*Science in China Series E: Technological Sciences*



www.scichina.com
tech.scichina.com
www.springerlink.com

# Multi-resolution representation of digital terrain models with terrain features preservation

LI QingQuan[1], WANG Zhi[1,2†] & YANG BiSheng[1]

[1] State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan 430079, China;
[2] School of Remote Sensing and Information Engineering, Wuhan University, Wuhan 430079, China

**Abstract multi-resolution TIN model is an important issue in the contexts of visualization, virtual reality (VR), and geographic information systems (GIS). This paper proposes a new method for constructing multi-resolution TIN models with multi-scale topographic features preservation. The proposed method is driven by a half-edge collapse operation in a greedy framework and employs a new quadric error metric to efficiently measure geometric errors. We define topographic features in a multi-scale manner using a center-surround operator on Gaussian-weighted mean curvatures. Experimental results demonstrate that the proposed method performs better than previous methods in terms of topographic features preservation, and is able to achieve multi-resolution TIN models with a higher accuracy.**

digital terrain models, level of detail, differential-geometry, quadric error metrics, topographic feature

## 1 Introduction

The representation of digital terrain models at different levels of accuracy and resolution has an impact on applications such as geographic information systems (GISs)[1], virtual reality (VR), progressive transmission of spatial data[2], mobile visualizations, and Web-GIS[3]. Multi-resolution terrain models allow for representation, analysis and manipulation of terrain data at variable resolutions, and provide a promising solution for the progressive transmission of spatial data, spatial data compression, mobile visualizations, and so on. However, the existing methods and algorithms mainly focus on the accuracy and running times of generating the levels-of-details (LoDs) of terrains. Less attention has been paid to features preservation of terrains, particularly at a low resolution model. Suppose that the original terrain features are lost at a low resolution terrain

Received December 12, 2007; accepted January 14, 2008
doi: 10.1007/s11431-008-5015-4
†Corresponding author (email: qqli@whu.edu.cn)
Supported by the National Basic Research Program of China ("973") (Grant No. 2006CB705500), the National Natural Science Foundation of China (Grant No. 40571134), the National Hi-Tech Research and Development Program of China (Grant Nos. 2007AA12Z241, 2007AA12Z212) and Outstanding Scholar of Ministry of Education of China (Grant No. NCET-07-0643)

model. Poor visualization effects and spatial analysis results will be generated.

This paper aims to propose an algorithm for generating multi-resolution terrain models with a good performance in terms of rapid running time, good preservation of terrain features. Two key components are encompassed in the proposed algorithm, namely, detecting and adaptively ranking terrain features based on the Gaussian-weighted of surface curvatures, measuring deviations between the original terrain model and its approximations based on a new error metric. The former aims to preserve terrain features during the generation of multi-resolution terrain models hence improve the accuracy of multi-resolution terrain models in terms of the RMSEs and Hausdorff distances (Hausdorff distance is defined as the maximal Euclidean distance between any point of original model and the closest point of its approximation.); the latter aims to achieve rapid running time.

## 2 Previous work

There has been extensive research on generating multi-resolution terrain models. For the surveying of various simplification schemes, readers can refer to Luebke et al.[4]. The local operators (vertex removal, edge collapse, and triangle collapse) are commonly used and easy to implement for generating multi-resolution models. Moreover, edge collapse and triangle collapse operators can be conceptualized as gradually shrinking the appropriate geometric primitive (edge and triangle, respectively) to a single vertex. They are well suited for implementing geomorphing between successive LoDs. The advantage of the iterative edge collapse operator is its hierarchical structure which is essential to retaining the topological relationship of the terrain model.

The major difference among these edge collapse algorithms is in the selection of the candidate edges and determination of new vertices. Generally, two kinds of methods, namely, quadric error metrics and the theory of local differential-geometry, are extensively used to determine the positions of new vertices. Garland and Heckbert[5] used quadric error metrics and, based on this, developed QSlim software. Yang et al.[6] propose a new method which extends the full-edge collapse and vertex split algorithms, to dynamically generate multi-resolution TIN models. However, these algorithms and methods have difficulties in well preserving terrain features at a low resolution terrain model. Lee et al.[7] implemented a mesh saliency model for capturing visually interesting regions on a mesh. This method modified the quadrics-based simplification method (QSlim) of Garland[8] by weighting the quadrics with mesh saliency. Their work provides a good start in merging perceptual criteria with mathematical measures based on discrete differential geometry for terrains. However, their algorithm is quite time consuming.

## 3 The proposed algorithm

The proposed algorithm is driven by half-edge collapse operator for generating multi-resolution terrain models, and aims to improve the preservation of terrain features of multi-resolution terrain models. The proposed algorithm detects and ranks terrain features according to the theory of local differential-geometry, and uses a weighted quadric error metric to determine the positions of new vertices. Thus, terrain features are able to well preserve and the accuracy (e.g. RMSEs of elevations and Hausdorff distances) of multi-resolution terrain models is improved.

