# View-Dependent Multiresolution Representation for a Height Map

Yong H. Chung, Won K. Hwam, Dae S. Chang, Jung-Ju Choi, and Sang C. Park, *Member, IACSIT*

***Abstract*—Terrain visualization is a difficult problem for applications requiring accurate image of large datasets at high frame rates. We propose new view-dependent-multi-resolution representation method of large-scale terrain, which inherits the accuracy of 3D-methods as well as the memory efficient of 2.5D-methods. The proposed approach maintains a full decimation procedure of a height map for a maximum tolerance, and creates End-Start-tree, which is used to represent multi-resolution triangular mesh. For the approximation of a height map, the proposed algorithm employs the QEM(quadric error metrics) method. We demonstrate new visualization algorithm of large-scale terrain.**

***Index Terms*—Terrain, Height map, Digital elevation model, View-dependent LOD, Quadric error metrics, simplification.**

## I. INTRODUCTION

To efficiently visualize a large-scale terrain data, the use of a level-of-detail (LOD) has been widely accepted. LOD techniques increase the efficiency of rendering by decreasing the workload on graphics pipeline stages, usually vertex transformations. In general, relatively small objects either discrete LOD or continuous LOD is usually sufficient, but terrains have some special properties we can exploit compared to generic LOD algorithms. Because the terrain models are usually continuous and large, the same model can simultaneously be both very close to the viewer and very far away from him. This large size of the terrain models makes it unfeasible to use a single LOD throughout the model and thus necessitates view-dependent LOD.[1] View-dependent LOD can be thought as selective refinement of continuous LOD. Nearby regions of large objects can be rendered with more detail than the regions in the distance. View-dependent LOD meshes can be computed automatically using mesh simplification techniques [2].

Most algorithms expect the terrain to be stored as a height map which is a raster image used to store values in a uniform grid for display in 3D computer graphics. Previous results, which can be applied to terrain data simplification, can be roughly classified into two main categories; '3D-methods' [3]-[8], and '2.5D-methods' [9]-[11]. While 3D-methods deal with general 3D triangular meshes, 2.5D-methods are tailored to height fields which are 2.5D surface. The simplification methods for general 3D triangular meshes have extensively received increasing attention. Several algorithms have been formulated for simplifying surfaces. However, it is not possible to directly apply those algorithms to the simplification of a height map. A traditional way is to apply those algorithms after converting a height map into a triangular mesh. However, the conversion procedure will require an additional computational load as well as a large amount of memory, because the initial triangular mesh may contain millions of triangles. A typical data structure for a triangular mesh has the least amount information to represent the topology and geometry of a mesh. Usually, a general triangular mesh requires at least 20 times of memory amount than a height map does.

To avoid above difficulties, the simplification methods for 2.5D height map are developed. Most of 2.5D-methods for height map simplification are based on quad-tree or binary-tree.[1] Quadtrees and Binary trees are commonly used hierarchy for terrain tessellation. Both tree data structures produce semi-regular meshes and thus simplify the traversal algorithms and memory layout compared to the irregular meshes by 3D simplification methods. Usually, 2.5D-methods are more simple and more efficient than 3D-methods, because they do not have to maintain the complex topology information of a triangular mesh. However, 3D-methods approximate the original more accurately than 2.5D-methods by using the same number of triangles, because 3D-methods have no constraints in the shape of approximating triangles.

The objective of this paper is to develop a new multi-resolution representation method of terrains called a 'VD-MR-height-map(View-dependent-Multi-resolution-height-map)', supporting the hybrid simplification procedure for a height map, which inherits the accuracy of 3D-methods as well as the memory efficiency of 2.5D methods. The remainder of this paper is organized as follows: Section 2 presents the overall approach to the generation of view-dependent multi-resolution models, and Section 3 gives a detailed description of the proposed algorithm. Finally, concluding remarks are presented in Section 4.

