

**RESEARCH PAPER**

Available Online at [www.jgrcs.info](http://www.jgrcs.info)

## THREE DIMENSIONAL DATA REDUCTION ALGORITHM BASED ON SURFACE FITTING

Hussein Abo-surrah, Saidghoniemy, E.A.Zanaty, Ashraf Afifi

College of Computers and Information Technology

Taif University, Taif, Saudi Arabia.

E-mail: [n.allam@tu.edu.sa](mailto:n.allam@tu.edu.sa)

**Abstract:** This paper proposes an approach for 3D data reduction based on estimating the surface normal vectors for handling the amount of data acquired by laser scanning. The data points are partitioned into cells based on their x, y, and z-axis positions. For normal vector computation, we fit the point data of each cell to implicit general quadric. Then, the shortest distance is directly estimated by intersecting the implicit surface with a line passing through the given point according to the estimated orthogonal orientation, which is necessary for normal vectors computation. The points in each cell are assigned their corresponding estimated normal vectors. For each cell, the points are reduced according to the normal vectors direction. The median point is chosen as representative point of the cell if the assigned normals point to the same direction, otherwise the average point is selected. The performance of the proposed method is illustrated using a range of point clouds scanned from typical engineering surfaces. Keyword: Normal vector, surface fitting, 3D data reduction

### 1. INTRODUCTION

Reverse engineering refers to the process of creating engineering design data from existing physical parts by acquiring its surface data using a scanning or measurement device. Improved data acquisition methods, especially using laser scanning, now make it possible to process scattered point clouds in three dimensional spaces with high accuracy. It has now become a realistic expectation to generate exact and continuous models, which can be directly transferred to, and utilized by CAD/CAM systems. The process of reverse engineering can be divided into four phases: data acquisition, data preprocessing, segmentation and surface fitting [1-5].

This paper focuses on the data preprocessing phase and in particular in reducing the amount of data for different activities: reverse engineering [1], visualization [2] and object recognition [3]. We present a 3D data reduction technique for managing the amount of data acquired by laser scanning. We used a binary search data structure to subdivide the point data into cells. The normal vectors for each cell are estimated by fitting the point data to implicit general quadric based on finding the set of parameters that minimize some distance measures between the given set of points and the fitted surface. For each cell, the normals are assigned to points. The points at each cell are reduced according to normal vectors, i.e. we compute the angle between normal vector and arbitrary vector. The median point and the points with smallest angles than a prescribed value are chosen as the representative points of the cell.

The rest of the paper is organized as follows. Section 2 describes the related work. The proposed technique is presented in section 3. Section 4 gives experimental results. Our conclusion is presented in section 5.

### 2. RELATED WORK

After data acquisition, pre-processing of the data set is required for noise filtering, establishing connectivity between adjacent points, reducing redundancy and merging point clouds taken from multiple views. A major problem in this phase is that certain types of scanner produce vast amounts of data, the processing of which presents serious problem. Rather than processing all of these data at every stage of reconstruction process, an alternative is to use a strategy in which the data is initially reduced, then a model is constructed from this restricted set of data, The full set of data is only used to improve this preliminary model (if necessary). The challenge is to maintain sufficient information from which to calculate surface reliably without distorting those surfaces or their boundaries [6-7].

Recently, some researchers have proposed several techniques to manage the amount of points acquired by laser scanners [7-20]. These techniques are grouped into two categories, the discrete space such as Fourier or wavelets transform, and real space image.

The first category techniques include the discrete Fourier transform (DFT) introduced in [7] and extended in [6] and the discrete wavelet transform (DWT) introduced in [9]. Those techniques are used to reduce the redundant points in one dimension or two dimensions. Recently, Gongde [10] introduced a modification of the real discrete Fourier Transform and its inverse transform to filter the noise and to perform reduction on the data whilst preserving the trend of global moving of time series. The idea of Fourier Transform technique is to reduce the massive amount of data provided by the input device, or arising as a result of transmission of the image. However, the discrete wavelet transform of a signal is calculated and the resultant wavelet coefficients are passed through a threshold testing. In this case, the coefficients that are smaller than a certain value are removed. Then the resultant coefficients are used to reconstruct the signal [9]. It is not the objective of this paper to work in this direction, the techniques presented in this category are practically difficult to be manipulated with very dense 3D data.