### 3.1 Detecting and ranking terrain features

The most widely used set of topographic characteristics[9] is the subdivision of all points on a

surface into one of plane, peak, pit, ridge, channel, and pass (as illustrated in Figure 1), which are defined as terrain features. The proposed algorithm first detects the terrain features according to the curvature of terrain surface.

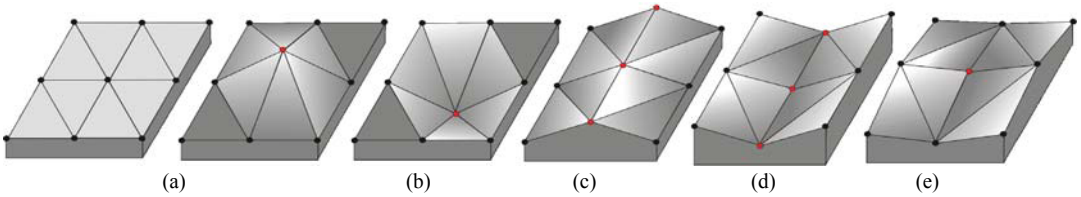


**Figure 1** The category of terrain features. (a) Peak; (b) pit; (c) ridge; (d) channel; (e) pass.

The curvature of terrain surface indicates the property of a terrain surface. The mean curvatures of vertices of a surface indicate whether they are feature vertex or zero-feature vertex. A lot of studies have been done for estimating discrete mean curvature of TIN based surfaces. Unfortunately, no one is widely accepted as the most accurate method or the best method for curvature estimation. Surazhsky and Gotsman[10] showed that the paraboloid fit method is the best one for estimating the mean curvature of meshes. However, the paraboloid fit method is time-consuming and may have extreme difficulties for fitting the osculating quadric in the area of high curvature. Generally, the following formula is widely used to estimate the discrete mean curvature of each vertex $v_i$ of TIN based surfaces because of its efficiency, accuracy, and generality.

$$M(v_i)=\frac{\sum_{j=1}^{n}\| e_{i,j}\|\beta_j}{\frac{4}{3}\sum_{j=1}^{n}\text{area}(f_j)}, \tag{1}$$

where $n$ is the number of adjacent triangles of $v_i$, $\|e_{i,j}\|$ is the length of edge $e_{i,j}$, area($f_j$) is the area of each triangle adjacent to edge $e_{i,j}$, and $\beta_j$ is the dihedral angle between two triangles that incident on edge $e_{i,j}$ (as illuminated in Figure 2).

As shown in Figure 3(a), we compute mean curvature for each vertex. Then, for each vertex, we compute the Gaussian-weighted average of the mean curvatures of vertices within a radius $2\sigma$, where $\sigma$ is Gaussian's standard deviation (as illustrated in Figure 3(b)). The topographic features are determined at different scales by varying $\sigma$. As shown in Figure 3(b), the multi-scale model is used to ignore local perturbations that go against the overall trend of the linear feature. Let the mean curvature map $M$ define a mapping from each vertex of a TIN model to its mean curvature, i.e. let $M(v)$ denote the mean curvature of vertex $v$. Let the neighborhood $N(v, \sigma)$ for a vertex $v$, be the set of points within a distance $\sigma$. In our scheme, we use the Euclidean distance $N(v, \sigma)$ =

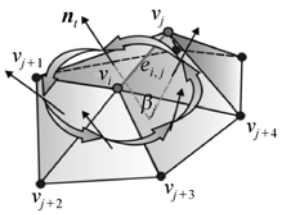


**Figure 2** Calculation of discrete mean curvature.

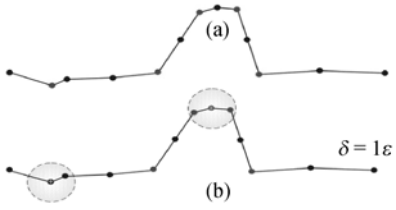


**Figure 3** Center-surround mechanism for estimating of the mean curvature.

$\{x:\|x-y\|<\sigma, x$ is a vertex on the TIN surface$\}$. Let $G(M(v), \sigma)$ denote the Gaussian-weighted average of the mean curvature. We compute this as

$$G(M(v),\sigma)=\frac{\sum_{x\in N(v,2\sigma)} M(x)\exp[-\|x-v\|^2/(2\sigma^2)]}{\sum_{x\in N(v,2\sigma)} \exp[-\|x-v\|^2/(2\sigma^2)]}, \quad (2)$$

where $\sigma$ is the standard deviation of the Gaussian filter. For all the results in this paper we have used seven scales $\sigma\in\{1\varepsilon, 2\varepsilon, 3\varepsilon, 4\varepsilon, 5\varepsilon, 6\varepsilon, 7\varepsilon,\}$, where $\varepsilon$ is defined as 0.3% of the length of the diagonal of the bounding box of the model[7].