## II. APPROACH

As mentioned above, the proposed procedure in this paper supports the view-dependent multi-resolution models of a height map by inheriting the accuracy of 3D-methods as well

Manuscript received September 20, 2012; revised November 19, 2012. This work was supported by the Defense Acquisition Program Administration (DAPA) and the Agency for Defense Development (ADD), Korea Research Foundation under the Contract No. UD110006MD (DAPA), UD100009DD, UD120035JD, 2012-0007496. The authors wish to express sincere gratitude for the financial support.

Y. H. Chung, W. K. Hwam, D. S. Chang and S. C. Park are with the Department of Industrial Engineering, Ajou University, Suwon, Republic of Korea (e-mail: yongho1230@gmail.com, lunacy@ajou.ac.kr, webmacome@ajou.ac.kr, scpark@ajou.ac.kr).

Jung Ju Choi is with the Department of Media, Ajou University, Suwon, Republic of Korea (e-mail: jungju@ajou.ac.kr).

DOI: 10.7763/IJIMT.2013.V4.369

as the memory efficiency of 2.5D methods. All of the existing algorithms that can interactively perform view-dependent, locally-adaptive terrain meshing, including ours, depend on a pre-defined multi-resolution terrain representation that is used to build the adaptive triangle mesh for a frame. The overall scheme of the proposed algorithm consists of two main steps, as shown Fig. 1: 1). The construction of VD-MR-height-map; 2) The extraction of simplified, view-dependent triangular mesh from the height map using VD-MR-height-map.

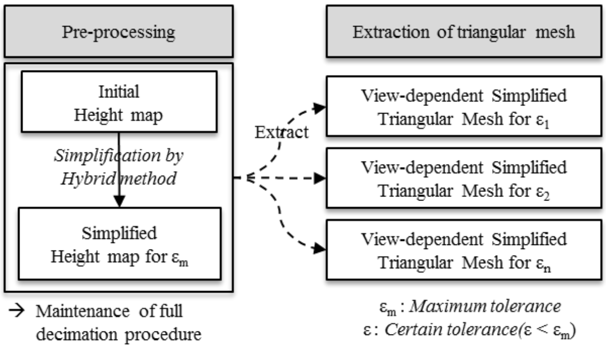


Fig. 1. Pre-processing and triangular mesh extraction of the proposed algorithm

For the first step, the proposed approach maintain a full decimation procedure of a height map for a maximum tolerance ($\varepsilon_m$), which defined by user. For the approximation of a height map, this paper employs the edge contraction approach based on the QEM (quadric error metrics) method[12]. The QEM is algorithm for simplification of polygonal models. Because the QEM assume that the model consists of triangles only, it is necessary to interpret the height map as a triangular mesh. To do so, it is necessary to determine each grid cell in a height map. The shorter of the two diagonals between the points is determined, and this is used to identify the two triplets of points that may become triangles.

$$\Delta(v) = \sum_{p\in planes(v)} (v^T p)(p^T v) = \sum_{p\in planes(v)} v^T (pp^T) v$$
$$= v^T \Big(\sum_{p\in planes(v)} K_p\Big) v = v^T (Q) v \qquad (1)$$

$$K_p = PP^T = \begin{bmatrix} a^2 & ab & ac & ad \\ ab & b^2 & bc & bd \\ ac & bc & c^2 & cd \\ ad & bd & cd & d^2 \end{bmatrix} \qquad (2)$$

The QEM method is based on the observation that each vertex is the solution of the intersection of a set of planes – namely, the planes of the triangles that meet at the vertex. It is possible to associate a set of planes with each vertex, and also to define the error of the vertex with respect to this set as the sum of squared distances to its planes, where $p = [a, b, c, d]^T$ represents the plane defined by the equation $ax+by+cz+d = 0$ where $a^2+b^2+c^2=1$. The error metric can be rewritten in quadratic form as equation (1).