The simplest approach of the second category techniques is the neighbourhood averaging in which a point is replaced by the average value of the points contained in some neighbourhood when a point is close to the average value, otherwise, we keep it unchanged. Thus, if we are near to some sort of edge, there will be a large change in points, and points on both sides of it will not be close to the average value [11]. In Martin technique [12], a neighbourhood around the point under consideration is used, but this time the point value is replaced by the median point in the neighbourhood. Their method used a uniform grid with a median filtering approach [12], which has been widely used in image processing. Their method, however, has drawbacks due to the use of the uniform size grids that can be insensitive in capturing a part shape. However, the weakness of this method is that deleting points from every cell without constraint, can distort the surface, especially when the cell belongs to the edge. Also, replacing one cell by one point may distort the surface or its boundary.

There are data reduction methods for reducing the number of polygons in a polygon model, especially for models with triangular patches. The data points for these models were not generated by laser scanners, but they were created for other purposes such as rendering and analysis. The authors [13-16] have also presented a method of data reduction for triangulation files based on an iterative triangle removal principle. As a measure of the reduction of file size, each triangulation is weighted according to the principal curvature estimates at its vertices and interior angles. The weakness of this method is that it takes a long time to generate triangulation and estimate its curvature.

Filter-based methods that try to constrain the factor ratio of scanned data, have been presented for subsampling the data, including randomized sampling, uniform sampling, normal-space sampling and covariance sampling [17-18]. Randomized sampling selects points at random, uniform sampling draws equally distributed samples from the input point cloud. Normal space sampling, as proposed by Rusinkiewicz and Levoy, aims at constraining translational sliding of input meshes, generated from the point cloud [17]. Their algorithm tries to ensure that the normals of the selected points uniformly populate the sphere of directions. Covariance sampling is proposed by Gelfand et al. [18] and extends the normal space approach [19-20]. They identify whether a pair of meshes will be unstable in the iterative closest point (ICP) algorithms by estimating a covariance matrix from a sparse uniform sampling of the input.

### 3. THE PROPOSED METHOD

In this section, we present an overview of the proposed method. This method consists of four steps. In the first step, data points are subdivided into cells based on their positions with respect to x, y, and z-axes, respectively. The second step is to fit the point data to implicit general quadric. In the third step, the surface normal is estimated. The procedure in the fourth step is used to reduce the data in each cell, in the normal vectors assigned to the points. For each cell, an arbitrary direction is selected and referred to as a reference direction. The angles between the reference direction and the normal vectors are estimated. According to the values of these angles, a median point is chosen as representative point of the cell, otherwise the average point is selected. Here, a non-uniform grid method is proposed in which the size of grids can be varied based on the local shape of the component. The size of a grid is defined automatically corresponding to a user defined threshold and it depends on the intended data reduction ratio for the given part shape. In the following, we shall describe these steps in more details.

### 4. POINT DATA PARTITIONING

Let the set of points which we want to handle be put in an arbitrary array  $P$  with x, y and z directions. We used a binary search data structure to subdivide the point data as in Figure 1. The point data is subdivided into two cells, each cell can be subdivided into two cells if needed. We refer to the partitioning stage as levels, so the set of points is partitioned based on  $\ell$  levels,  $level_0, level_1, \dots, level_\ell$ . Partitioning these into  $2^0, 2^1, \dots, 2^\ell$  denotes the corresponding cells' number respectively. This procedure is summarized by the algorithm 1.

#### **Algorithm 1:**

- 1-  $k=1$
- 2- Sort the points in  $P$  corresponding to x-direction in an array

- 3- Partition the points in  $P$  into cells and put them in the sub-arrays  $A_k, k=1, \dots, 2^k$
- 4- Sort the points in  $A_k, k=1, \dots, 2^k$  corresponding to  $y$ -direction
- 5- Partition the points in  $A_k$  into cells and put them into the sub-arrays  $B_k, k=1, \dots, 2^{k+1}$
- 6- Sort the points in  $B_k, k=1, \dots, 2^{k+1}$  corresponding to  $z$ -direction in an array
- 7- Put  $c_k, k = 1, \dots, 2^{k+2}$  to equal 0
- 8- Partition the points  $B_k, k=1, \dots, 2^{k+1}$  into cells and put these it into sub-arrays  $C_k, k=1, \dots, 2^{k+2}$
- 9- Put  $A_k, k = 1, \dots, 2^{k+2}, B_k, k = 1, \dots, 2^{k+2}$  to equal 0,  $k=k+3$
- 10- Put  $P = C_k, k = 1, \dots, 2^{k+2}$
- 11- Stop if the number of points in one array of  $C_k, k=1, \dots, 2^{k+2}$  is lesser than or equal 16 points
- 12- End