Let the topographic feature map $F$ define a mapping from each vertex of a TIN model to its feature, i.e. let $F(v)$ denote the feature of vertex $v$. As shown in Figure 4 (b), the feature map $F$ may have far too many "bumpy" being flagged as features. We apply a non-linear suppression operator $S$ similar to the one proposed by Lee et al.[7]. This suppression operator promotes salience maps with a small number of high values (Figure 4(c)). Thus, non-nonlinear suppression helps us in reducing the number of topographic feature points. For each feature value $F_i$, we compute the maximum salient value $M_i$ and the average $A_i$ of the local maxima excluding the global maximum at that scale. Finally, we multiply $F_i$ by the factor $(M_i-A_i)^2$. $F_i=S(F_i)$ denotes this non-linear normalization of suppression. Figure 4(c)—(g) gives an overview of topographic feature map in different scales. We can see that the topographic features are more coherent in the large-scales.

Figure 5 shows the detection of topographic features such as peak, pit, ridge, channel and pass using our method in different scales. We use pseudo-colors to texture the surface according to the feature weights: Warmer colors (reds and yellows) show high weights, cooler colors (greens) show low weights, and blues show zero-feature. We guide the order of iterative half-edge collapses using a weight map $\omega$ derived from the topographic feature map $F$. We found that using the simplification weights based on a non-linear amplification of the feature map gives us good results. We think that the reason behind this is that by amplifying the salient feature vertices we are ensuring that they are preserved longer than the zero-feature vertices with high collapse costs. In our algorithm, we use values of Gaussian-weighted mean curvature to evaluate the point to the extent of terrain feature. Then the nonlinear suppression will help us distinguishing feature point from zero-feature point. In order to improve the speed of processing, we do not classify different features, such as peak, pit, ridge, channel and pass in our algorithm. However, the feature classification can easily be achieved according to the rules of Wood[9].

### 3.2 The calculation of new quadric error metric

The general edge collapse simplification involves two major steps: Choosing a measure that specifies the cost of collapsing an edge, and choosing the position of vertex that replaces the edge . Kobbelt et al.[11] proved the topological operator to vertex placement does not have a significant impact on the simplification results. The quality of the simplification result is much more sensitive to the criteria which decides where to execute the reduction operation. Therefore, Kobbelt et al.[11] recommend to use the topological operator itself as simple as possible. Following these considerations, we use half-edge collapse operator (as illustrated in Figure 6) to simplify the input TIN models. We employ the half-edge collapse operator to simplify the topology of original TIN because this operator does not contain any unset degrees of freedom. Moreover, this reduction operation does not create new geometry, i.e., the position of new vertex. The vertices of the decimated

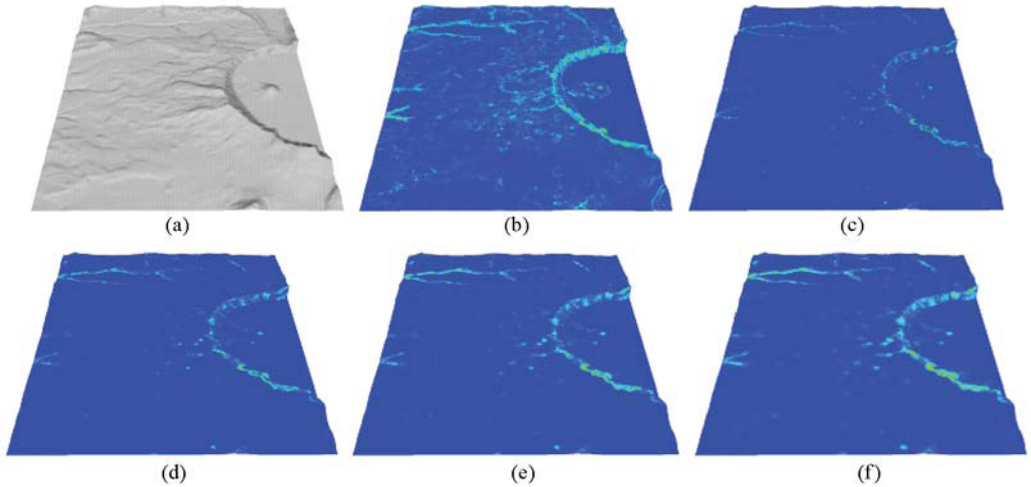


**Figure 4** Topographic feature detection. (a) Shows the "Crater" model; (b) shows its mean curvature distribution; (c)—(f) show the salient features at scales of $1\varepsilon$, $2\varepsilon$, $3\varepsilon$, $4\varepsilon$.

