Int J Adv Manuf Technol (2001) 18:201–210

The International Journal of
Advanced
Manufacturing
Technology

Point Data Reduction Using 3D Grids

K. H. Lee, H. Woo and T. Suk

Kwangju Institute of Science and Technology (K-JIST), Korea

Reverse engineering refers to the process of obtaining a CAD model from an existing physical part. Advances in laser scanning technologies have facilitated this process by sampling part surface data with speed and accuracy. With the help of this technology, it is now possible to acquire the geometry of a part having complex and freeform surfaces. However, it creates the burden of large amounts of point data which must be manipulated, therefore, data from a laser scan must be significantly reduced to proceed with the computations and to lower the storage requirement. Many point data reduction methods for image processing have been developed in the past. However, there is little published work on laser-scanned data, and what exists focuses only on 2D point data.

This paper presents a data reduction method that reduces the amount of 3D point data using part geometry information. The method reduces the point data, based on normal values of points using 3D grids. The method is applied to two sample models and the results are discussed.

Keywords: Data reduction; Reverse engineering; Freeform surfaces; 3D grid

1. Introduction

Reverse engineering in this study refers to a technology that allows a CAD model to be obtained from an existing physical part. This is necessary when the CAD model for an existing part is not available for various reasons, e.g.:

1. When a clay model is first built by a designer.
2. When a part has undergone many design changes.
3. When the drawing of a part is lost or no longer available.

In a reverse engineering process, a CAD model is created based on the point data sampled from the surfaces of a part. Initially, point data are generated by scanning or probing the part surface. These point data generally require pre-processing operations, such as the arrangement of points or the removal of spikes. The pre-processed point data are then segmented if a surface model is required. A polygonised model, such as one with a stereolithography (STL) format, can also be generated from the point data.

In capturing the surface data of a part, either a contact-type measuring device or a non-contact-type device has been used. The contact-type devices provide better accuracy, but these are generally very slow for point data acquisition. Recent advances in laser scanning technologies have provided a non-contact-type measuring device that can scan parts with a very high speed and with good accuracy. These machines enabled us to capture the surface data of a part with complex and freeform surfaces. Now, however, another problem has been created because of the large amount of point data that is obtained in the data acquisition step. These data require a large storage space and increase computational time significantly.

Data reduction, therefore, has become an important issue in reverse engineering. Many data reduction methods have been proposed in the area of image processing, but they were mostly designed for dealing with mesh point data. Only a few methods were developed that could be applied directly to the point data generated from measurement devices, but they were also limited to specific types, such as 2D or planar sets of point data. The proposed 3D grid-based point data reduction method can be applied to a 3D point cloud data. This paper first reviews the existing point data reduction methods in Section 2. The 3D grid method is described in Section 3, and the proposed method is applied to two sample models and the results are discussed in Section 4.

2. Review of the Past Point Data Reduction Methods

In visualising or analysing a scanned object, meshed models, in particular those using triangular patches, have been used. For these triangulated models, data reduction is performed by reducing the number of triangles based on application-dependent criteria; then, with the remaining nodes, triangulation is performed repeatedly. The procedure of a conventional reduction method is shown in Fig. 1. Data reduction methods based on meshed models can be divided into two categories:

Correspondence and offprint requests to: Dr K. H. Lee, Kwangju Institute of Science and Technology (K-JIST), 1 Oryong-dong, Puk-gu, Kwangsan-ku, Kwangju, 500–712, Korea. E-mail: lee@kyebek.kjist.ac.kr

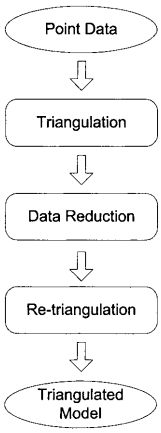

Fig. 1. The procedure of a conventional reduction method based on triangulated models.

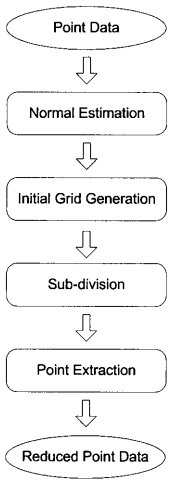

Fig. 2. The procedure of the 3D grid method.

manipulation of the triangulated point data and the use of level-of-detail (LOD) methods.

Chen et al. [1] proposed a genetic algorithm for optimised retriangulation, in which they presented an optimised STL file generation method for reducing laser scanned data. Data reduction was performed by decreasing the number of triangles in an STL file, using the normal vectors of the triangles. After removal of some of the triangles, retriangulation is performed. Hamann [2] presented a new reduction method for triangulation files based on an iterative triangle removal principle. As a measure of reduction in the file size, each triangulation is weighted according to the principal curvature estimates at its vertices and according to the interior angles. Véron and Léon [3] introduced an approach to reduce the number of points in a polyhedral model using error zones assigned to each point of the initial polyhedron, so that the simplified polyhedron intersects with each error zone. Schroeder et al. [4] have developed an algorithm that simplifies the mesh by removing vertices. Vertices are identified through a distance-to-plane criterion, where an average plane is formed through a vertex and through its adjacent vertices. If the vertex is within a specified distance of the average plane, the vertex is deleted. Hoppe et al. [5] developed a data reduction scheme in terms of a mesh optimisation problem, by ordering the edges according to an energy minimisation function.

LOD construction algorithms can be applied to structured or unstructured 3D meshes. Fischer and Park [6] generated multilevel-of-detail models for design and manufacturing in reverse engineering. The 3D meshes are represented by a 2D structure: the quadtree structure. The proposed algorithm extracts one level from the multilevel model according to a given error tolerance [7,8]. Using this method, the reconstructed geometric model is represented by hierarchical levels of detail.