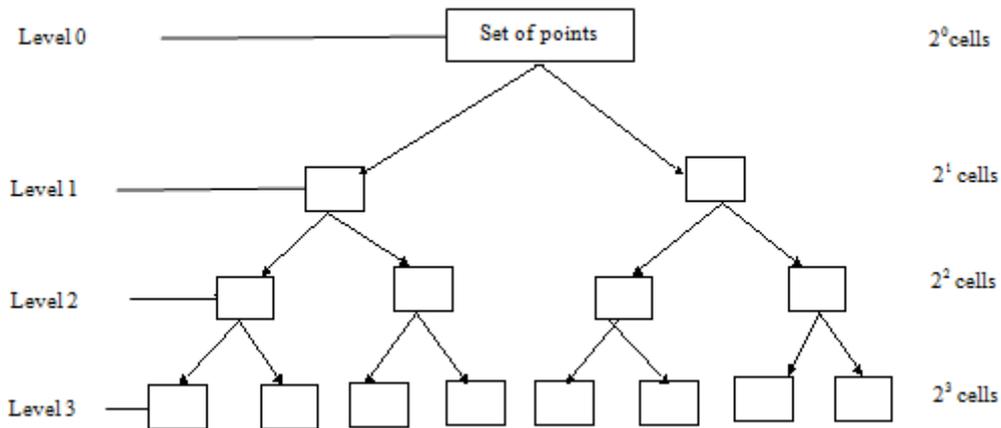


Figure1: The point data is subdivided into  $2^3$  cells.

### 5. SURFACE FITTING

The method begins with a set of 3D points for each cell. It is assumed that the data has been pre-processed to remove gross outliers [21]. A variety of surface forms had been proposed including planar [22], quadratic or cubic [23-24] and parametric quadratic surfaces [21]. To treating planar [22] and quadratic or cubic [23] cases individually, the explicit general quadratic surface is proposed [25]. The method is stable and easy to implement, however it has limited geometric shape description. The general implicit quadratic surface, is represented by:

$$F(b, x) = b_1x^2 + b_2y^2 + b_3z^2 + 2b_4xy + 2b_5xz + 2b_6yz + 2b_7x + 2b_8y + 2b_9z + 1 = 0$$

The fitting process can be performed by the well-known least squares method as follows:

$$F(x, y, z) = \sum_{i=1}^N (b_1x_i^2 + b_2y_i^2 + b_3z_i^2 + 2b_4x_iy_i + 2b_5x_iz_i + 2b_6y_iz_i + 2b_7x_i + 2b_8y_i + 2b_9z_i + 1)^2 = 0$$

For estimating the coefficients from  $b_1$  to  $b_9$ , we differntate the function  $F(x,y,z)$ :

$$\frac{\partial F}{\partial b_1} = 0, \quad \frac{\partial F}{\partial b_2} = 0, \quad \frac{\partial F}{\partial b_3} = 0, \quad \frac{\partial F}{\partial b_4} = 0, \quad \frac{\partial F}{\partial b_5} = 0, \quad \frac{\partial F}{\partial b_6} = 0, \quad \frac{\partial F}{\partial b_7} = 0, \quad \frac{\partial F}{\partial b_8} = 0, \quad \frac{\partial F}{\partial b_9} = 0$$

which leads to the linear system:

$$\begin{bmatrix}
 \sum_{i=0}^n x_i^4 & \sum_{i=0}^n x_i^2 y_i^2 & \sum_{i=0}^n x_i^2 z_i^2 & \sum_{i=0}^n 2x_i^3 y_i & \sum_{i=0}^n 2x_i^3 z_i & \sum_{i=0}^n 2x_i^2 y_i z_i & \sum_{i=0}^n 2x_i^3 & \sum_{i=0}^n 2x_i^2 y_i & \sum_{i=0}^n 2x_i^2 z_i \\
 \sum_{i=0}^n x_i^2 y_i^2 & \sum_{i=0}^n y_i^4 & \sum_{i=0}^n y_i^2 z_i^2 & \sum_{i=0}^n 2x_i y_i^3 & \sum_{i=0}^n 2x_i y_i^2 z_i & \sum_{i=0}^n 2y_i^3 z_i & \sum_{i=0}^n 2x_i y_i^2 & \sum_{i=0}^n 2y_i^3 & \sum_{i=0}^n 2y_i^2 z_i \\
 \sum_{i=0}^n x_i^2 z_i^2 & \sum_{i=0}^n y_i^2 z_i^2 & \sum_{i=0}^n z_i^4 & \sum_{i=0}^n 2x_i y_i z_i^2 & \sum_{i=0}^n 2x_i z_i^2 & \sum_{i=0}^n 2y_i z_i^2 & \sum_{i=0}^n 2x_i z_i^2 & \sum_{i=0}^n 2y_i z_i^2 & \sum_{i=0}^n 2z_i^3 \\
 \sum_{i=0}^n x_i^3 y_i & \sum_{i=0}^n x_i y_i^3 & \sum_{i=0}^n x_i y_i z_i^2 & \sum_{i=0}^n 2x_i^2 y_i^2 & \sum_{i=0}^n 2x_i^2 y_i z_i & \sum_{i=0}^n 2x_i y_i^2 z_i & \sum_{i=0}^n 2x_i^2 y_i & \sum_{i=0}^n 2x_i y_i^2 & \sum_{i=0}^n 2x_i y_i z_i \\
 \sum_{i=0}^n x_i^3 z_i & \sum_{i=0}^n x_i y_i^2 z_i & \sum_{i=0}^n x_i z_i^3 & \sum_{i=0}^n 2x_i^2 y_i z_i & \sum_{i=0}^n 2x_i^2 z_i^2 & \sum_{i=0}^n 2x_i y_i z_i^2 & \sum_{i=0}^n 2x_i^2 z_i & \sum_{i=0}^n 2x_i y_i z_i & \sum_{i=0}^n 2x_i z_i^2 \\
 \sum_{i=0}^n x_i^2 y_i z_i & \sum_{i=0}^n y_i^3 z_i & \sum_{i=0}^n y_i z_i^3 & \sum_{i=0}^n 2x_i y_i^2 z_i & \sum_{i=0}^n 2x_i y_i z_i^2 & \sum_{i=0}^n 2y_i^2 z_i^2 & \sum_{i=0}^n 2x_i y_i z_i & \sum_{i=0}^n 2y_i^2 z_i & \sum_{i=0}^n 2y_i z_i^2 \\
 \sum_{i=0}^n x_i^3 & \sum_{i=0}^n x_i y_i^2 & \sum_{i=0}^n x_i z_i^2 & \sum_{i=0}^n 2x_i^2 y_i & \sum_{i=0}^n 2x_i^2 z_i & \sum_{i=0}^n 2x_i y_i z_i & \sum_{i=0}^n 2x_i^2 & \sum_{i=0}^n 2x_i y_i & \sum_{i=0}^n 2x_i z_i \\
 \sum_{i=0}^n x_i^2 y_i & \sum_{i=0}^n y_i^3 & \sum_{i=0}^n y_i z_i^2 & \sum_{i=0}^n 2x_i y_i^2 & \sum_{i=0}^n 2x_i y_i z_i & \sum_{i=0}^n 2y_i^2 z_i & \sum_{i=0}^n 2x_i y_i & \sum_{i=0}^n 2y_i^2 & \sum_{i=0}^n 2y_i z_i \\
 \sum_{i=0}^n x_i^2 z_i & \sum_{i=0}^n y_i^2 z_i & \sum_{i=0}^n z_i^3 & \sum_{i=0}^n 2x_i y_i z_i & \sum_{i=0}^n x_i z_i^2 & \sum_{i=0}^n 2y_i z_i^2 & \sum_{i=0}^n 2x_i z_i & \sum_{i=0}^n 2y_i z_i & \sum_{i=0}^n 2z_i^2
 \end{bmatrix}
 \begin{bmatrix}
 b_1 \\
 b_2 \\
 b_3 \\
 b_4 \\
 b_5 \\
 b_6 \\
 b_7 \\
 b_8 \\
 b_9
 \end{bmatrix}
 =
 \begin{bmatrix}
 -\sum_{i=0}^n x_i^2 \\
 -\sum_{i=0}^n y_i^2 \\
 -\sum_{i=0}^n z_i^2 \\
 -\sum_{i=0}^n x_i y_i \\
 -\sum_{i=0}^n 2x_i z_i \\
 -\sum_{i=0}^n y_i z_i \\
 -\sum_{i=0}^n x_i \\
 -\sum_{i=0}^n y_i \\
 -\sum_{i=0}^n z_i
 \end{bmatrix}$$

Solving the linear system with, e.g., matlab we get the values of  $b_1, b_2, b_3, b_4, b_5, b_6, b_7, b_8, b_9$ .