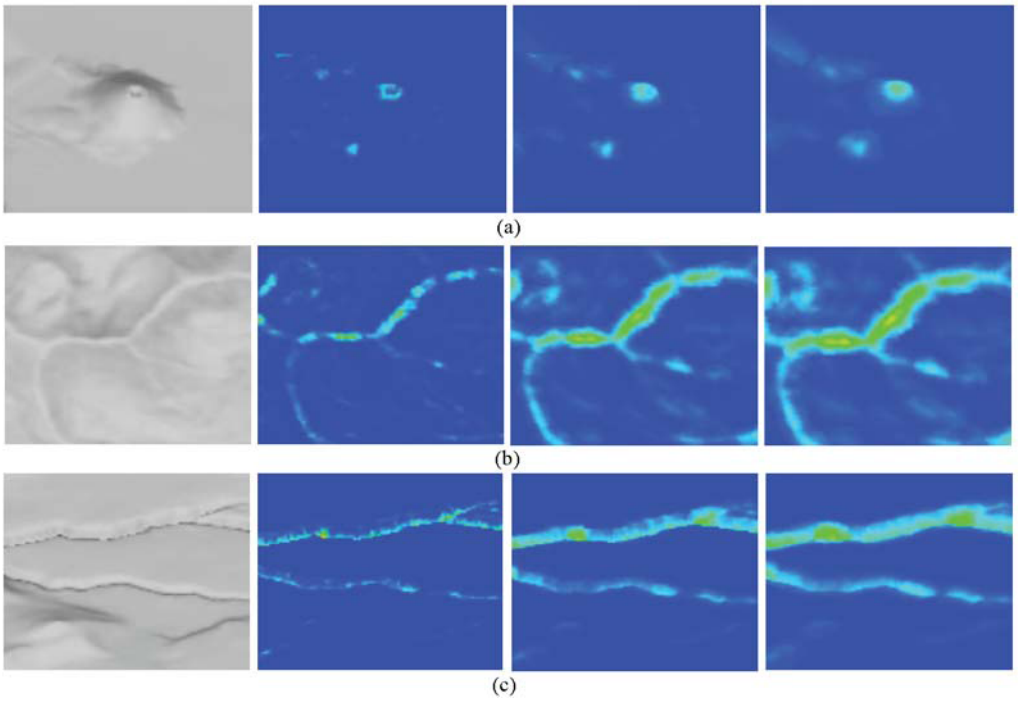


**Figure 5** Multi-scale topographic feature measurement: Images show the topographic features at scales of $1\varepsilon$, $3\varepsilon$, $5\varepsilon$. (a) Peak and pit; (b) ridge and pass; (c) channel.

TIN generated by half-edge collapse are always a proper subset of the original vertices.

Since we are concerned with rapid producing approximations which remain faithful to the original topographic features, the geometric errors should reflect how much that half-edge collapse changes the surface and the error measures also should be cheap to evaluate. Previous QEM-based methods estimate the simplification errors by accumulating the distance of the new vertex from the original surfaces. In order to achieve rapid simplification, we adopted a more efficient measure of these errors. In our approach, the simplification error is measured by the distance of vertex from the

"new" planes which undergo a transformation following a half-edge collapse.

As shown in Figure 6, a half-edge collapse $(v_i, v_j)\rightarrow v_j$ causes the triangular mesh $T_n$ to degenerate and the remaining triangles $T_{n-1}=T_n-T_s$ to undergo a transformation. A straight solution to estimate the current error during the reduction process is to compute the deviation of the submesh $T_{n-1}$ that replaces the mesh $T_n$ in the $n$th step of the simplification. After singular triangles $T_s$ are removed, the transformation of $(v_i, v_j)\rightarrow v_j$ can be interpreted to be pull the vertex $v_i$ into $v_j$ and its adjacent triangles $T_{t(i)}$ are transformed to $T'_{t(i)}$ by replacing $v_i$ with $v_j$, and vice versa. These triangles $T_{t(i)}$ (red triangles in Figure 6) are defined as the triangles which surround the vertex $v_i$ but are not adjacent to $v_j$. In our algorithm, a big difference from the method of Garland and Heckbert[5] and Garland[8] and other methods is that the new quadric error metric is defined over the triangles ($T'_{t(i)}$ or $T'_{t(j)}$) that have been transformed in a half-edge collapse ($(v_i, v_j)\rightarrow v_j$ or $(v_i, v_j)\rightarrow v_i$) rather than all of the neighbor triangles. As shown in Figure 7, we make an error quadric bounding the region by accumulating both the squared distance $E_{Q(v_i)}$ and $E_{Q(v_j)}$. $E_{Q(v_i)}$ is the squared distance of the vertex $v_i$ from the "new" planes $T'_{t(i)}$. $E_{Q(v_i)}$ is the squared distance of the vertex $v_j$ from the "new" planes $T'_{t(j)}$.

