Direct Extraction of a Simplified Triangular Mesh from a Range Image

Sang C. Park[1], Chang M. Park[2], T. J. Park[3] and H. T. Park[4]

[1]Ajou University, scpark@ajou.ac.kr
[2]Ajou University, cmpark@ajou.ac.kr
[3]Ajou University, legrandblue99@hanmail.net
[4] Ajou University, taiji416@lycos.co.kr

ABSTRACT

Presented in the paper is a procedure for the direct extraction of a simplified triangular mesh from a range image. Although there are many existing algorithms for computing a simplified geometric model, we can not use those algorithms for a range image because most of them have been developed in the context of triangular meshes. An intuitive way is to apply those algorithms after converting a range image into a triangular mesh (initial triangular mesh). However, the conversion procedure will require additional computational load as well as a large amount of memory because the initial triangular mesh has to contain millions of triangles. Since the proposed approach makes use of the inherent attributes of a range image for extracting a simplified triangular mesh directly, we do not have to maintain a heavy triangular mesh containing millions of triangles

Keywords: simplification, range image, flag map, Z-map.

1. INTRODUCTION

There has been much interest in developing techniques of non-contact measurement of 3D objects. The mapping of 3D objects and reconstructing them in a CAD system is a subject of optical measurement and geometrical modeling, and it is becoming an increasingly important topic in computer vision, with applications in fields such as reverse engineering, inspection of geometrical properties of mechanical components, object recognition, guidance for robotic vision, medical applications, and virtual environment construction. For the non-contact measurement based on vision cameras, there have been two alternative techniques: stereovision and structured light system (SLS). Structured light system [7,8] avoids the so-called correspondence problem of passive stereo vision, and getting more popular in industry inspections due to fast measuring speed, very simple optical arrangement, non-contact, moderate accuracy, low cost, and robust nature in the presence of ambient light source in situation. Three-dimensional range data are spatial coordinates for surface points of an object and they are useful for 3D object matching, object recognition, and dimensional measurement. A structured lighting system is a general method of acquiring surface range data from a scene. For such systems, an object is illuminated from a structured light source and this scene is imaged by a camera. An example of the output from a structured light system is shown in Figure 1.

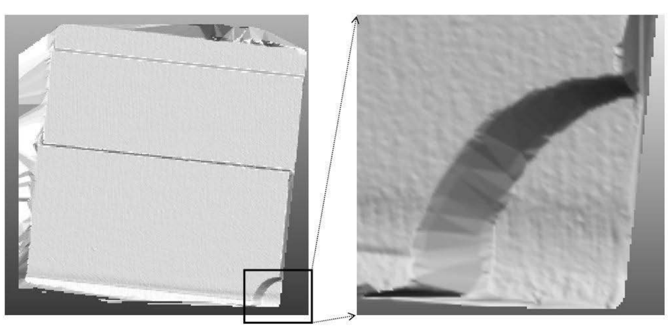


Fig. 1. A range image.

A range image from SLS can be considered as a Z-map which is a special form of a discrete non-parametric surface model [1]. A Z-map is a 2D-array of real numbers in which Z-values of the surface, sampled at the regular grid-points, are stored. For rendering a range image (Z-map), we need to extract triangles from the range data, which can be done intuitively. The number of points of a range image depends on the resolution of the camera. The range image, shown in Figure 1, is captured by a camera with a 1000x1000 resolution, and contains 740,000 points. For rendering, we need to triangulate the range image, and the triangulation gives 1.5 million triangles. Many applications in CAD/CAM require complex, highly detailed models. Consequently, the range images are often acquired at a very high resolution to accommodate this need for detail. However, the full complexity of such models is not always required, and the required level of detail actually may vary significantly according to the application. Highly detailed geometric models are necessary to satisfy a growing expectation for realism in computer graphics, while the increasing complexity of the models makes them expensive to store, transmit, and render. To speedup the processing of complex model, model simplification and multi-resolution models are motivated whenever a representation at low resolution is adequate to the needs of the applications. Since the simplification problem of a complex geometric model has been paid a great amount of attention by many researchers, many simplification algorithms have been developed, which can be classified into two main categories; 1) Vertex decimation[5,6], and 2) Edge contraction[3,4]. Although there are many proven simplification algorithms, it is not easy to apply those algorithms to the simplification of a range image directly, because most of them are developed in the context of triangular meshes. An intuitive way is to apply those algorithms after converting a range image into a triangular mesh (initial triangular mesh). However, the conversion procedure will require additional computational load as well as a large amount of memory because the initial triangular mesh has to contain millions of triangles. Figure 2 shows a typical data structure for a triangular mesh having the least amount information for representing the topology and the geometry. A range image captured by a 2 Mega pixel camera requires at most 8 Mega bytes (2M*4), because it consists of 2 million float numbers, depth values (Z-values) at the regular grid positions. However, the required amount of memory for the initial triangular mesh for the range image becomes 130 Mega bytes (4M*24+2M*17), because the initial triangular mesh has to contain 4 million triangles and 2 million vertices. Considering the triangular mesh structure has least information, the actual required memory for simplification would be much greater than just the size of the triangular mesh. That serves as a motivation for exploring the possibility of finding a procedure through which a simplified triangular mesh can be directly generated from a range image without converting the range image into an initial triangular mesh.