The fundamental error quadric $K_P$ can be used to find the squared distance of any point in space to the plane p. We can sum these fundamental quadrics together, and represent an entire set of planes by a single matrix Q. For a vertex, the Q matrix can be computed by summing every KP of the triangles sharing the vertex. After computing the Q matrices for all vertices, it is possible to perform the edge contractions in order to simplify the geometry. While the original algorithm of the QEM method finds an optimal position for the contraction target, the proposed approach in this paper chooses $V_i$ or $V_j$ for the contraction target. To do this, the contraction cost in our approach is defined as follows:

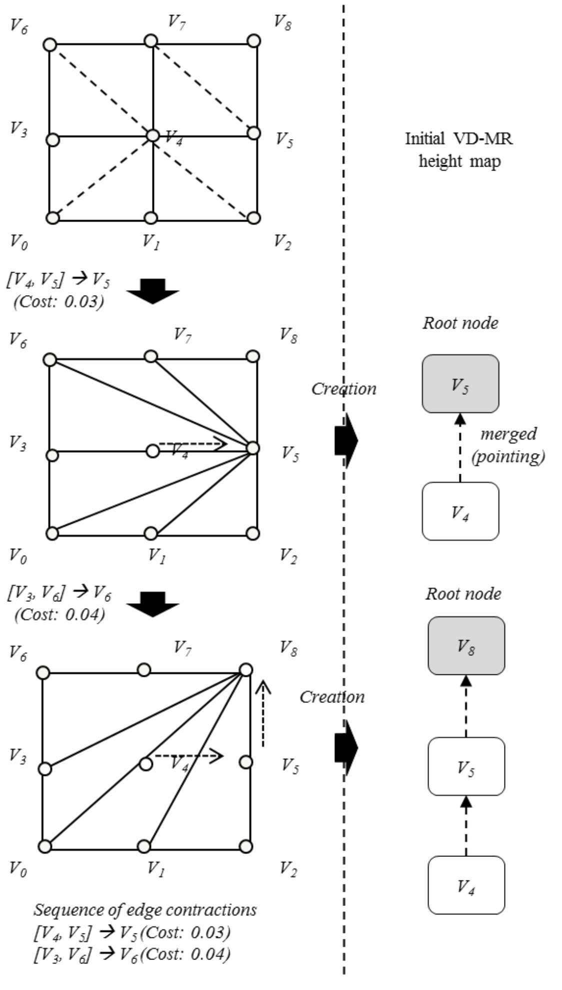
*Contraction cost of an edge [$V_i$, $V_j$]*

If ( $v_i^T (Q_i + Q_j) v_i$ < $v_j^T (Q_i + Q_j) v_j$ )

cost = $\sqrt{v_i^T (Q_i + Q_j) v_i}$ ; // $V_i$: contraction target, [$V_i$, $V_j$] →$V_i$

else

cost = $\sqrt{v_j^T (Q_i + Q_j) v_j}$ ; // $V_j$: contraction target, [$V_i$, $V_j$] →$V_j$



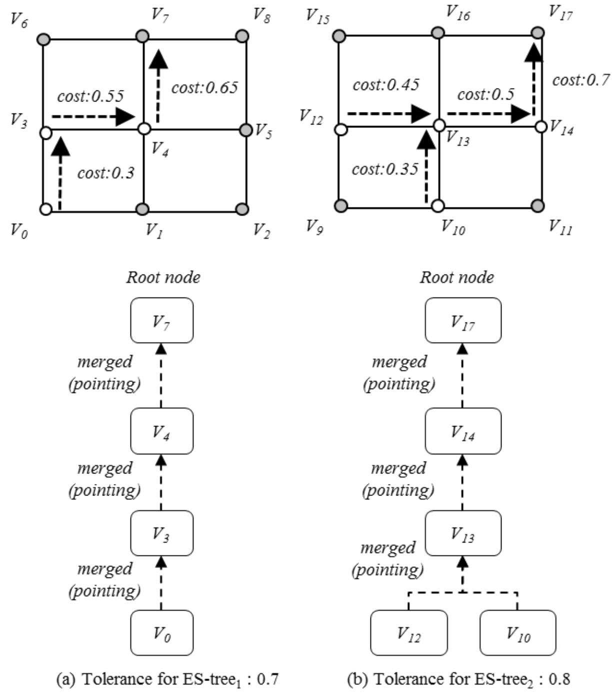
(a) Sequence of edge contractions (b) End-Start tree

Fig. 2. Sequence of edge contractions.