"Quadric"[5] provides a very convenient representation for the squared distance. Readers can refer to ref. [8] for the detail of "quadric" representation. In our algorithm, we employ this "quadric" to represent the squared distance for its computationally efficient. For example, as shown in Figure 7, given one of the "new" planes $T'_t(i)$ which is the set of all points $n\times\boldsymbol{v}+d=0$ with a unit

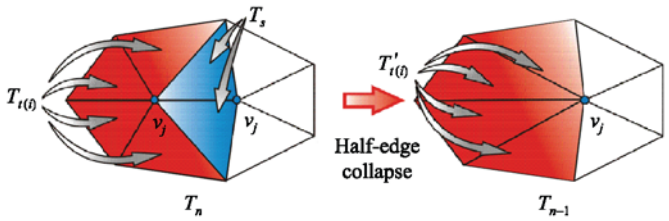


**Figure 6** Half-edge collapse.

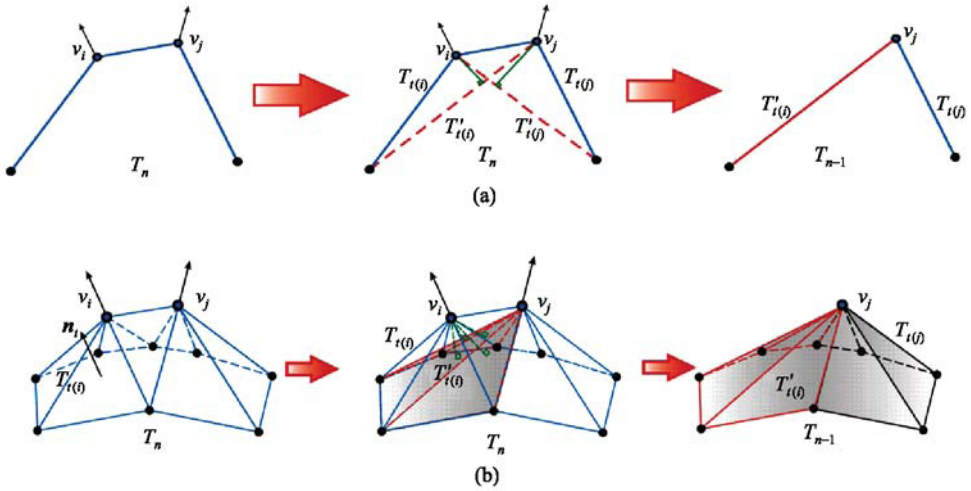


**Figure 7** Measuring the cost of half-edge collapse. (a) Measuring the cost of half-edge collapse and choosing the position of the vertex in 2D; (b) measuring the cost of half-edge collapse and choosing the position of the vertex in 3D.

normal $\boldsymbol{n} = [a\ b\ c]^{\mathrm{T}}$ and a scalar constant $d$, for point $v_i$, the quadric $\boldsymbol{Q}$ is defined

$$\boldsymbol{Q} = (\boldsymbol{A}, \boldsymbol{b}, c) = (\boldsymbol{n}\boldsymbol{n}^{\mathrm{T}}, d\boldsymbol{n}, d^2), \tag{3}$$

where $\boldsymbol{A}$ is a 3×3 matrix, $\boldsymbol{A}=\boldsymbol{n}\boldsymbol{n}^{\mathrm{T}}$, $\boldsymbol{b}$ is a 3-vector, $\boldsymbol{b}=d\boldsymbol{n}$, and $c$ is a scalar. The squared distance $Q(v_i)$ of point $v_i$ to the plane $T_t'(i)$ can be calculated using the second order equation:

$$\boldsymbol{Q}(v_i) = \boldsymbol{v}_i^{\mathrm{T}} \boldsymbol{A}\boldsymbol{v} + 2b^{\mathrm{T}}\boldsymbol{v}_i + c. \tag{4}$$

Therefore, given a set of fundamental quadrics $Q_i(v_i)$ determined by a set of planes $T_t'(i)$, the quadric error $E_{Q(v_i)}$ is computed by the sum of the quadrics $Q_i(v_i)$:

$$E_{Q(v_i)} = \sum_i^n Q_i(v_i) = \left(\sum_i^n Q_i\right)(v_i). \tag{5}$$

In the greedy framework, the collapsed half-edge is selected in an increasing order of cost, which is calculated according to the new quadric error metric. In general, a good simplification algorithm should preserve important regions, such as the regions with salient topographic features. Consequently, the edges in the regions with salient topographic features should have heavier weights. Therefore, these two distances $E_{Q(v_i)}$ and $E_{Q(v_j)}$ should be weighted by the topographic importance of $v_i$ and $v_j$, respectively. Then, we take the minimum as the cost of this half-edge collapse $C_{e_{i,j}}$ and implement half edge collapse $(v_i, v_j)\rightarrow v_j$ or $(v_i, v_j)\rightarrow v_i$ according to the minimum cost. Therefore, the new quadric error metric can be written as

