangular faces, as illustrated in Figure 10a. When the edge swap produces a degenerate triangle whose vertices are collinear, the triangle is removed at a cost of possible increase in approximation error for the counterpart triangles.

The swap operation is controlled by a mesh regularity factor α in a swap criterion

$$\xi_{(T_i,T_j)\to(t_i,t_j)} = \alpha R_\theta + (1-\alpha) R_\varepsilon, \qquad (8)$$

where R_θ and R_ε respectively represent alteration ratios of minimum angles and approximation errors of two different triangulations, $(T_i$-$T_j)$ and $(t_i$-$t_j)$, obtained by a swap operation. If the value ξ is greater than a unity, the edge must be swapped. The parameter α controls regularity of triangular meshes. When α=1, the resulting triangulation has Delaunay properties [14], while α=0 produces pure

data-dependent triangulation. Thus, for $0<\alpha<1$, the function allows slivers to some degree if the error reduction is significant and the triangles are not excessively thin.

4.2.2. Vertex removal. Another technique employed for refinement is a *vertex removal,* in which the number of triangles is reduced by removing unnecessary vertices. Figure 10b illustrates the vertex removal operation. If a v-neighborhood $\mathcal{N}(v)$, defined as triangles sharing a vertex v, has little variation in surface normal, then the vertex v can be removed without a significant loss of accuracy in approximation. After the removal, the remaining polygon (shaded in gray) is re-triangulated with a constrained Delaunay triangulation method, by which non-convex polygons are triangulated with Delaunay properties.

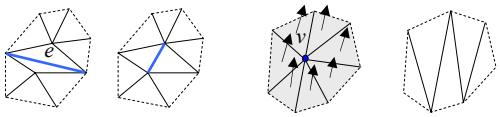
The variation of surface normal can be measured by autocorrelation analysis of the normal vectors $\mathbf{n}_i$ in $\mathcal{N}(v)$. Let k be the number of triangular patches in $\mathcal{N}(v)$. Then, a 3×3 autocorrelation matrix is defined as

$$\mathbf{R}(v) = \frac{1}{k}\sum_{i=1}^{k} \mathbf{n}_i \mathbf{n}_i^{\mathrm{T}} \ . \qquad (9)$$

Typically, the largest eigenvalue of $\mathbf{R}(v)$ corresponds to deviation from the origin, and two small eigenvalues represent variation of the surface normal from the average direction. The criterion for vertex removal, therefore, is to evaluate the two small eigenvalues to see if they are less than a threshold δ_λ. This is another important factor determining mesh quality. A high value of δ_λ will smooth out high frequencies in triangulation. In contrast, a dense and more accurate triangular mesh is generated with low δ_λ.

4.3 Successive split and swap

Successive use of the edge split and swap procedures can produce all subdivision templates commonly used in a triangular mesh subdivision scheme [13]. The procedure is illustrated in Figure 11, where the subdivision templates are shaded in gray. For the sake of drawing this effect, the split and swap operations are iterated three times for each resolution level.



(a) Edge swap (b) Vertex removal

Figure 10. Mesh refinement.

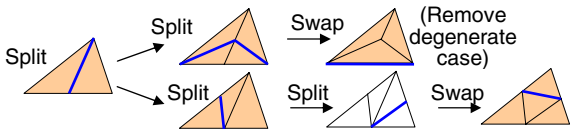


Figure 11. Successive split and swap operations. All subdivision templates (shaded) commonly used in mesh optimization can be produced.

5. Experimental results

Six height fields have been used for evaluating the proposed triangulation method: two datasets each of synthetic data, terrain data, and range images. Figure 12 shows the datasets. Synthetic datasets (`BALL` and `BOX`) vary from 0 to 250 units. The two terrain datasets are digital elevation data for the island of Hawaii (`ISLAND`) and of Crater Lake (`LAKE`), Oregon, downloaded from the USGS. `ISLAND` ranges from 0 (sea level) to 4,245 m, and `LAKE` from 1,730 to 2,477 m. These terrain datasets were clipped and sub-sampled to have the size of 256×256. Range image `BOARD` was acquired using a structured-light method, and its height varies from 0 to 45 in steps of 1/16 inch. The other range image, `PERC`, was downloaded from the University of South Florida and converted so that 39 cm represents the farthest point from the viewer (the background wall) and 276.8 cm represents the nearest point (the foremost floor point). The dataset `PERC` is of size 512×512. The range images contain a considerable amount of noise.

5.1 Multilevel triangulation

The graph in Figure 13 shows that the number of triangles increases as the resolution level m decreases, lowering maximum approximation error. Here, five data sets are decomposed up to level 5 with spline wavelets, and error criteria are set to 1 unit, 2 meters, and 2/16 inches for synthetic, terrain, and range data, respectively. Terrain datasets show a sudden increase in the number of triangles at m=3 and 2. This is due to high wavelet detail energy in mountainous regions. In Figure 14, triangulations for each level and a rendered output for the finest approximation are depicted for the `ISLAND` dataset. As m decreases, triangle subdivision takes place only in ridge and peak regions.