Through the decimation procedure, edge contractions cause the topology irregular as shown in Fig. 2. VD-MR-height map is efficient data structure to represent the topology irregular of a height map. While a height map is an array of real numbers, a VD-MR-height-map is an array of containers, each of which can store a real number as well as the decimation related information. As a preprocessing step, a tree of a VD-MR-height-map is built. If a node is not a root node, the node has a parent node. The parent node of child node cannot be more than two. If a tree consists of a parent node and a child node, the parent node is ‘a end vertex’, the child node is ‘a start vertex’. In this paper, this tree, called a ‘ES-tree’, is used to represent multi-resolution triangular mesh. As a preprocessing step, height map consists of several ES-tree which has each tolerance. ES-tree is used to build continuous adaptive meshes. For view-dependent visualization of height map, each ES-tree has certain tolerance which is proportional to the distance between root node and camera.

## III. ALGORITHM

As mentioned above, a VD-MR-height-map is an array of containers, each of which can store a real number as well as the decimation related information. A VD-MR-height-map has the following information; 1) *value*, a real number indicating the height; 2) *Q*, a matrix for QEM; 3) *End_id*, the id of the end of this vertex if the vertex is merged; 4) *cost*, the edge contraction cost if the vertex is merged; 5) *Tree_tolerance*, the contraction cost for ES-tree. Fig. 4 shows local areas of a height map. ES-tree consists of four nodes, and ES-tree has tolerance 0.7. For example, $V_4$ of the VD-MR-height-map contains following information, $V_4$[*value* = height$_4$, *Q* = $Q_4$, *End_id* = id of $V_7$, *cost* = 0.65, *Tree_tolerance* = 0.7]. Because $V_1$, $V_2$, $V_5$, $V_6$, $V_8$ do not have End_id, they are also root node. Thus, there are six ES-trees in Fig. 3(a). In the same way, Fig. 3(b) shows another ES-tree$_2$ which has tolerance 0.8. The tolerance 0.8 of ES_tree$_2$ represents that the distance between the root node of ES-tree$_2$ and camera is farther than the distance between the root node of ES-tree$_1$ and camera. However, it is not that all node of ES-tree$_2$ in comparison with ES-tree$_1$ is far.



(a) Tolerance for ES-tree$_1$ : 0.7 (b) Tolerance for ES-tree$_2$ : 0.8

Fig. 3. VD-MR-height-map and ES-tree for the tolerance

An edge contraction makes the topology irregular, and the VD-MR-height-map store the topology changes by using the ‘End_id’, instead of maintaining a heavy data structure such as a general triangular mesh. The construction procedure of the VD-MR-height-map can be described as Fig. 4.

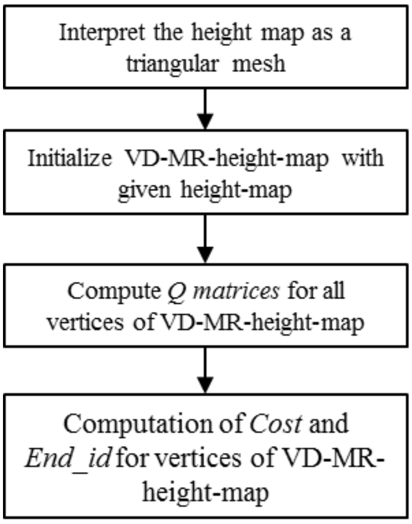


Fig. 4. *Tree-tolerance* computation procedure

Once the VD-MR-height-map is constructed for $\varepsilon_m$, we can extract a simplified triangular mesh for certain *Tree-tolerance* ($0 < \varepsilon_t < \varepsilon_m$) by applying edge contractions having costs less than ε. The Tree tolerance $\varepsilon_t$ is decided by the distance between the root node of ES-tree and camera, as follow :

*Tree_tolerance = Distance * Simplification coefficient*
*- Simplification coefficient is static.*
*- Simplification coefficient is decided by some tests.*
*- Tree_tolerance is smaller than maximum tolerance*($\varepsilon_m$)