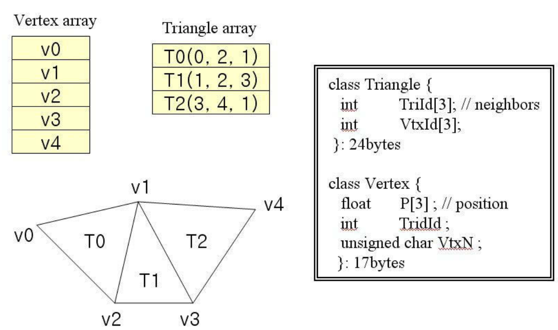


Fig. 2. A basic data structure for a triangular mesh.

The objective of this paper is to develop a procedure extracting a simplified triangular mesh from a range image directly. Since the proposed approach makes use of the inherent attributes of a range image (i.e. Z-map attributes) for extracting a simplified triangular mesh directly, we do not have to maintain a heavy triangular mesh containing millions of triangles. The remainder of this paper is organized as follows. Section 2 presents the overall approach to the extraction of a simplified triangular mesh from a range image, and Section 3 describes the details of the proposed algorithm with some illustrative examples. Finally, concluding remarks are presented in Section 4.

2. APPROACH TO EXTRACT A SIMPLIFIED TRIANGULAR MESH FROM A RANGE IMAGE

The input of the problem is a range image and a tolerance, and our objective is to develop an algorithm which extracts the simplified geometry in the form of a triangular mesh. Other than the computational efficiency in terms of time and memory, there is a very important technological requirement for the algorithm, ‘Feature preservation’. The simplified triangular mesh should preserve the feature shapes as much as possible with minimum number of triangles. The proposed approach employs the iterative contraction of edges (vertex pairs). An edge contraction, which we denote [vi, vj] → vi (or vj), connects all incident edges of vj to vi, and delete the vertex vj. Subsequently, any edges or faces which have become degenerate are removed, as shown in Figure 3. Normally, an edge contraction removes an edge and two triangles sharing the edge. Although the edge contraction can be implemented easily on a triangular mesh, it is not very intuitive to implement the operation on a range image which does not support the irregular topology. The behind idea of the paper is to employ an additional data structure, called a 'flag map' (to be described later), for a range image to support the irregular topology like a triangular mesh. While a range image is an array of depth values (float numbers), a flag map is an array of containers each of which can store topology related information. For the simplification, the paper employs the edge contraction approach based on the QEM (quadric error metrics) method [2].

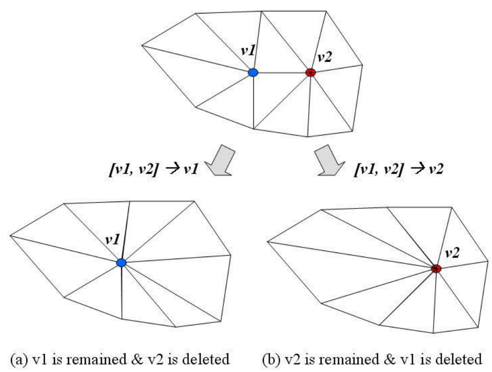


Fig. 3. Examples of edge contractions.