The final triangulations and approximations for the synthetic datasets are shown in Figure 15. For the `BALL` dataset, the triangulation is fairly well distributed in good symmetry, with large triangles around the center and smaller ones toward the boundary. This is because the wavelet detail energy at the boundary is high. The dataset `BOX` has sharp edges and corners. These features make the approximation error stay high until the final triangulation is achieved. The proposed method successfully places large triangles on the faces of the box and triangle edges at box edges. Since the method does not perform exhaustive search for optimal triangulation, the final output still has a few redundant triangles, but it can be a good initial mesh for any further optimization process.

5.2 Level-of-detail control

The level of detail is controlled by varying the threshold δ_d for wavelet detail energy. Results are summarized in Table 1 for the `LAKE` dataset. At δ_d=0.1, the triangulation approximates the data with a maximum

error of 2 meters, and the error increases as does δ_d. The table also shows the number of vertices used in each triangulation. With only 2.92% of the original set of data points, the approximation error is kept below 2m. Figure 16 shows four triangulations according to four different detail energy thresholds.

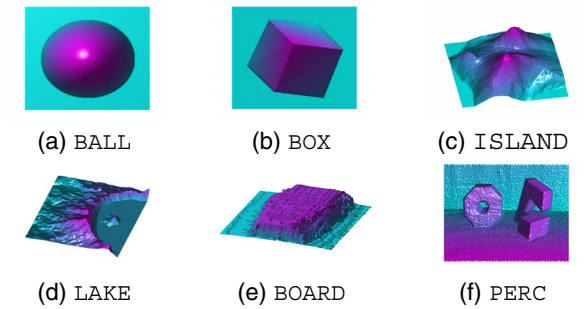


Figure 12. Datasets for experiments.

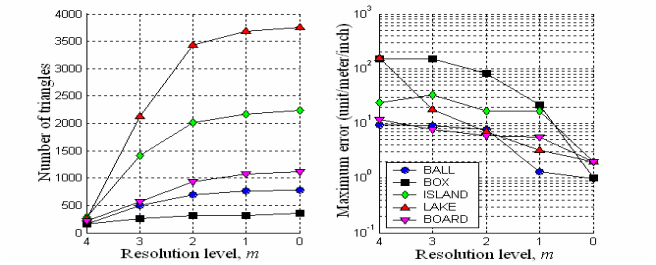


Figure 13. The number of triangles and maximum error according to resolution level m.

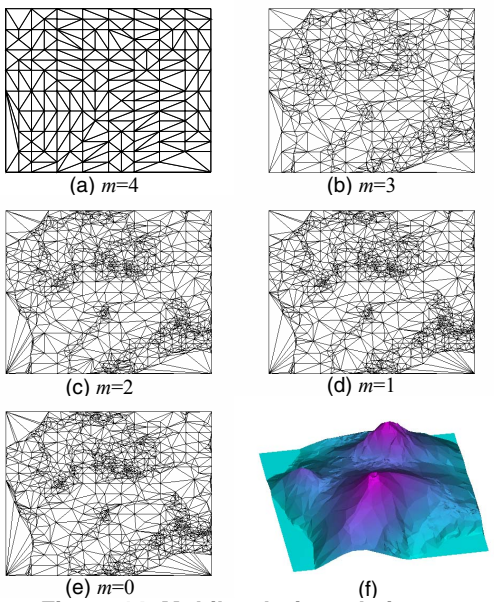


Figure 14. Multilevel triangulation.

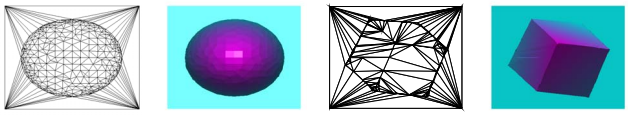


Figure 15. Final meshes and approximations for synthetic datasets.

Table 1. Level-of-detail control with δ_d.

δ_d	0.1	1	2	4	6	8	10
$\lvert\hat{\Gamma}_0\rvert$	3758	3526	2872	1811	1393	1180	1065
$\lvert\hat{V}_0\rvert$	1916	1800	1470	941	735	631	570
%	2.92	2.75	2.25	1.44	1.12	0.96	0.87
ε_0	2	7.0	11.4	16.9	28.0	29.5	30.5

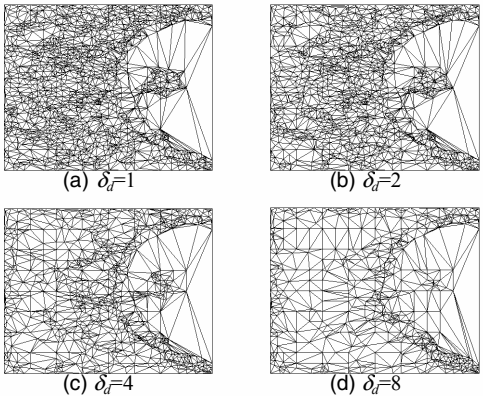


Figure 16. Triangular meshes with different δ_d.