The procedure to compute Tree-tolerance can be described as follow:

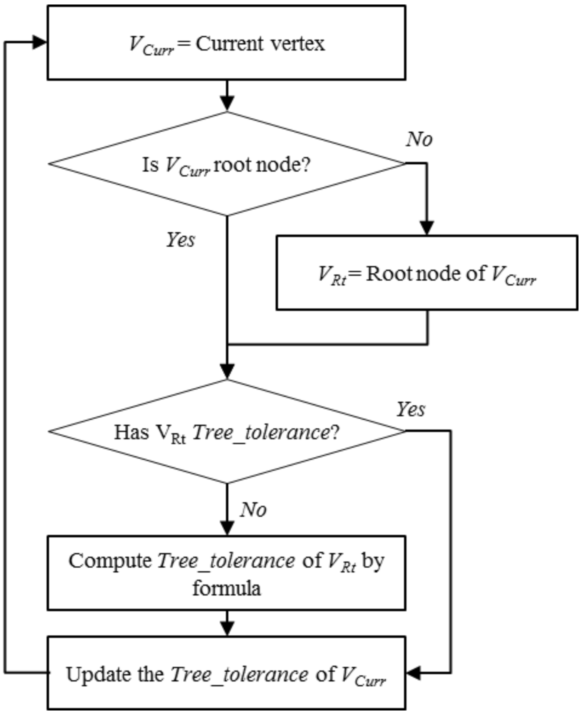


Fig. 5. *Tree_tolerance* computation procedure

As shown Fig. 5, the Tree-tolerance of a certain vertex is decided by root node. Thus, all nodes of a certain ES-tree have same Tree-tolerance.

The generation procedure of a simplified triangular mesh from a VD-MR-height-map can be described as follow. A certain cell consists of two triangles. For each vertex of the two triangles, we can trace the end vertex in a recursively way for a certain tolerance. The recursive tracing procedure can be described, as shown in Fig. 6.

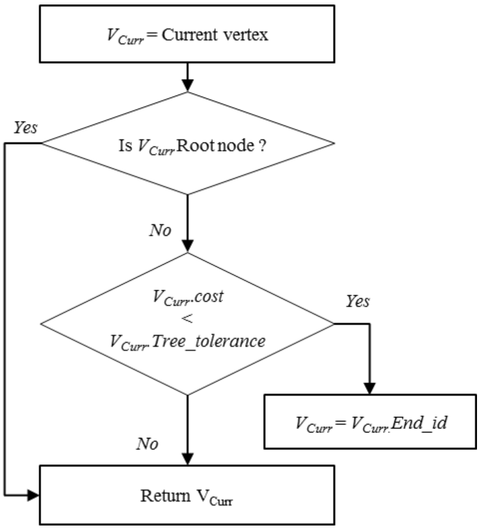
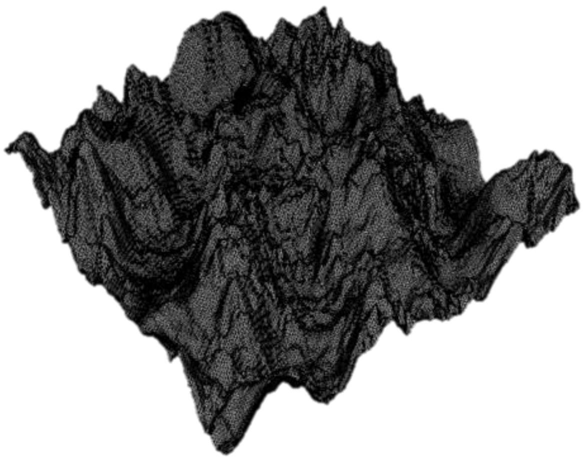