For reducing the laser scanned data, conventional sampling methods, such as uniform sampling, chordal deviation sampling, and space sampling, have been used widely because of their simplicity and fast computation time [9,10]. The uniform sampling method reduces the number of points in a data set, with the user-specified sampling rate. In this method, the point cloud is sampled at every ith point, where i is the sampling rate. This method requires the points in the scanned data to be ordered in advance. The chordal deviation method selectively extracts points from the point clouds, while preserving the original point cloud to a specified accuracy. This operation is performed defining two parameters: the maximum deviation distance, and the maximum distance between points. The maximum deviation distance guarantees that all removed points are within the specified tolerance from the remaining ones. The maximum distance between points guarantees that if the distance to a considered point from the last retained point is larger than the specified distance, the considered point is retained. This method can preserve the points of high variation and information to define shape variations in the data. It can also be used to identify feature lines and edges. The space sampling method extracts points with a defined 3D neighbourhood size. In this method, only points that are further away than a specified distance from one another are kept. All these sampling methods are dependent on the order of points in the scan data.

The data reduction method proposed in this paper uses a grid method and the related research is described below.

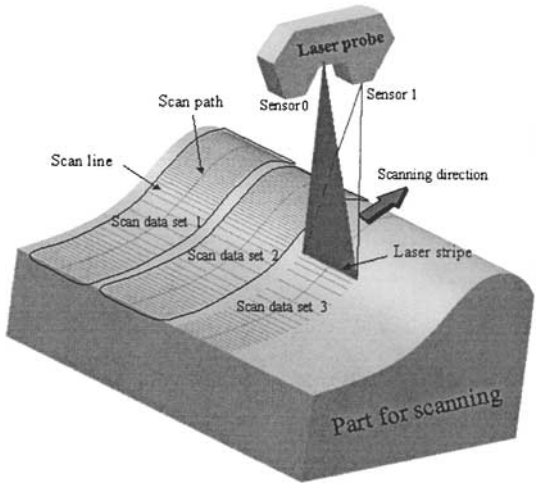

Fig. 3. Laser scanning.

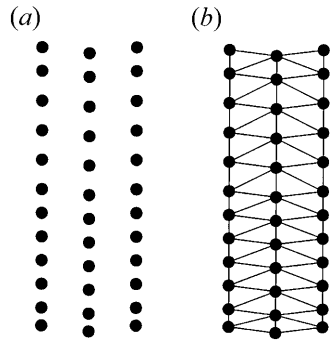

Fig. 4. Triangulation of point data. (*a*) Point data measured by a line laser scanner. (*b*) Triangulated point data.

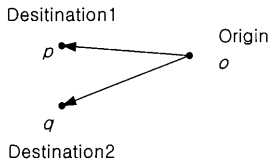

Fig. 5. Two edges of a triangle.

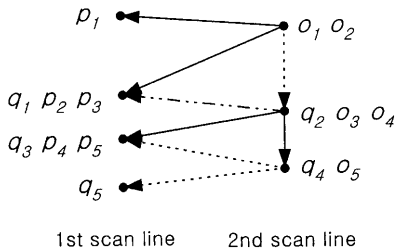

Fig. 6. Edge generation of two scan lines.

Martin et al. [11] proposed a data reduction method using 2D uniform grids in their EU Copernicus project. They used uniform sized 2D grids with a "median filtering" approach, which has been widely used in image processing. The procedure starts by choosing a grid structure, and input data points are assigned to the corresponding grid. For the points assigned to a given grid, a point in the median location is selected to represent data points belonging to that grid. This approach intends to overcome the problem of z-axis inaccuracies, resulting from laser scanning, that are not resolved by averaging or simple sampling methods. It, however, has drawbacks since it uses only uniform grids without any consideration of the part shape.

Lee et al. [12] and Suk [13] proposed improved 2D grid methods that effectively reduce point data, using part geometry information. They used two methods that can be applied to different part shapes:

1. A simple part shape, such as quadrics.
2. A complicated shape with freeform surfaces.

The one-directional non-uniform grid method can be applied to the former and the bidirectional non-uniform grid method to the latter. The one-directional non-uniform grid method extracts points where a significant curvature change occurs and

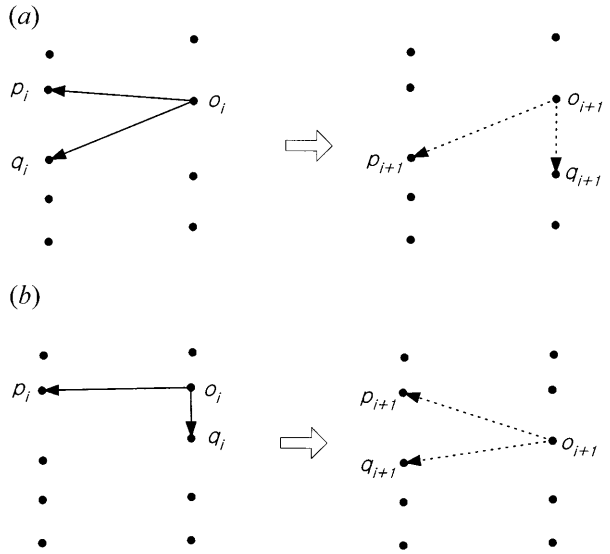

Fig. 7. Two cases in edge determination. (*a*) Case 1. (*b*) Case 2.

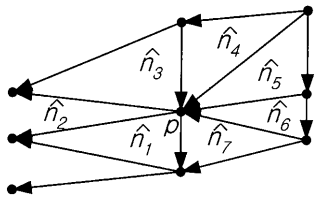

Fig. 8. Normal calculation.

non-uniform grids are generated along a direction based upon these points. If the length of a grid is too long, it is divided into maximum allowable lengths defined by the user. After generating the non-uniform grids, a representative point is selected within each cell, based on median filtering.