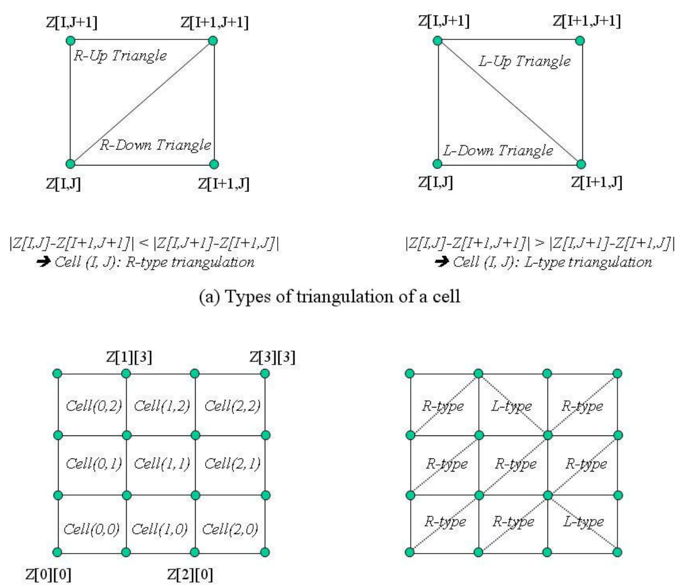


Fig. 4. Triangulation of a range image.

The overall scheme of the proposed algorithm consists of four main steps: 1) Initialization of a flag map by interpreting a given range image as a triangular mesh; 2) Compute the Q matrices (quadric error matrices [2]) for all vertices in the range image; 3) Perform edge contractions, and update the flag map to reflect the topology changes caused by the edge contractions; and 4) Extract a simplified triangular mesh from the range image by using the flag map. Let's assume there is a given range image, which is a regular array having Ni*Nj Z-values (vertices). For the first step, it is necessary to interpret the range image as a triangular mesh. To do so, we need to decide the triangulation of each grid cell of the range image (Figure 4). For a grid cell, there are only two choices, R-type or L-type. As shown in Figure 4-(a), we use a simple rule to determine the triangulation type of a cell. We can then store the triangulation type of each cell (R-type or L-type) to the corresponding grid point (container) of the flag map, an Ni*Nj array of containers.

Before explaining the second step of the proposed approach, it is necessary to address a brief explanation on the concept of a Q matrix (Details of the QEM method can be found in [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 (triangles) with each vertex, and we can define the error of the vertex with respect to this set as the sum of squared distances to its planes:

$$\Delta(v) = \Delta([v_x, v_y, v_z, 1]^T) = \sum_{p \in planes(v)} (p^T v)^2$$

where $\boldsymbol{p} = [a, b, c, d]^T$ 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 as a quadratic form:

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

where K_p is the matrix:

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

The fundamental error quadric K_p can be used to find the squared distance of any point in space to the plane $\boldsymbol{p}$. We can sum these fundamental quadrics together and represent an entire set of planes by a single matrix Q. Remember that the second step of the proposed approach is to compute Q matrices of all vertices in the range image. For a vertex in a range image, the Q matrix can be computed by summing all fundamental error quadric (K_p) of the triangles sharing the vertex, as shown in Figure 5.

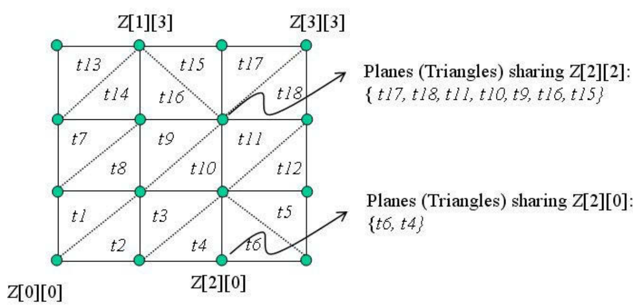


Fig. 5. Planes for computing the Q matrix for a vertex.

After computing Q matrices for all vertices, we can perform edge contractions (the third step of the proposed approach) to simplify the geometry. To do so, it is necessary to compute the contraction cost for every edge [vi, vj], which can be described as follows.

■ ***Contraction cost of an edge [vi, vj]***

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

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

else

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

Since the contraction cost can be considered as the sum of distances to the planes sharing vi or vj, we can contract the edge if the cost is smaller than the given tolerance. We implicitly track sets of planes using a single matrix; instead of computing a set union (planes(v1) U planes(v2)) we simply add two quadrics (Q1+Q2). The details of the edge contraction procedure will be addressed in the next section.