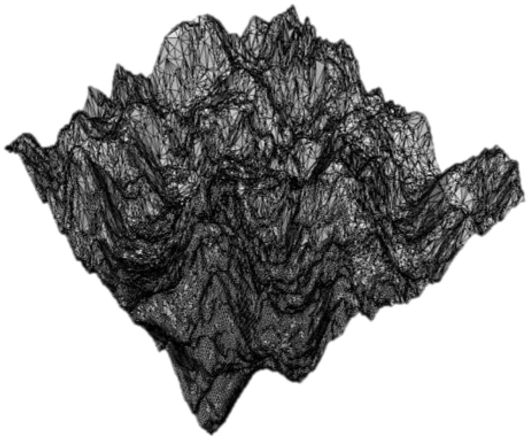
Fig. 6. The generation procedure of a simplified triangular mesh

Fig. 7 shows simplified triangular meshes extracted with view-dependent LOD. Because maximum tolerance for pre-processing is equal, the number of ES-trees of each terrain is equal. Nearby regions of terrain data can be rendered with more detail than the regions in the distance, and view-dependent LOD for terrain data results from simplification coefficient, as shown in Fig. 7(b) and Fig. 7(c).



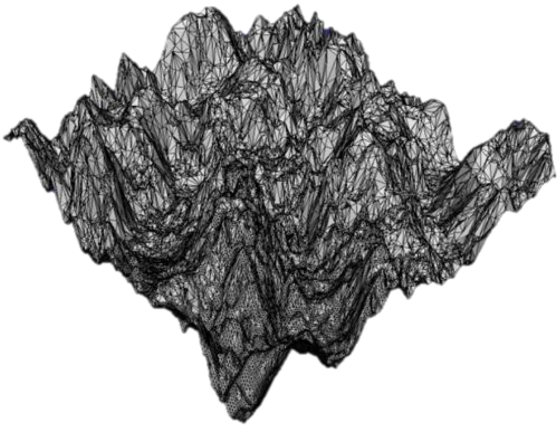
(a) 100% (Original height map)

- Maximum Tolerance : 1.0
- # of ES-trees : 8,703
- # of Triangles : 99,012



(b) 57% of the original number of triangles

- Maximum Tolerance : 1.0
- # of ES-trees : 8,703
- Simplification Coefficient : 0.002
- # of Triangles : 56,616



(c) 41% of the original number of triangles

- Maximum Tolerance : 1.0
- # of ES-trees : 8,703
- Simplification Coefficient : 0.003
- # of Triangles : 40,281

Fig. 7. Simplified triangular mesh with view-dependent LOD

## IV. CONCLUSION

For visualization of large-scale terrain, view-dependent LOD technique is essential. View-dependent LOD tries to construct the best representation for the current view, and critical component of it is a simplification technique. Previous results, which can be applied to terrain data simplification, can be roughly classified into two main categories; ‘3D-methods’, and ‘2.5D-methods’. Usually, 2.5D-methods are more simple & memory efficient than 3D-methods, however, 3D-methods approximate the original more accurately than 2.5D-methods by using the same number of triangles.

This paper presents new view-dependent-multi-resolution representation method of large-scale terrain, which inherits the accuracy of 3D-methods as well as the memory efficient of 2.5D-methods. For the approximation of a height map, the proposed algorithm employ the edge contraction approach based on the QEM (quadric error metrics) method. The overall scheme of the proposed algorithm consists of two main steps:

1) The construction of VD-MR-height-map; 2) The extraction of simplified, view-dependent triangular mesh. For the first step, the proposed algorithm maintains a full decimation procedure of a height map for a maximum tolerance, and creates ES-tree, which is used to represent multi-resolution triangular mesh. For the second step, a simplified triangular mesh can be traced to the end vertex in a recursively way for a certain tolerance.

## ACKNOWLEDGMENT

This work was supported by the Defense Acquisition Program Administration (DAPA) and the Agency for Defense Development (ADD), Korea Research Foundation under the Contract No. UD110006MD (DAPA), UD100009DD, UD120035JD, 2012-0007496. The authors wish to express sincere gratitude for the financial support.

## REFERENCES

[1] M. A. Duchaineau, M. Wolinsky, David E. Sigeti, M. C. Miller, C. Aldrich, and M. B. Mineev-Weinstein, “ROAMing terrain: real-time optimally adapting meshes,” *97 proceedings of the 8th conference on Visualization*, 1997, pp. 81-88.