$$C_{e_{i,j}} = \min\{\omega_i E_{Q(v_i)}, \omega_j E_{Q(v_j)}\}, \tag{6}$$

where $\omega_i$ and $\omega_j$ are the weights from the feature map $F$ for topographic salience of vertex $v_i$ and $v_j$, respectively. In the case of Figure 7, $\omega_i E_{Q(v_i)} < \omega_j E_{Q(v_j)}$, so $C_{e_{i,j}} = \omega_i E_{Q(v_i)}$, the half-edge collapse $(v_i, v_j)\rightarrow v_j$ should be implemented. For the edges on the boundary, we use "boundary constraints" invented by Garland to restrain the algorithm from reducing boundary indiscriminately. Reader can refer to Garland[8] for details. Since the new quadric error metric does not involve the decimated triangles ($T_s$) or evaluating the optimal position of new vertex for error measurement and cost determination, it is more computationally efficient than the previous QEM.

## 4 Results and discussion

Among previous TIN model simplifications, the quadric error metric (QEM) based method[5,8] holds much promise in terms of its time efficiency and relatively high quality of approximations. Garland and Zhou[12] extend the QEM-based algorithm to simplify simplicial complexes of any type embedded in Euclidean spaces of any dimension and based on this, developed new GSlim software. However, the performance of their newer GSlim system on triangulated models is essentially identical to that of the earlier QSlim 2.0[8]. Surazhsky and Gotsman[10] examined nine software packages for mesh simplification, including both commercial (Geomagic Studio 5.0, Rapidform 2004, 3ds max 7, Maya 5.0, Action3D Reducer 1.1, SIM Rational Reducer 3.1 and VizUp Professional 1.5) and academic offerings (QSlim 2.0 and Memoryless Simplification). They tested these software packages on the seven models of different sizes, properties and acquisition sources. According to their experimental results, they concluded that the Hausdorff distance reflects visual fidelity better than the average distance. The possible reason is that a large deviation

from the original surface even at just a small localized feature of the mesh can significantly affect the visual perception of the model, and this will be reflected in the Hausdorff distance even if the rest of the simplified mesh is very close to the original. In their experiments, Geomagic Studio was the leader with respect to the Hausdorff distance.

For the above reasons, in the experiments, we use "Crater" model to compare our scheme with QSlim 2.0, which use area-based weights and optimizes vertex locations, and Geomagic Studio 8, which is the latest version of Geomagic Studio, for generating multi-resolution models in terms of visual performance, geometric errors (RMS), Hausdorff distance and time performance. The proposed approach was implemented in C++ language on Windows XP operation system platform. The experiment was undertaken in a 3.0 GHz Intel Pentium IV machine with 512 MB of main memory.

Figures 8 and 9 show the multi-resolution "Crater" model generated by QSlim, Geomagic Studio 8 and our scheme from 199114 triangular faces to 4000, 2000, and 300 triangular faces, respectively. It can be seen that our new algorithm has better performance in terms of the preservation the topographic features, such as peak, pit, ridge, channel, pass and various blending shapes that transition between them. Table 1 shows the error results of QSlim 2.0, Geomagic Studio 8 and our new scheme. The relative errors are measured by Metro 4.06[13] which is based on sampling and point-to-surface distance computation. It can be seen that all the results generated by our new scheme show more accurate than those of QSlim 2.0 and Geomagic Studio 8 in terms of RMS