### 6. SURFACE NORMAL ESTIMATION

Estimation of surface normals is a fundamental task in many reverse engineering algorithms. After performing the fittedsurface from points at each cell, we compute the orthogonal distance between a point  $p$  and an implicit general quadric [26]:

$$F(b, x, y, z) = b_1 x^2 + b_2 y^2 + b_3 z^2 + 2b_4 xy + 2b_5 xz + 2b_6 yz + 2b_7 x + 2b_8 y + 2b_9 z + 1 = 0$$

This can be formulated as first finding the closest point  $p_o$  on the surface. This point corresponds to the intersection of

the surface with a line passing through  $p$  and orthogonal to the surface at  $p_o$ ,  $\vec{n}_o \equiv (a_o, b_o, c_o)$ ; thus, the orthogonal

distance can be expressed as  $|p - p_o| = \alpha \vec{n}_o$  for some scalar  $\alpha$ :  $\frac{x_p - x_o}{a_o} = \frac{y_p - y_o}{b_o} = \frac{z_p - z_o}{c_o}$  with

$F(b, x, y, z) = 0$ , fits the data optimally in the sense of least squares. It consists in approximating the surface normal

vector  $\vec{n}_o \equiv (a_o, b_o, c_o)$  at the intersection point  $p_o$ , by means of a vector  $h \equiv (a_h, b_h, c_h)$  parallel to the tetrahedron height segment.

The tetrahedron is defined by points:  $p$ ,  $f(x_p, y, z) = 0$ ,  $f(x, y_p, z) = 0$  and  $f(x, y, z_p) = 0$ . The estimation

of the orthogonal distance could be easily obtained from the tetrahedron height segment. Since the fitted curve is defined by an implicit quadric equation  $f(b, x, y, z) = 0$ . the intersection point can be found by representing the

tetrahedron height by means of a parametric equation  $x = x_p + a_h u$ ,  $y = y_p + b_h u$ ,  $z = z_p + c_h u$  then by replacing parametric equation in the implicit quadric expression and by solving that quadric equation, the two values satisfying the implicit equation are found. The nearest one corresponding to the intersection of the tetrahedron height segment and the implicit quadric curve  $|p - p_o|$  is the orthogonal distance estimation.

Let  $r$ ,  $s$  and  $t$  be the three intersections with quadric surface which create a triangular planer patch.

Vectors  $\vec{rs}$  and  $\vec{rt}$  are contained in the plane and their cross product is orthogonal to the planer patch. In other words, it is parallel to the tetrahedron height segment  $\vec{h} \equiv (a_h, b_h, c_h)$  which is used as an estimation of the surface normal vector  $\vec{n}_o \equiv (a_o, b_o, c_o)$  at the intersection point  $p_o$ .

### 7. 3D POINTS REDUCTION

The points in each cell are assigned their corresponding estimated normal vectors  $\vec{n}_i : i = 1, \dots, \beta_j$ . Since cells are treated independently, edges are preserved whilst the number of points is reduced.

The angle criterion guarantees that the angle between the reference normal vector  $\vec{N}$  and the normal vectors of points  $\vec{n}_i : i = 1, \dots, \beta_j$  in one cell can determine geometric shape properties. If the angles are smaller than a prescribed value  $\lambda$ , then the points have different geometric properties and then the median point and the points with smallest angles than a prescribed value are chosen as representative point of the cell. This can reduce large data sets whilst maintaining the information and accuracy contained in the original data. This will be advantageous for surface reconstruction and hence for follow on activities, especially in the manufacturing process.

The performance of 3D points reduction can be described in algorithm 2.