3. EDGE CONTRACTION PROCEDURE

The basic structure of the edge contraction procedure can be summarized as follows: 1) Compute the contraction cost for every edge of the range image; 2) Place all edges in a heap keyed on the contraction cost with the minimum cost edge at the top; and 3) Iteratively remove the edge (V_i, V_j) of least contraction cost from the heap, contract the edge and update the costs of all edges affected by the contraction. The computing method of the edge contraction cost is already mentioned in the previous section. The most important issue here is how to maintain the topology changes caused by the edge contractions. As mentioned earlier, we employ an additional data structure, a flag map, to maintain the topology changes. Figure 6-(a) shows a simple range image consisting of 9 grid points. Let's assume that the edge ($V_{0,1}$, $V_{1,1}$) has the minimum contraction cost with the contraction target $V_{0,1}$. We can then perform the edge contraction, [$V_{0,1}$, $V_{1,1}$] → $V_{0,1}$, and the resulting geometry is shown in Figure 6-(b). But the problem here is that the regular-grid structure of a range image can not represent the resulting geometry. That's why we need an additional data structure, a flag map. Figure 6-(d) shows an initial flag map having the triangulation type of every cell and the Q matrices for all vertices. To represent the resulting geometry after the edge contraction, [$V_{0,1}$, $V_{1,1}$] → $V_{0,1}$, we need to update the flag map instead of changing the range image. As shown in Figure 6-(e), the updated flag map shows two differences from the initial flag map; 1) The Q matrix of $V_{0,1}$ becomes $Q_{0,1}$+$Q_{1,1}$, and 2) The vertex $V_{1,1}$ has an arrow (pointer) to the vertex $V_{0,1}$, which indicates that $V_{1,1}$ is no longer a valid vertex because it has been merged into $V_{0,1}$. In this case, we call the vertex $V_{0,1}$, an 'ancestor vertex' of $V_{1,1}$. After the first edge contraction, it is necessary to update the costs of the remaining edges affected by the contraction, and reorder the heap. For the next stage, we assume that the edge ($V_{0,1}$, $V_{0,2}$) has the minimum contraction cost with the contraction target $V_{0,2}$. Then, we can contract the edge, [$V_{0,1}$, $V_{0,2}$] → $V_{0,2}$, and the resulting geometry is shown in Figure 6-(c). The corresponding flag map is shown in Figure 6-(f). After the second edge contraction the Q matrix of $V_{0,2}$ becomes $Q_{0,2}$+$Q_{0,1}$+$Q_{1,1}$, and we call $V_{0,2}$, an ancestor vertex of $V_{0,}$

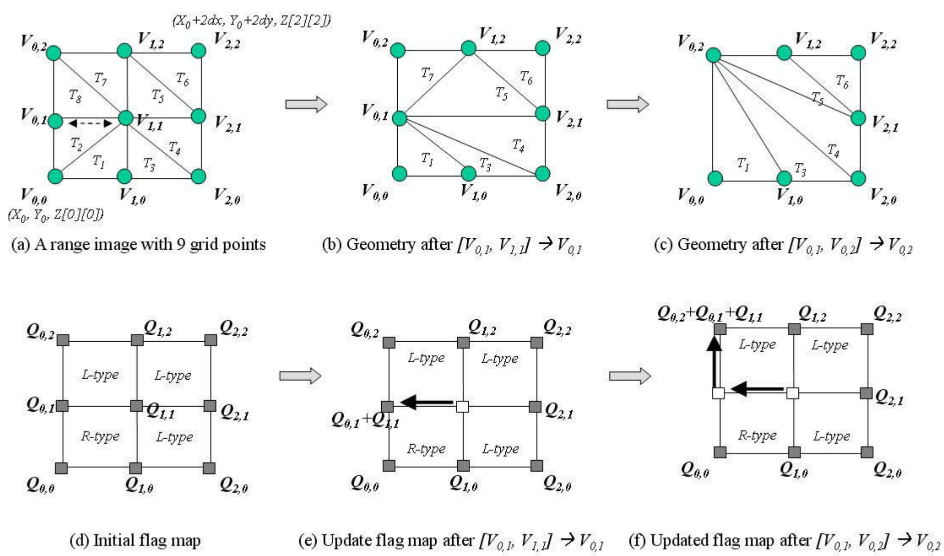
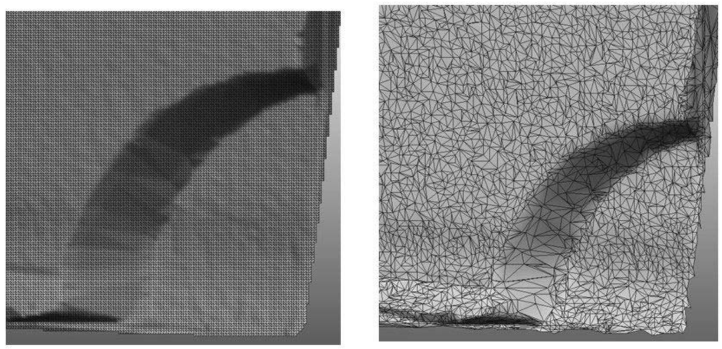


Fig. 6. An example of edge contractions.