The bidirectional non-uniform grid method uses point normal values as part geometry information. A planar set of point data is gridded in the same manner as for the uniform grid method and each grid is subdivided based on its point normal values. During subdivision of the grids, the quadtree is used, that is, a grid is divided into four subgrids, if necessary.

With the non-uniform grids generated after subdivision, a representative point is extracted from each grid. These 2D grid methods, however, work only for data acquired with one scanning direction and they require the merging of point data after data reduction if a complete 3D point data model is needed.

In this paper, a 3D grid method that can overcome the limitations of the previous methods is proposed. This method deals directly with the entire 3D point data constructed by the registration and integration of multiple scanned data sets and does not require merging of points in advance.

3. The 3D Grid Method

The 3D grid method can handle the entire surface of a 3D object, whether it is a single point cloud or consists of multiple

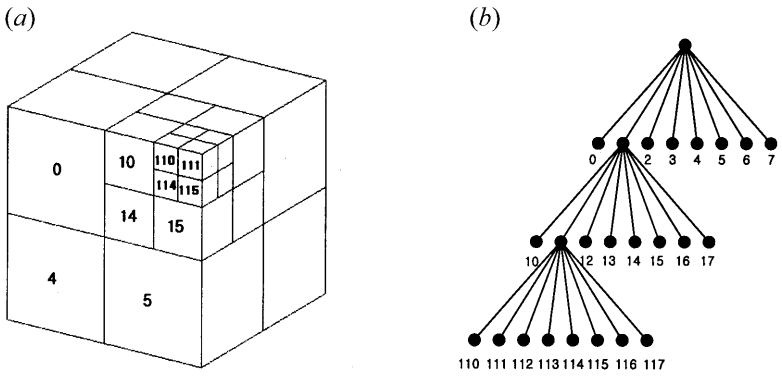

Fig. 9. Octree structure. (*a*) Cube having octants after subdividing. (*b*) Octree.

point clouds. However, where an object consists of multiple point clouds, they must be registered under a single coordinate system. The proposed method uses point normal values on the surface of the part, from which 3D non-uniform grids are generated using the standard deviation of the normal values. Data reduction is performed by selecting one representative point and discarding other points from each grid. The procedure of the 3D grid method is shown in Fig. 2, and the details are described in the following subsections.

3.1 Normal Estimation

To obtain information about the surface geometry of a part, various curvature values, such as Gaussian, mean, and principal curvatures, have been used. Since it is difficult to extract this information about the surface of the part directly from point data, the model is usually triangulated; so that, the curvature data or normal values of the triangles are obtained. In the proposed method, the normal values of points, the so-called point normals, are used instead to extract the geometric information of the part.

Depending on the scanning devices or scanning methods, point data can be classified as structured or unstructured. The point data obtained by laser scanners or Moiré scanners are listed as structured data whereas the point data obtained by portable coordinate measuring machines (CMMs) or handheld scanners are unstructured. In estimating point normals, Delaunay triangulation is usually used, regardless of point data types. For unstructured point data, point normals are calculated after Delaunay triangulation. For structured point data, however, different normal estimation methods can be used, considering the pattern of the point data. In this paper, a quick and simple normal estimation algorithm is developed for the point data obtained by a laser scanner.

Point data generated by a laser scanner with a stripe type light source has an inherent order. Since a scan path is defined as a series of line segments as shown in Fig. 3, each line in the path is ordered, as are the points in each scan line. The normal estimation values can be calculated quickly using this pattern.

In the scanning operation of non-contact devices such as laser and Moiré scanners, an object surface must be measured from one direction or from multiple directions. When a single direction is used, the part surface can be scanned at once at one orientation. When multiple directions are required, the part must be scanned several times at different orientations. In general, because of model complexity and measurement limitations, multiple digitised data sets scanned with different orientations are required to cover the entire surface of a 3D object. Upon completion of scanning all surfaces, the point clouds from different views should be registered together to construct a complete 3D model [14].

The 2D Delaunay triangulation can be used for normal estimation of the point data obtained from a single direction. Delaunay triangulation can be applied directly to the given set of points, and this method has been used widely for generating the triangular mesh from scattered point data. A Delaunay triangle has the property that its circumscribing circle does not contain any other point [15]. Two-dimensional algorithms can be used for constructing the triangles from the single-view scan data set. However, 2D Delaunay triangulation contains unnecessary operations and it takes a long time for the structured point data acquired by the laser scanner. The point normal estimation algorithm used in this paper is described below. After triangulation, point normals are calculated from the circumscribed triangles.

3.1.1 The proposed normal estimation method

A part is usually scanned several times at different orientations and for every scan, a single-view point cloud is obtained. For these point clouds, 2D triangulation can be used for calculating the normal estimates. Figure 4(*a*) shows a set of point data viewed from the laser probe. In order to estimate point normals,

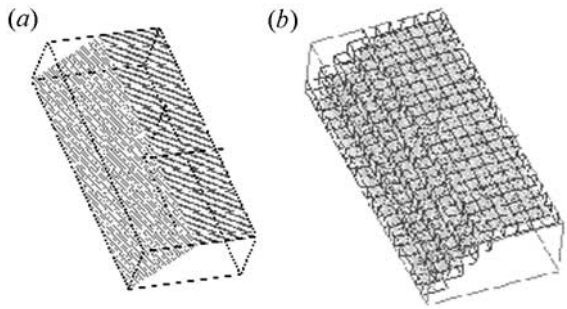

Fig. 10. Bounding box and initial grids.