5.3 Mesh regularity factor

To investigate the effect on mesh regularity factor α introduced in equation (8), different α values were tested with the ISLAND dataset. Table 2 shows the number of triangles $|\hat{\Gamma}_0|$ generated for each value of α. Here, the same error tolerance is set for all triangulations. The case α=0 allows triangle slivers, generating a pure data dependent triangulation, while α=1 leads to Delaunay triangulation. The $|\hat{\Gamma}_0|$ value generally increases with α, but it is also possible that low α causes more triangles than high α (=0.2). Our experiments have revealed that α=0.6 to 0.8 gives the best performance in terms of mesh regularity and the number of triangles.

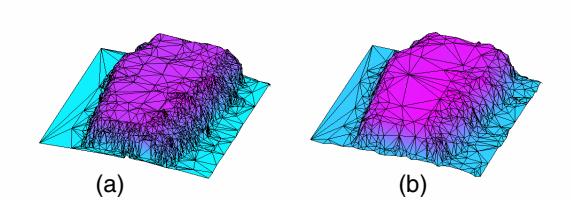
Table 2. Mesh regularity.

α	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
$\lvert\hat{\Gamma}_0\rvert$	1912	2005	2167	2152	2218	2321	2369	2383	2492	2511	2651

5.4 Noisy data and selective triangulation

Triangulations of the noisy dataset BOARD are shown in Figure 17. Noise causes a many tiny triangles, as shown around the boundary of the scanned board (17a), and most of them are not necessary for further processing such as boundary detection. A wavelet de-noising technique [9] reduces the noise level in the data and leads to avoid unnecessary triangles (17b). A similar effect can be achieved with the level-of-detail control capability.

Figure 18 shows an example of selective triangulation that is similar to one presented in [3]. Many computer vision tasks deal with a couple of small objects in a scene. Therefore, extracting out some candidates could expedite the tasks. For these purpose, the wavelet-based approach



(a) (b)

Figure 17. Noisy data. Wavelet de-noising is applied to reduce the noise level in the dataset.

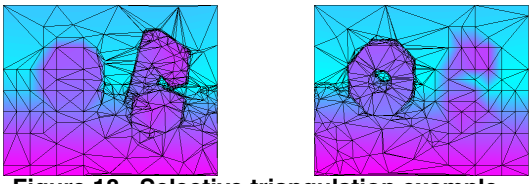


Figure 18. Selective triangulation example.

has advantage: as the resolution level increases, non-target regions are identified and corresponding wavelet coefficients are suppressed by which no further subdivision and refinement take place in those regions, while target objects proceed to a full reconstruction.

6. Conclusions

A novel wavelet-based hierarchical triangular mesh generation method has been presented and tested for approximating range and intensity images. The method utilizes wavelet coefficients in order to extract multiresolution shape information from the underlying data, and this is used to guide the triangulation process. In particular, initial triangulation is very fast through the use of predefined templates. A new edge split operation subdivides the initial mesh into finer approximations. Experimental results show that the proposed method provides a better compromise in terms of speed, error of fit, and mesh size. For example, the method of [22] approximates the dataset `LAKE` with 5,000 vertices producing 3 meters of RMS error in about 15 seconds, but for the same dataset our method uses 4,132 points to obtain a maximum error of less than 1 meter. Our unoptimized code takes 8 seconds for this result on a 1GHz PC. With other mesh operations such as edge swap and vertex removal, the method is capable of multiresolution triangular mesh generation, level-of-detail control, and selective triangulation.

7. References

[1] J.-L. Starck, F. Murtagh and A. Bijaoul, *Image Processing and Data Analysis: The Multiscale Approach*. Cambridge University Press, 1998.

[2] S. Takahashi, Y. Shinagawa, and T. Kunii, "Curve and Surface Design Using Multiresolution Constraints", *Computer Graphics International*, 1997, pp.121-130.

[3] M.H. Gross, O.G. Staadt, and R. Gatti, "Efficient Triangular Surface Approximations Using Wavelets and Quadtree Data Structures", *IEEE Trans. on Visualization and Computer Graphics*, vol.2, no.2, 1996, pp.130-143.

[4] H.J. Yu and J.B. Ra, "Fast Triangular Mesh Approximation of Surface Data Using Wavelet Coefficients", *The Visual Computer*, vol. 15, Springer-Verlag, 1999, pp.9-20.