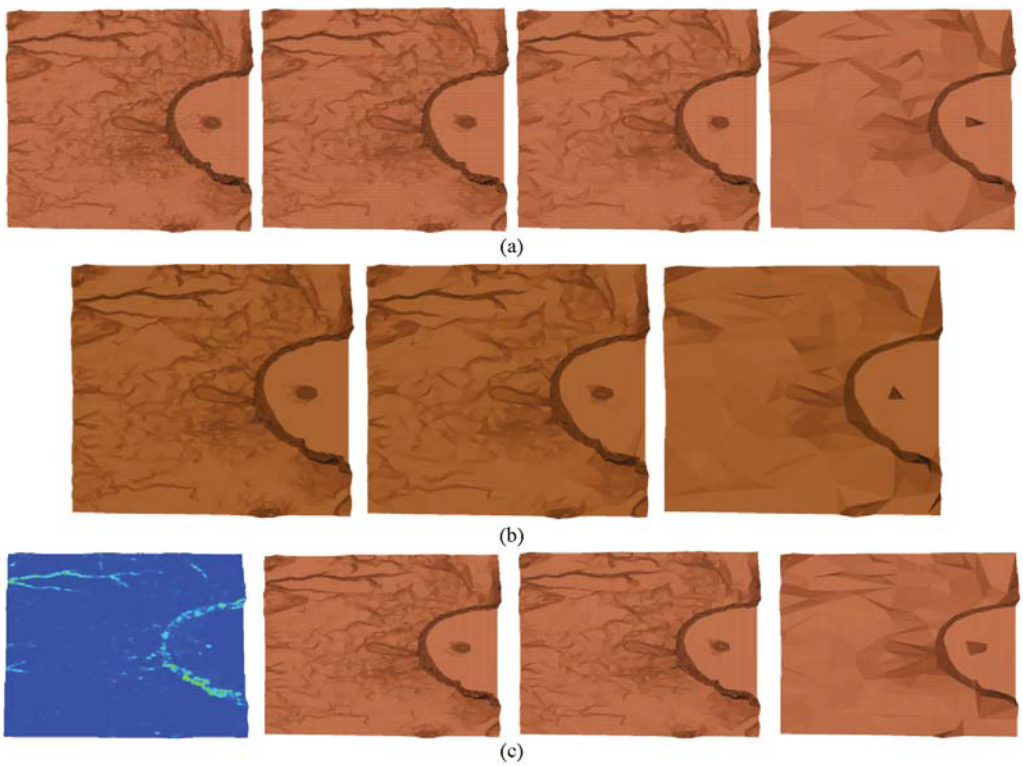


**Figure 8** The visual comparison of multi-resolution "Carter" model generated by QSlim, Geomagic 8 and our proposed method. (a) A original "Crater" model (199114 triangles) and the results from QSlim (4000, 2000, 300 triangles, respectively); (b) the results from Geomagic Studio 8 (4000, 2000, 300 triangles, respectively); (c) topographic feature map ($\sigma = 3\varepsilon$) and results from our scheme (4000, 2000, 300 triangles, respectively).

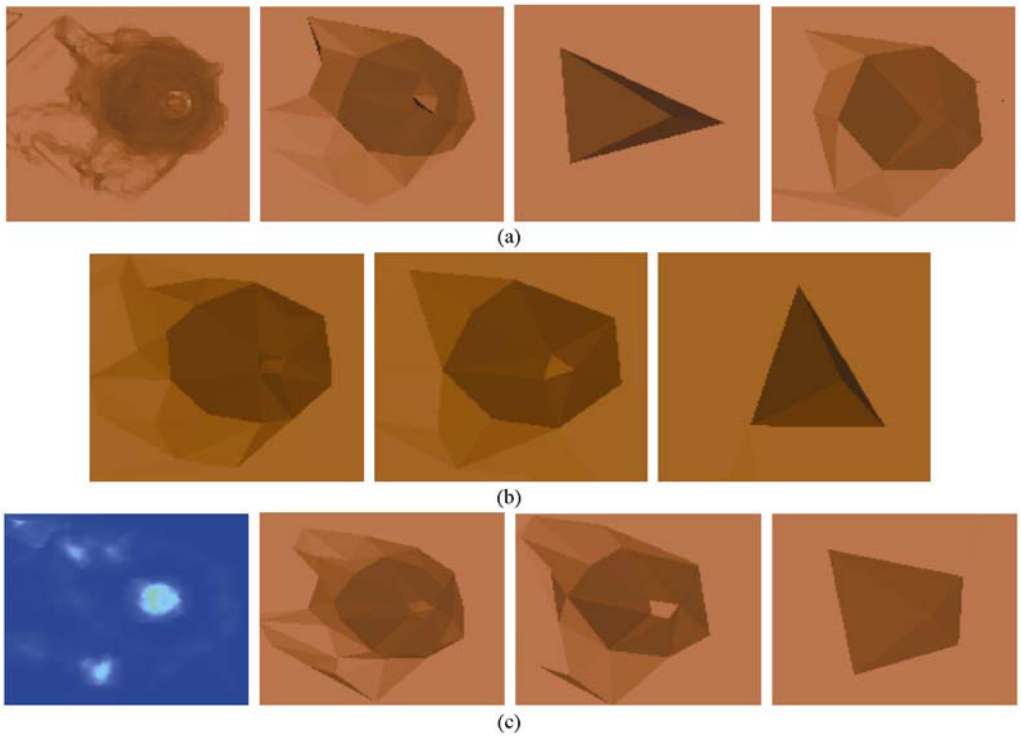


**Figure 9** The visual comparison of "peak" by QSlim (Garland[8]), Geomagic Studio 8 and our proposed method. (a) The "peak" of original "Crater" model and the results simplified by QSlim (4000, 2000 and 300 triangles, respectively); (b) the results simplified by Geomagic Studio 8 (4000, 2000 and 300 triangles, respectively); (c) topographic feature map ($\sigma = 3\varepsilon$) and the results simplified by proposed method (4000, 2000 and 300 triangles, respectively).