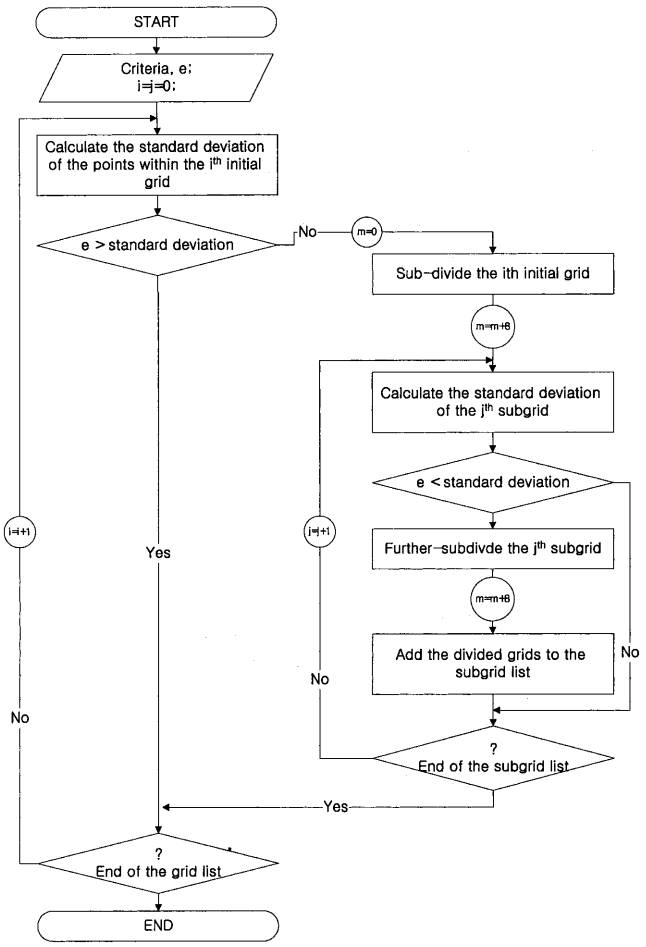

Fig. 11. Procedure of grid subdivision.

the data set from each scan shown in Fig. 3 must be triangulated. On the data arranged in this way, triangulation can be performed based on two neighbouring scan lines. Figure 4(*b*) shows an example of triangulation between the scan lines. Three vertices of a triangle cannot be in one scan line; that is, when two vertices are in one scan line, the last one must be in the other. Therefore, two edges are used for calculation since they are sufficient to obtain the normal value of a triangle, as shown in Fig. 5. Using the cross-product of the two edges, the normal value of a triangle is calculated and it is normalised as shown in Eq. (1).

$$\hat{N} = \frac{\mathbf{op} \times \mathbf{oq}}{|\mathbf{op} \times \mathbf{oq}|} \tag{1}$$

It is assumed that the origin of each two-edge stays on the second line, as shown in Fig. 6. Since this method is based on 2D triangulation and the x-coordinate values of each scan line are the same, the distance along the y-axis is the criterion for deciding edges. With the given two scan lines, the first edge for the first two-edge runs from the first point, o_1, of the second line to the first point, p_1, of the first line. Once the first edge is decided, the destination of the second edge must be the next closest point to the origin. When the destination vertex, q_i, of the second edge is selected from the first line, the second edge, $\overline{o_i q_i}$, becomes the first edge, $\overline{o_{i+1} p_{i+1}}$, of the next two-edge (see Fig. 7(*a*)). When the destination vertex, q_i, is selected from the second line, a new edge, $\overline{o_{i+1} p_{i+1}}$, that starts from the destination vertex, q_i, of the second edge to the destination point, p_i, of the first two-edge is generated for the first edge, $o_{i+1} p_{i+1}$, of the next two-edge so that the origin, o_i, moves to the destination point, q_{i+1}, of the second edge (see Fig. 7(*b*)). By determining the edges with this strategy, the normal orientations of triangles are arranged.

After calculation of normal value estimates for all two-edges, point normals are determined. As an example, for point P shown in Fig. 8, a group of two-edges that has P as one of their vertices is found. To determine the point normal estimate at the point, the normal values of the group are averaged and normalised using the following equation:

$$\hat{p}_{normal} = \frac{\sum_{i=1}^{m} \hat{n}_i}{m} \Bigg/ \left| \frac{\sum_{i=1}^{m} \hat{n}_i}{m} \right| \tag{2}$$

where n is the number of two-edges.

3.2 3D Grid Subdivision Using an Octree

Spatial decomposition methods based on octree structures have been proposed for use as approximate representations of geometric objects [16,17]. The basic concept of the octree representation is of placing the object of interest in a parallelepiped, typically a cube, which totally encloses it. As shown by Fig. 9, this parallelepiped is then subdivided into its 8 octants, which are then recursively subdivided a number of times based on the criteria defined by the application. In approximating a geometric object, the octants completely inside or outside the object are not subdivided further, while those octants which contain a portion of the object's boundary continue to be subdivided to the required level. The concept of octrees is used here for data reduction. The criterion used for subdividing a cube is the standard deviation of point normal values.

3.3 Grid Generation and Subdivision

After calculating the point normals, the normal values are stored using a point data structure, which has x-, y-, and z-coordinates and x-, y-, and z-normal components. Then, all the point clouds belonging to an object must go through a registration process.

After the registration is completed, a bounding box is created and then the initial grid is generated. According to the shape of the scanned parts, the number of initial cells should be specified. The shortest axis of the bounding box among the x-, y-, and z-axes, is selected to decide this number, which is determined by dividing the shortest axis of the bounding box with the interactive input-value by the user.