**Algorithm 2:**

1. Set  $j=1$ .
2. Define  $\vec{N}$  on the arbitrary direction.
3. Select normal vectors  $\vec{n}_i : i = 1, \dots, \beta_j$  in cell  $C_j$ .
4. Compute angles,  $\theta_i$ , between  $\vec{N}$  and  $\vec{n}_i$ .
5. If  $|\theta_i - \theta_m| < \lambda$   $i = 1, \dots, \beta_j$ , where  $\lambda$  is a user defined value and  $\theta_m$  is the function corresponding the median point, the median point is chosen, else the average point is selected.
6. Select corresponding original points.
7. Remove residual points in  $C_j$ .
8.  $j=j+1$ .
9. Repeat step 3 through 7 until all the cells have been processed.
10. Stop

### 8. EXPERIMENTAL RESULTS

To assess the performance of the proposed reduction algorithm, it is applied to simulated and actual scanned data. Before applying the reduction algorithm, the initial point cloud is pre-processed to remove any gross outliers. The thresholds for the reduction algorithm were fixed at  $\lambda=0.1$ . Simulated data sets are considered first, i.e. point clouds taken from analytic planar, spherical and cylindrical components to which a number of randomly distributed noise points are superimposed (1.0% of original data size). The level of noise is generally larger than those typically expected in real data (0.1%) so that the reduction method can be assessed in extreme cases.

The performance of the reduction algorithm is assessed by fitting a surface to the data points pre- and post-reduction using standard least squares (LS) [27]. The expectation here is that if the method is performing well, the resulting LS fit should improve as the data is reduced. The error metric is taken as the average of the orthogonal distances between the fitted surface and the points. The results of various degree reductions and the corresponding mean errors are given in Table 1.

For example, cylinder part A was scanned giving 2676 points with a LS mean error=1.009. The point set was then reduced twice. For reduction degree  $r=1$ , the resulting number of points was 1900, i.e. a reduction ratio of 29%, with a mean error=0.002376. Degree reduction  $r=2$  gave 570 points, reduction ratio 78% and mean error=0.002029. The results for all the test cases (Table 1) exhibit the same improved LS fit. This gives evidence that the proposed method reduces the point set whilst maintaining the integrity of the reduced point cloud.

Figures 2(a)-5(a) show the original data sets and Figures 2(b)-5(b) and 5(c) the resulting reduced data sets for the analytical surfaces in Table 1. In all cases the resulting reduced data sets retain enough data to describe the underlying analytic surface.

Table 1. Scanned analytic surfaces with simulated noise.

Surface	Number of points	Mean Error	Reduction degree( $r$ )	Reduction Ratio(%)
---------	------------------	------------	-------------------------	--------------------

<b>Sphere</b>	553	2.0030	0	48
	287	0.00768	1	
<b>Plane</b>	1654	0.9670	0	51
	805	0.010043	1	
<b>Cylinder A</b>	2676	1.0094	0	29
	1900	0.002376	1	
	570	0.002029	2	
<b>Cylinder B</b>	1854	2.0054	0	59
	753	0.0000928	1	

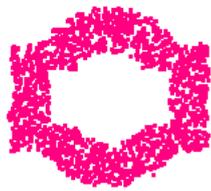


Fig..2a.Planar data n=1654, Mean = 0.967.



Fig..2b.Reduced set n = 805, Mean =0.00097.

Fig..3a.Cylindrical data n =1854, Mean =1.022.

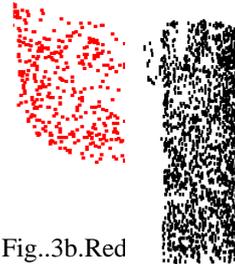
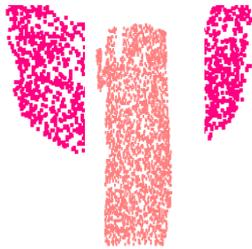


Fig..3b.Red 3, Mean = 0.0087.

Fig..4a.Cylindrical data n = 2676, Mean =1.0094.

Fig.4b.Reduced set n = 1900, Mean =0.00643.



Fig. 4c.Reduced again n = 570, Mean = 0.00048.



Fig.5a. Sphere n = 543, Mean = 2.0054.



Fig.5b. Reduced n = 287,Mean = 0.0034.

As an independent test, two different benchmark data sets: Bajaj and CurvedBox-curve [16] are segmented[25] pre- and post-reduction using the proposed method and the resulting segmentations are visually compared. The original segmentations are shown in figs 6(a) and 7(a). They have 16172 and 27792 points respectively. Applying the reduction algorithm ( $k = 16$ ,  $\epsilon=0.25$ ) gives 12045 points (26% reduction) in the Bajaj set case and 17454 points (38% reduction) in the CurvedBox-curve set. The resulting segmentations are shown in Figures 6(b) and 7(b). Comparing Figures 6(a) and 7(b), and 6(a) and 7(b) indicates that the data reduction has not degraded the information and gives further evidence that the data reduction algorithm is behaving sensibly.