As a result of the edge contractions shown in Figure 6, we obtain the flag map (Figure 6-(f)) instead of the simplified geometry (Figure 6-(c)). This means we need to have a method to extract the final geometry from the flag map and the original range image. As shown in Figure 6-(f), we know the triangulation types of all cells from the flag map. From the triangulation type, we can identify the two original triangles for each cell. For example, the original triangles of Cell(0,0) are T_1:($V_{0,0}$, $V_{1,1}$, $V_{0,1}$) and T_2:($V_{0,0}$, $V_{1,0}$, $V_{1,1}$), because it is a R-type cell. But the final flag map shows that the two vertices $V_{1,1}$ and $V_{0,1}$ are no loner valid, because they are merged into their ancestor vertex $V_{0,2}$. Considering the invalid vertices, the triangles T_1:($V_{0,0}$, $V_{1,1}$, $V_{0,1}$) and T_2:($V_{0,0}$, $V_{1,0}$, $V_{1,1}$) should be modified as T_1:($V_{0,0}$, $V_{0,2}$, $V_{0,2}$) and T_2:($V_{0,0}$, $V_{1,0}$, $V_{0,2}$), respectively. Observe that T_1 is an invalid (degenerate) triangle because it has two same vertices. As a result, Cell(0,0) produces only one triangle, T_2:($V_{0,0}$, $V_{1,0}$, $V_{0,2}$), after the simplification procedure. In this way, we can extract the simplified triangular mesh, shown in Figure 6-(c), directly from the flag map. Figure 7 shows an example of the simplification procedure.



(a) Range Image (100%) (b) Simplified Triangular Mesh (7%)

Fig. 7. A simplified triangular mesh from a range image.

4. SUMMARY

This paper presents a procedure to extract a simplified triangular mesh from a range image directly, without converting the range image into an initial triangular mesh having millions of triangles. Since the proposed approach extracts simplified triangles directly from a range image, we do not have to convert the range image into a triangular mesh (initial triangular mesh) which requires a large amount of memory. One of the major problems is that how to maintain the topology changes, caused by edge contractions, without having a triangular mesh. The key idea of the paper is to employ an additional data structure, called a 'flag map', for a range image to support the irregular topology changes happening during the simplification procedure. While a range image is an array of depth values, a flag map is an array of containers each of which can store topology related information. Since a flag map maintains all topology changes during the simplification procedure, we can easily extract simplified triangular mesh from a flag map and a range image after the simplification procedure.

5. REFERENCES

[1] Choi, B. K. and Jerard, R.B., *Sculptured Surface Machining – theory and applications*, Kluwer Academic Publishers, 1998.

[2] Garland, M. and Heckbert, P. S., Surface simplification using quadric error metrics, Proceedings of SIGGRAPH 97, 1997, pp 209-16.

[3] Hoppe, H., Progressive meshes, Proceedings of SIGGRAPH 96, 1996, pp 99-108.

[4] Hoppe, H., DeRose, T., Duchamp, T., McDonald J. and Stuetzle, W., Mesh optimization, Proceedings of SIGGRAPH 93, 1993, pp 19-26.

[5] Schroeder, W. J., Zarge, J. A. and Lorensen, W. E., Decimation of triangle meshes, *Computer Graphics*, Vol. 26, No. 3, 1992, pp 65-70.

[6] Soucy, M. and Laurendeau, D., Multiresolution surface modeling based on hierarchical triangulation, *Computer Vision and Image Understanding*, Vol. 63, No. 1, 1996, pp 1-14.

[7] Son, S., Park, H. and Lee, K., Automated laser scanning system for reverse engineering and inspection, *International Journal of Machine Tools & Manufacture*, Vol. 42, 2002, pp 889-97.

[8] Valkenburg, R. J. and Mcivor, A. M., Accurate 3D measurement using a structured light system, *Image and Vision Computing*, Vol. 16, 1998, pp 99-110.