Figure 10(*a*) shows a bounding box encasing a part that has a sloped surface, and Fig. 10(*b*) shows its initial grid and cells. Among these cells, unnecessary ones that do not contain

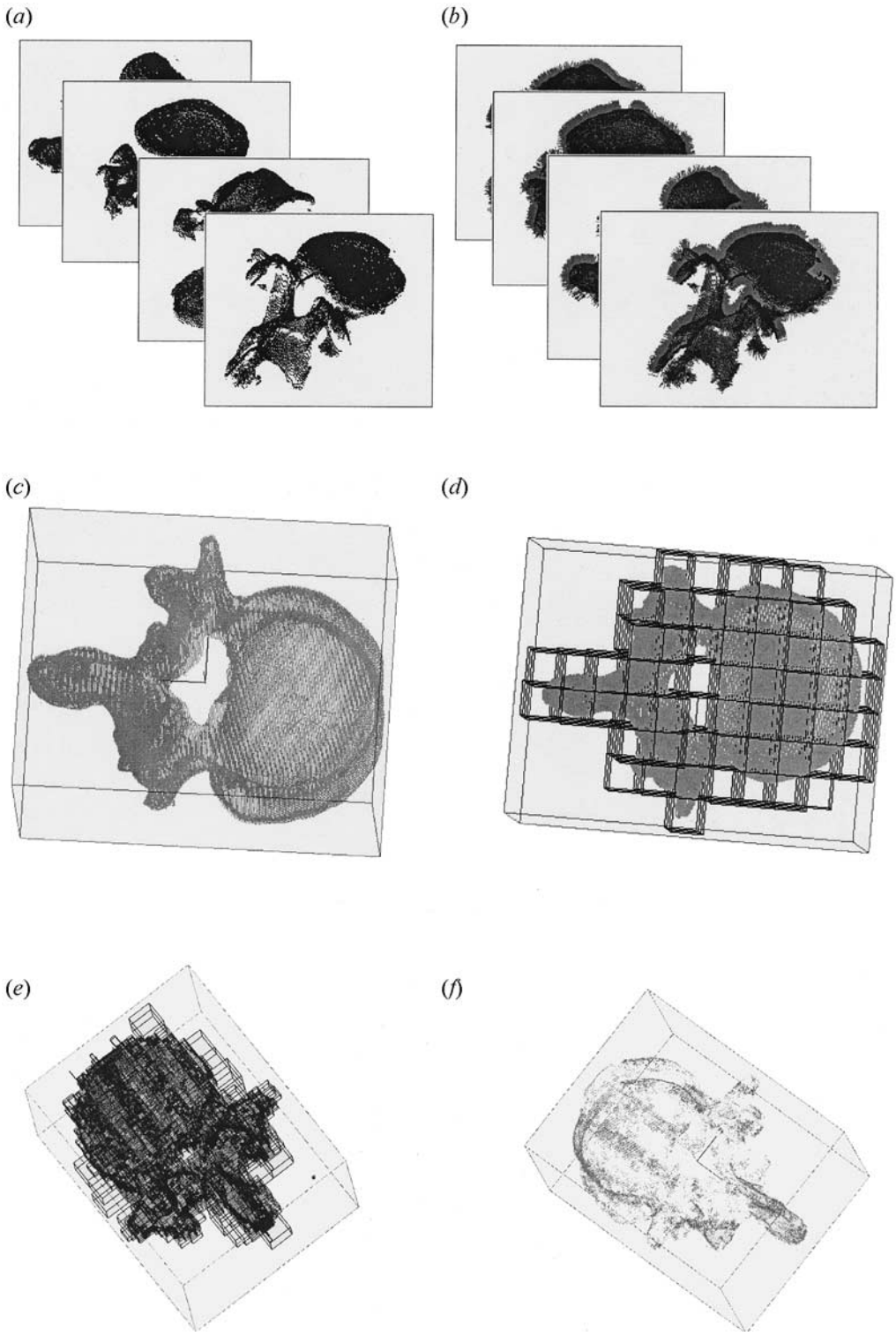

Fig. 12. Example of a human vertebra. (*a*) Scanned point data. (*b*) Point data with normal. (*c*) Complete model with the bounding box. (*d*) Initial grids. (*e*) Non-uniform 3D grids. (*f*) Reduced point data.

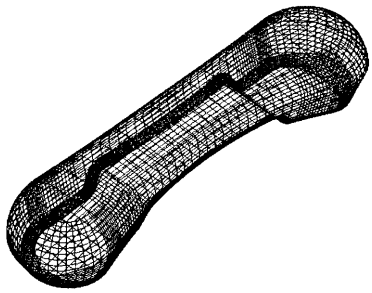

Fig. 13. The CAD model of the phone.

any points are eliminated. Then each cell is subdivided using octree decomposition and empty cells are again eliminated. For subdivision, the standard deviation of point normals within each initial cell is calculated. When the standard deviation is larger than the user-defined tolerance, the cell is divided into 8 cells. This subdivision process continues until the divided cells meet the termination conditions. The termination conditions are met when the standard deviation of point normals within a cell is smaller than the given tolerance, or a cell contains only one point. Figure 11 shows the subdivision process using a flowchart.

3.4 Extraction of Points

As a result of cell subdivision, many cells are generated where the part geometry changes greatly, whereas a few grids appear where the part geometry shows little change. From these cells, the points that can represent part geometry are extracted. In selecting a point that represents the points within a given cell, the average of the normal values is used. Therefore, a point whose normal value is closest to the average of the points within the cell is chosen, and the selected point is regarded as the most representative point among the points within that cell.

In this method, the level of data reduction is determined by two factors: the number of initial cells, and the size of the user-defined tolerance. For surface fitting of a plane, as an example, the data points must appear within a certain interval so that a surface fitting operation can be performed. In this case, the size of the initial cells is determined by these intervals, after which the number is decided accordingly.