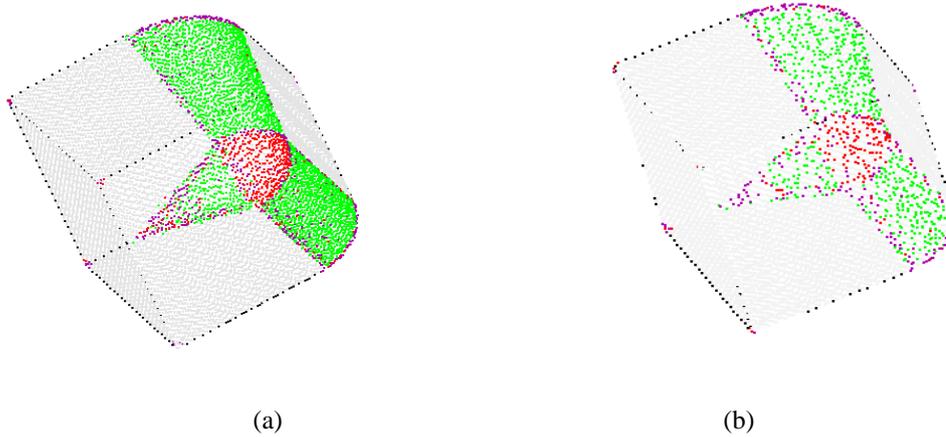
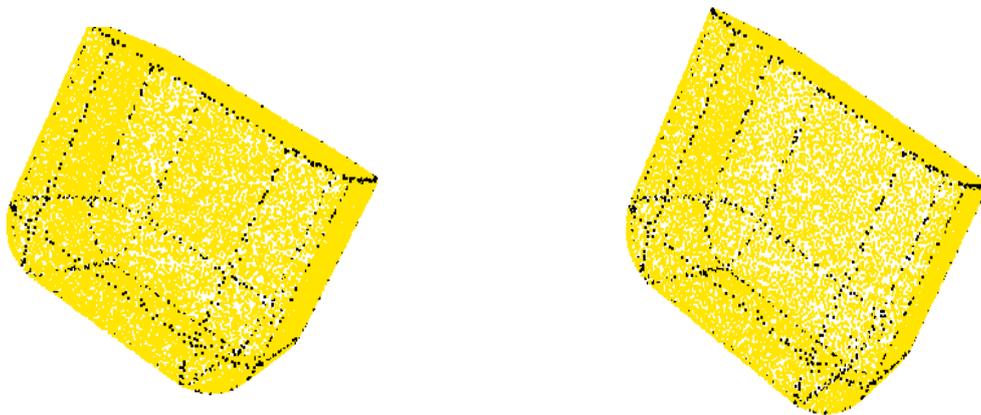


Figure 6: Segmentation of Bajaj data (a) pre- (b) post reduction.



(a)(b)

Figure 7: Segmentation of CurveBox data (a) pre- (b) post- reduction.

## 9. CONCLUSION

The proposed approach is designed to handle data sets of various types, including scanned (laser or otherwise), meshed and triangulated data, of varying density. It can be used to filter data or as a post process to data segmentation or surface reconstruction in reverse engineering. The representative point of a cell is geometrically selected rather than using a

mean or median and is therefore more influenced by the local shape of the point set. The algorithm also differentiates between interior and edge points. Thus the reduction process is constrained so that edges can be better maintained. This is particularly useful when the density of the scanned data is low along boundaries.

Finally the reduction process is governed by threshold values  $\lambda$ .  $\lambda$  controls the amount of data reduction within each cell. Thus the user is able to specify the amount of data to be reduced and control the algorithm to suit specific data sets. For example a large value of  $\lambda$  will suffice for planar data, reducing the computational time for the reduction process. For freeform shapes, the selection of  $\lambda$  is governed by the required level of data reduction. However, if the noise level in the data is high, the thresholds can be reduced, effectively increasing the number of partitions of the data and reducing the number of points and hence the variability within each cell.

The algorithm has been demonstrated to perform as expected, reducing the data whilst maintaining enough shape information to reproduce the original shape. Further experimentation on data sets varying the threshold values has produced equally encouraging results. It is therefore worthy of further consideration. One adaptation would be to extend the algorithm to recognise curvature information which should further improve its shape preserving nature.