[2] H. Hoppe, “View-dependent refinement of progressive meshes,” *97 proceedings of the 24th annual conference on Computer graphics and interactive techniques*, 1997, pp. 189-198.

[3] R. Sivan, and H. Samet, “Algorithms for Constructing Quadtree Surface Maps,” *In Proceeding 5th International Symposium on Spatial Data Handling*, 1992, pp. 361-370.

[4] H. S. Kim, H. K. Choi and K. H. Lee, “Novel Evaluation of the Edge Cost for Mesh Simplification with Vertex Color Attribute,” *Proceedings of the Society of CAD/CAM Engineers Conference*, 2006, pp. 764-770.

[5] H. S. Kim, H. K. Choi, and K. H. Lee, “Mesh Simplification for Preserving Geometric and Color Feature”, *Proceedings of the Society of CAD/CAM Engineers Conference*, 2007, pp. 253-259.

[6] W. D. Li, Y. L. Cai, and W. F. Lu, “A 3D Simplification Algorithm for Distributed Visualization,” *Computers in Industry*, vol. 58, issue. 3, pp. 211-226, 2007.

[7] K. H. Lee, H. Woo, and T. Suk, “Data Reduction Methods for Reverse Engineering,” *The International journal of advanced manufacturing technology*, vol. 17, no. 10, pp. 735-743, 2001.

[8] H. Hoppe, “Progressive Meshes,” *In Proceeding SIGGRAPH*, pp. 99-108, 1996.

[9] D. Hebert and H. Kim, “Image encoding with triangulation wavelets,” *In Proceeding SPIE*, pp. 381-392, 1995.

[10] P. Lindstrom and D. Koller, “Real-time Continuous Level of Detail Rendering of Height Fields,” *In Proceeding SIGGRAPH*, pp. 109-118, 1996.

[11] R. Sivan and H. Samet, “Algorithms for Constructing Quadtree Surface Maps,” *In Proceeding 5th International Symposium on Spatial Data Handling*, pp.361-370, 1992.

[12] M. Garland, and P. S. Heckbert, “Surface Simplification Using Quadric Error Metrics,” *In Proceeding SIGGRAPH*, pp. 209-216, 1997.

**Yong H. Chung** was born in Korea, and received his BS degree from Ajou University in 2011 He is currently pursuing his master’s degree in the Department of Industrial Engineering at Ajou University, Republic of Korea. He is interested in CAD/CAM, Robotics, Modeling & Simulation. Mr. Chung is conducting research in the fields of geometric algorithms in CAD/CAM.

**Won K. Hwam** was born in Korea, and received his BS degree from Ajou University in 2011. He is currently pursuing his master’s degree in the Department of Industrial Engineering at Ajou University, Republic of Korea. He is interested in Distributed System, Synthetic environment, Modeling & Simulation. Mr. Hwam is conducting research in the fields of Defense Modeling & Simulation.

**Dae S. Chang** was born in Korea, and received his BS degree from Ajou University in 2012. He is currently pursuing his master’s degree in the Department of Industrial Engineering at Ajou University, Republic of Korea. He is interested in Discrete event system, Modeling & Simulation. Mr. Chang is conducting research in the fields of CAD/CAM/CAPP software system.

**Jung Ju. Choi** was born in Korea. He is an Associate Professor of Digital Media at Ajou University, Korea. He received BS degree from KAIST in 1990, and received his MS, Ph.D. degrees in Computer Science from POSTECH in 1992, 1997. His main research interests include non-photorealistic animation and rendering, image and video processing, and mobile graphics systems.

**Sang C. Park** is an assistant professor in the Department of Industrial & Information Systems Engineering at Ajou University. Before joining Ajou, he worked for DaimlerChrysler Corp. and Cubick-Tek Co., developing commercial and inhouse CAD/CAM/CAPP/simulation software systems. He received his BS, MS, and PhD degrees from KAIST in 1994, 1996, and 2000, respectively, all in industrial engineering. His research interests include geometric algorithms in CAD/CAM, process planning, engineering knowledge management, and discrete event system simulation.