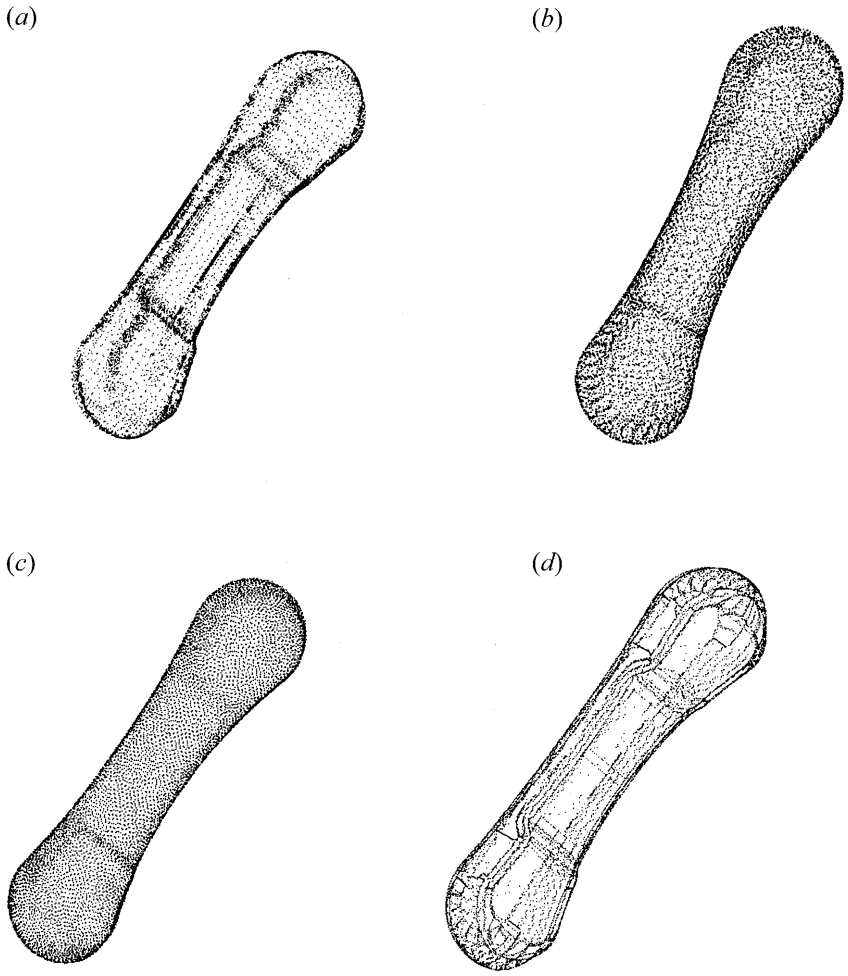

Fig. 14. Reduced point data of the phone. (*a*) 3D grid method. (*b*) Uniform sampling. (*c*) Space sampling. (*d*) Chordal deviation sampling.

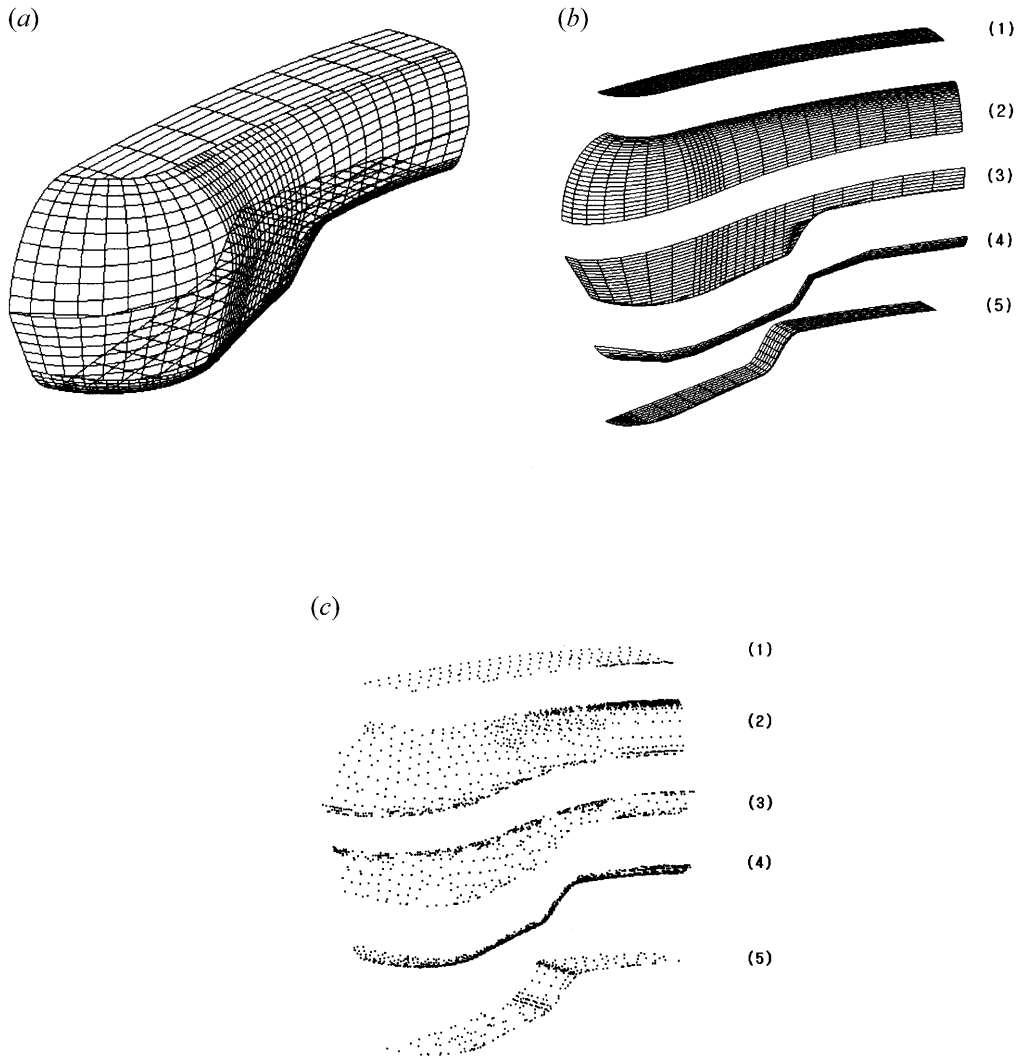

Fig. 15. Quarter section of the phone model. (*a*) Quarter section of the model. (*b*) Five surface patches. (*c*) Segmented point data.