[5] M. Yaou and W. Chang, "Fast Surface Interpolation Using Multiresolution Wavelet Transform", *IEEE Trans. on PAMI*, vol.16, no.7, 1994, pp.673-688.

[6] M. Maeda, K.Kumamaru, H. Zha, K. Inoue and S. Sawai, "3-D Surface Recovery From Range Images By Using Multi-resolution Wavelet Transform", *IEEE Intl. Conf. on Computational Cybernetics and Simulation*, vol.4, 1997, pp. 3654-3659.

[7] A. Dreger, M.H. Bross, and J. Schlegel, "Multiresolution Triangular B-Spline Surface", *Proc. of Computer Graphics International*, 1998, pp.166-177.

[8] L. Paster and A. Rodriguez, "Surface Approximation of 3D Objects From Irregularly Sampled Clouds of 3D Points Using Spherical Wavelets", *Intl. Conf. on Image Analysis and Processing*, 1999, pp.70-75.

[9] Mallat, S., *Wavelet Tour of Signal Processing*. Academic Press, 1998.

[10] M. Eck, T. DeRose, and T. Duchamp, "Multiresolution Analysis of Arbitrary Meshes", *Proc. of SIGGRAPH 95*, Los Angeles, California, 1995, pp.173-182.

[11] H. Hoppe, T. DeRose, and T. Duchamp, "Mesh Optimization", *Proc. of SIGGRAPH 93*, Anaheim, California, 1993, pp.19-26.

[12] M. Garland and P.S. Heckbert, "Surface Simplification Using Quadric Error Metrics", *Proc. of SIGGRAPH 97*, Los Angeles, California, 1997, pp.209-215.

[13] L. Scarlatos and T. Pavlidis, "Hierarchical Triangulations Using Cartographic Coherence", *CVGIP: Graphical Models and Image Processing*, 54(2), 1992, pp.147-161.

[14] Preparata, F.P. and M.I. Shamos, *Computational Geometry*, Springer-Verlag, New York, 1985.

[15] Stollnitz, E.J., T.D. DeRose, and D.H. Salesin, *Wavelets for Computer Graphics: Theory and Applications*, Morgan Kaufmann, San Francisco, 1996.

[16] P. Schroeder and W. Sweldens, "Spherical Wavelets: Efficiently Representing Functions on the Sphere", *Proc. of SIGGRAPH '95*, 1995, pp.161-172.

[17] M. Lounsbery, T.D. DeRose, and J. Warren, "Multiresolution Analysis for Surfaces of Arbitrary Topological Type", *ACM Trans. on Graphics*, vol.16, no.1, 1997, pp.34–73.

[18] C.R. Madhuram, S.S. Kim, R. Guha, G. Schiavone, and R. Mohapatra, "Multiresolution Representation of Non-Uniformly Sampled Terrain Databases Using Wavelets", *Conf. on Signals, Systems and Computers*, 1997, pp.988-992.

[19] G. Bonneau, "Multiresolution Analysis on Irregular Surface Meshes", *IEEE Trans. on Visualization and Computer Graphics*, vol.4, no.4, 1998, pp.365-378.

[20] M. Bertram, M.A. Duchaineau, B. Hamann, and K.I. Joy, "Bicubic Subdivision-Surface Wavelets for Large-Scale Isosurface Representation and Visualization", *Proc. of IEEE Visualization*, 2000, pp.389 -396, pp.579.

[21] S. Lee, *Wavelet-Based Multiresolution Surface Approximation from Height Fields*, Dissertation, Virginia Tech, 2002.

[22] M. Garland and P. Heckbert, "Fast Polygonal Approximation of Terrains and Height Fields", *Carnegie Mellon University*, Technical report. CMU-CS-95-181, 1995.

[23] M. Duchaineau, M. Wolinsky, D.E. Sigeti, M.C. Miller, C. Aldrich, and M.B. Mineev-Weinstein, "ROAMing Terrain: Real-Time Optimally Adapting Meshes", *Proc. of IEEE Visualization*, 1997, pp.81-88, pp.585.

[24] P. Lindstrom, D. Koller, W. Ribarsky, L.F. Hodges, N. Faust, G.A. Turner, "Real-Time, Continuous Level of Detail Rendering of Height Fields", *Proc. of ACM SIGGRAPH 96*, August 1996, pp.109-118.

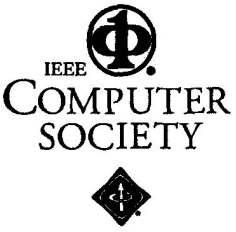
PROCEEDINGS OF THE

2004 IEEE COMPUTER SOCIETY CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION

CVPR 2004

Washington, D.C., USA
27 June-2 July 2004

Sponsored by
IEEE Computer Society



Los Alamitos, California
Washington • Brussels • Tokyo