**Table 1** Error and execution time comparisons between QSlim 2.0, Geomagic Studio 8, our new QEM and topographic feature preservation ($\sigma = 3\varepsilon$) with the "Crater" model. Boldface indicates the best result

| TIN model | Triangle number (reduction ratio) | RMS geometric error | | | |
|---|---|---|---|---|---|
| | | QSlim | Geomagic | New QEM | Feature preservation |
| Crater (199 K triangles) | 4 K (98%) | 0.1230 | 0.1279 | 0.1210 | 0.1104 |
| | 2 K (99%) | 0.2043 | 0.2163 | 0.2031 | 0.1885 |
| | 300 (99.75%) | 0.7108 | 0.8051 | 0.7101 | 0.6892 |
| TIN Model | Triangle number (reduction ratio) | Hausdorff distance | | | |
| | | QSlim | Geomagic | New QEM | Feature preservation |
| Crater (199 K triangles) | 4 K (98%) | 1.8520 | 2.2480 | 1.8048 | 1.5058 |
| | 2 K (99%) | 2.1073 | 2.1302 | 2.1035 | 1.7950 |
| | 300 (99.75%) | 7.7880 | 5.3312 | 5.1869 | 5.0975 |
| TIN Model | Triangle number (reduction ratio) | Execution time | | | |
| | | QSlim | Geomagic | New QEM | Feature preservation |
| Crater (199 K triangles) | 4 K (98%) | 6.5156 | >30 | 5.8531 | 6.5022 |
| | 2 K (99%) | 6.5316 | >30 | 5.8363 | 6.5127 |
| | 300 (99.75%) | 6.5451 | >30 | 5.8552 | 6.5204 |

geometric error and Hausdorff distance. Execution time shows that our scheme with new error metrics is the fastest of the methods, followed by our new scheme with feature preservation ($\delta = 3\varepsilon$), QSlim 2.0 and Geomagic Studio 8.

## 5 Conclusions

In this paper, we proposed and implemented a half-edge collapse method for constructing multi-resolution model with topographic features preservation. We have developed a model of topographic feature detection using center-surround filters with Gaussian-weighted mean curvatures in a multi-scale manner. We elaborated the calculation of the new quadric error metric and the weights from topographic feature map and showed how incorporating feature weights can visually enhance the results. We used TIN models to evaluate the performance of our proposed scheme in terms of feature preservation, geometric errors and execution times. Moreover, the multi-resolution models generated by our method were compared with those did by QSlim 2.0 and Geomagic. The comparisons show that our method is able to generate visually superior and more accurate multi-resolution models, and preserve important topographic features particularly at a low LoD. Moreover, the new error metric evaluates more quickly. It should be equally easy to integrate the weight map with any other mesh simplification scheme, such as QSlim 2.0.


1 Wu L, Liu Y, Zhang J, et al. Geographical Information System: Theory, Method and Applications (in Chinese). Beijing: Science Press, 2001

2 Yang B S, Li Q Q, Gong J Y. A robust and rapid algorithm for generating and transmitting multi-resolution three-dimensional models. Chin Sci Bull, 2006, 51(8): 987－993

3 Wu L, Liu Y, Tang D S, et al. Interoperable and distributed geographical information systems based on web service. Geogra and Geo-Inf Sci (in Chinese), 2003, 19(4): 28－32

4 Luebke D, Reddy M, Cohen J D, et al. Level of Detail for 3D Graphics. San Francisco: Morgan Kaufmannv, 2002

5 Garland M, Heckbert P S. Surface simplification using quadric error metrics. In: Proceedings of SIGGRAPH 1997. New York: ACM Press, 1997. 209－216

6 Yang B S, Shi W Z, Li Q Q. A dynamic method for generating multi-resolution TIN models. Photogramm Eng Rem Sys, 2005, 8: 917－926

7 Lee C H, Varshney A, Jacobs D. Mesh saliency. ACM Trans Graph, 2005, 24(3): 659－666

8 Garland M. Quadric-based polygonal surface simplification. Dissertation of Doctoral Degree. USA: CMU, 1999

9 Wood J D. The geomorphological characterisation of digital elevation models. Dissertation of Doctoral Degree. UK: University of Leicester, 1996

10 Surazhsky V, Gotsman C. A qualitative comparison of some mesh simplification software packages. http://www.cs.technion.ac.il/~gotsman/AmendedPubl/Vitaly/SimpStudy-final.pdf (Preprint), 2005

11 Kobbelt L, Campagna S, Seidel H P. A general framework for mesh decimation. In: Proceedings of Graphics Interface 1998. Vancouver: A K Peters Ltd, 1998. 43－50

12 Garland M, Zhou Y. Quadric-based simplification in any dimension. ACM Trans Graph, 2005, 24 (2): 209－239

13 Cignoni P, Rocchini C, Scopigno R. Metro: Measuring error on simplified surfaces. Comput Graph Forum, 1998, 17(2): 167－174