In the developed program, the number of initial cells is determined before the tolerance. The tolerance is the main factor in data reduction; the smaller the tolerance the greater the number of points that remain, and vice versa. In order to fit a surface to the reduced point data, the data should be distributed evenly. The distance between reduced points is dependent on the size of the initial cells, which may affect the accuracy of the model. If the user wants the remaining points to be distributed evenly to maintain accuracy, a great number of initial grids must be used. If the user wants a lower number of points while still maintaining the accuracy, the reduction process should start with a lower number of initial cells and a higher tolerance.

4. Application Examples

The proposed grid methods were applied to sample parts and the results are discussed for two models. In implementing the data reduction methods, a commercial laser scanner, Surveyor 1200 from Laser Design was used to acquire the point data and a commercial CMM, Bright Apex-204 from Mitutoyo was used to measure planar surfaces. A reverse engineering software package, Surfacer 7.01, was used for the sampling of data points, surface fitting, and error estimation. The algorithm for the 3D grid method was developed and written in C++, MFC (Microsoft Fundamental Classes), and OpenGL. The program for the 3D grid method was run on a Pentium II 350 PC.

4.1 Human Vertebra Model

The data reduction methods were tested using point data from a section of a human spine. Figure 12 shows the point data of a human vertebra during implementation of the 3D grid method. To obtain a full 3D model, the spine section was scanned at four orientations, as shown in Fig. 12(*a*). The point normals were then calculated for each set of point data, as shown in Fig. 12(*b*). After calculating the point normals, the point clouds from different orientations were registered in a single coordinate system to make a full 3D model. In order

Table 1. Comparison result for the 3D grid method.

Ratio	Number of points	Maximum deviation (mm)	Average deviation (mm)
Original			
86 448			
	Three-dimensional grid method		
1/3	29 120	0.16275	0.00975
1/5	17 736	0.18698	0.01398
1/10	8900	0.26807	0.01599
1/15	6975	0.57787	0.03605
1/50	1730	1.06150	0.11289
	Uniform sampling		
1/3	28 816	0.21641	0.01612
1/5	17 300	0.36726	0.02003
1/10	8645	0.34369	0.04270
1/15	6650	1.03837	0.05203
1/50	1729	2.70593	0.26496
	Space sampling		
1/3	29 156	0.26130	0.01697
1/5	17 754	0.43019	0.01826
1/10	8987	0.54235	0.02827
1/15	6981	1.30583	0.04716
1/50	1746	2.05261	0.26764
	Chordal deviation sampling		
1/3	29 124	0.24843	0.01231
1/5	17 736	0.36557	0.01705
1/10	8920	0.30546	0.03169
1/15	6962	0.95479	0.07427
1/50	1741	2.49202	0.36324

to make 3D non-uniform cells, a bounding box that encases the full 3D model was created, as shown in Fig. 12(*c*). Based on the bounding box, the initial cells were generated. Figure 12(*d*) shows the initial cells after the elimination of unnecessary ones. Then, with the given tolerance, recursive octree-based subdivision was performed. The non-uniform 3D cells as the result of subdivision are shown in Fig. 12(*e*). After subdivision, points were extracted from each cell and are shown in Fig. 12(*f*).

4.2 A Phone Model

The CAD model of a phone (Fig. 13), was also used to evaluate the performance of the 3D grid method. In this case, the point data is simulated by converting a surface model into an STL model with a small tolerance. The nodes of the STL model are regarded as measured points and no noise was added to the point data, so they can be considered as point data that already had noise filtering.

Figure 14 shows the phone model data for different methods using a data reduction of 90%. The phone model shown in Fig. 14(*a*) by the 3D grid method shows more points distributed at the edges compared to those shown in Figs 14(*b*) and 14(*c*), corresponding to the uniform and space sampling methods, respectively. For the point data reduced by chordal deviation sampling, the edges were preserved fairly well, as shown in Fig. 14(*d*), however, the remaining point data does not perform well for later surface fitting. When using the same number of points, the phone model generated by the 3D grid method maintains better detail at the edges.

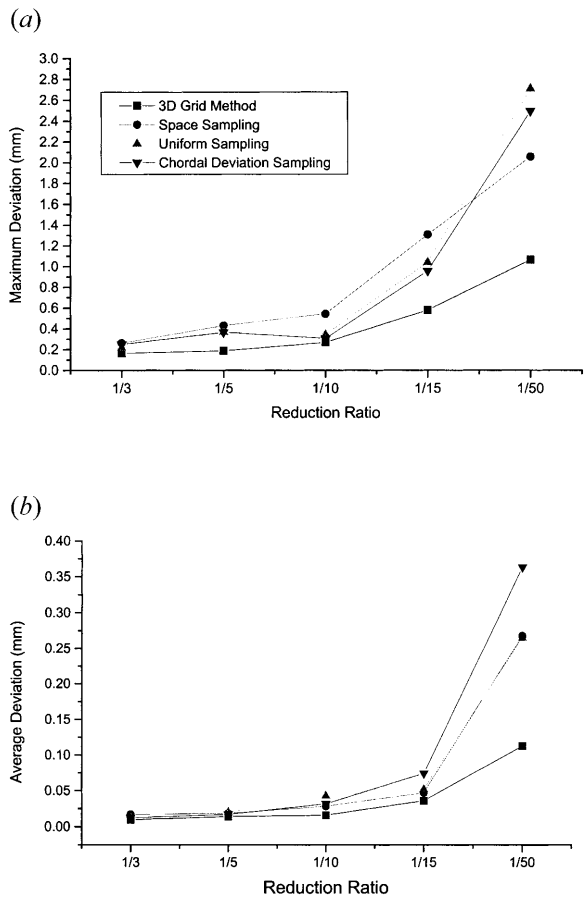

Fig. 16. Error analysis of the 3D grid method. (*a*) Maximum deviation. (*b*) Average deviation.

The phone model is symmetric with respect to both *x*- and *y*-axes, therefore, a quarter of the full model, shown in Fig. 15(*a*), was used to compare the performances between data reduction methods. This quarter section consists of five surface patches, as shown in Fig. 15(*b*), and they are used as reference surfaces for comparison. The results of the 3D grid method were compared with those from three conventional sampling methods: uniform sampling, space sampling, and chordal deviation sampling. The data reduction ratio for each method varied from 1/3 up to 1/50.

To compare the errors between the point clouds generated by the different reduction methods, each point cloud was segmented, as shown by Fig. 15(*c*), based on the reference surface patches from the CAD model. Then, each segmented point cloud was fitted to a surface using the "fit-free-form" command in Surfacer 9.0. The difference between the reference surface patch and the surface fitted by the point data for each reduction method is summarised in Table 1. Figure 16(*a*) shows the maximum error of each method, whereas Fig. 16(*b*) shows the average error. These graphs show that the point data reduced using the 3D grid method maintain a better accuracy than those reduced by the other methods. As the reduction ratio increases, error increases greatly in the other methods.

5. Conclusions

A point data reduction method using 3D grids is proposed in this paper. The method extracts points necessary for representing a part geometry by eliminating redundant or unnecessary points. When compared to conventional sampling methods, such as uniform, space, and chordal deviation sampling, the proposed method shows better performance in terms of accuracy when considering the same number of points.

Since 3D grids are generated while reducing point data, they can be used not only for data reduction, but also for other applications. Triangulation can be performed by the marching cube algorithm using 3D grids, because the cells of the 3D grids are nearly cubes. The resulting cells are similar to voxels in terms of their shapes and, therefore, they can also be used for volumetric representations. The cross-sectional slice data that is required for fabricating rapid prototyping (RP) parts can also be generated efficiently from the 3D grid based model. For example, if a part shape changes quickly at a certain height of the part, many small-sized cells will be distributed in that area. By setting the slice thickness to be the same as the grid size, the fine features of a part can be fabricated. It is assumed that the RP machine supports adaptive slicing in this case.

For further research, in the normal estimation algorithm, the lengths of edges should be considered in determining the normal values as the areas of triangles are considered in Gaussian and mean curvatures. Though noise filtering was beyond the scope of this research, effective noise filtering is essential for the quality of point data since the 3D grid method is noise-sensitive.

References

1. Y. H. Chen, C. T. Ng and Y. Z. Wang, "Data reduction in integrated reverse engineering and rapid prototyping", International Journal of Computer Integrated Manufacturing, 12(2), pp. 97–103, 1999.
2. B. Hamann, "A data reduction scheme for triangulation surfaces", Computer Aided Geometric Design, 11(2), pp. 197–214, 1994.
3. P. Véron and J. C. Léon, "Static polyhedron simplification using error measurements", Computer-Aided Design, 29(4), pp. 287–298, 1997.
4. W. J. Schroeder, J. A. Zarge and W. E. Lorensen, "Decimation of triangle meshes", Proceedings, Computer Graphics (SIGGRAPH '92), 26, pp. 65–70, July 1992.
5. H. Hoppe, T. DeRose, T. Duchamp, J. McDonald and W. Stuetzle, "Mesh optimization", Proceedings, ACM SIGGRAPH, pp. 19–26, 1993.
6. A. Fischer and S. Park, "Reverse engineering: multilevel-of-detail models for design and manufacturing", International Journal of Advanced Manufacturing Technology, 15, pp. 566–572, 1999.
7. S. J. Kim, W. K. Jeong and C. H. Kim, "LOD generation with discrete curvature error metric", Proceedings, Korean Israel Bi-National Conference on Geometrical Modeling and Computer Graphics in the World Wide Web Era, pp. 97–104. September 1999.
8. D. Cohen-Or, D. Levin and O. Remez, "Progressive compression of arbitrary triangular meshes", Proceedings, Korean Israel Bi-National Conference on Geometrical Modeling and Computer Graphics in the World Wide Web Era, pp. 83–96, September 1999.
9. Surfacer User's Guide, Imageware Inc., 1997.
10. CATIA User's Guide, CATIA. Cloud to Geometry (CGO), IBM Corp., 1998.
11. R. R. Martin, I. A. Stroud and A. D. Marshall, "Data reduction for reverse engineering", RECCAD, Deliverable Document 1 COPERUNICUS project, no. 1068, Computer and Automation Institute of Hungarian Academy of Science, January 1996.
12. K. Lee, H. Woo and T. Suk, "Accurate part shape acquisition for reverse engineering", Proceedings, the 25th International Conference on Computers and Industrial Engineering, pp. 52–55, March 1999.
13. T. Suk, "Data reduction methods for reverse engineering", Master's thesis, Kwangju Institute of Science and Technology, February 2000.
14. H. Yau, C. Chen and R. G. Wilhelm, "Registration and integration of multiple laser scanned data for reverse engineering of complex 3D models", International Journal of Production Research, 38(2), pp. 269–285, 2000.
15. T. Fang and L. Piegl, "Delaunay triangulation using a uniform grid", IEEE Computer Graphics and Applications, 13(4), pp. 36–47, 1993.
16. W. Schroeder, K. Marin and B. Lorensen, The Visualization Toolkit: An Object-Oriented Approach to 3D Graphics, Prentice Hall, 1997.
17. M. Sonka, V. Hlavac and R. Boyle, Image Processing, Analysis, and Machine Vision, PWS, 1999.