## REFERENCES

- [1] X. Li. Geometric property estimation from 3d range data points aided by local quadric surface fitting. In Proc. IEEE Ninth International Conference on Computer Aided Design and Computer Graphics, Hong Kong, Dec. 2005.
- [2] T. Masuda. Filling the signed distance field by fitting local quadrics. In Proc. IEEE International Symposium on 3D Data Processing, Visualization and Transmission, pages 1003–1010, Thessaloniki, Greece, Sep. 2004.
- [3] P. Csakany and A. Wallace. Representation and classification of 3-d objects. IEEE Trans. on Systems, Man, and Cybernetics Part B: Cybernetics, 33(4):638–647, 2003.
- [4] P. Besl and R. Jain. Segmentation through variable-order surface fitting. IEEE Trans. on Pattern Analysis and Machine Intelligence, 10(2):167–192, 1988.
- [5] D. Katsoulas, C. Cea, and D. Kosmopoulos. Superquadric segmentation in range images via fusion of region and boundary information. IEEE Trans. on Pattern Analysis and Machine Intelligence, 30(5):781–795, May 2008.
- [6] R.A. Jarvis, "A perspective on range finding techniques of computer vision", IEEE PAMI, 5(2), 122-139, 1983.
- [7] R. Agrawal, C. Faloutsos, A. Swami, "Efficient similarity search in sequenced databases", In Proceedings of the Fourth International Conference of Foundations of data organization and Algorithms, 1993.
- [8] C. Faloutsos, M. Ranganathan, Y. Manolopoulos, "Fast subsequence matching in time-series databases", In Proceedings of ACM SIGMOD Conf., Minneapolis, 1994.
- [9] C. Kin-pong, and W.F. Ada, "Efficient time series matching by wavelets", Proceedings of the 15th International Conference on Data Engineering, 1999.
- [10] G. Gongde, W. Hui, A.D. Bell, "Data reduction and noise filtering for predicting time series", WAIM, pp. 421-429, 2002.
- [11] A.D. Marshall, R.R. Martin, "Computer vision, models, and inspection," World Scientific Series in Robotics and Automated Systems (1992).
- [12] R.R. Martin, L.A. Stround, A.D. Marshall, "Data reduction for reverse engineering", RECCAD, Deliverable Document 1 COPERUNICUS project, No 10 1068, Computer and Automation Institute of Hungarian Academy of Science, January, 1996.
- [13] M. Fujimoto, Kariya, "An improved method for digitized data reduction for computer integrated manufacturing", Proceedings International Conference on Industrial Electronics, Control Instrumentation, and Automatic, pp. 896-901, 1992.
- [14] C. Daniel, L. David, and R. Offir, "Progressive compression of arbitrary triangular meshes", Proceedings, Korea Israel Bi-National conference on Geometrical Modeling and Computer Graphics in the world wide Web Era, Seoul, Korea, pp. 83-96, September, 1999.
- [15] B. Hamann, "A data reduction scheme for triangulation surface", CAD, 11, pp. 197-214, 1994.
- [16] W.J. Shroeder, J.A. Zarge, W.E. Lorensen, "Decimation of triangle meshes", Computer Graphics, 26(2), pp. 65-69, July 1992.
- [17] S. Rusinkiewicz and M. Levoy, "Efficient variants of the ICP algorithm", Proc. IEEE 3DIM, pages 145 – 152 Canada, 2001.
- [18] N. Gelfand, S. Rusinkiewicz, and M. Levoy, "Geometrically stable sampling for the ICP algorithm", Proc. IEEE 3DIM, Canada, 2003.
- [19] E.A. Zanaty, and R.J. Cripps, "3D Shape preserving data reduction for reverse engineering" submitted to ACM transaction on computer graphics, 2006.
- [20] E. A. Zanaty, Ashraf Afifi, "Reducing Random and Uniform 3D Points for Acquisition System", International Journal of Advanced Research in Computer Science and Software Engineering, Vol.3, No. 11, 2013.

- [21] M. Uhercik, "Implicit surface reconstruction using local approximation from unorganized 3D points", Msc., thesis, ComeniusUniversity, Bratislava, Slovakia, 2005.
- [22] T. Varady., P.Benko, Geza Kos, "Reverse engineering regular objects: simple segmentation and surface fitting procedures", International Journal of Shape Modeling, Vol. 4, No. 3, pp. 127-141, 1998.
- [23] M. Vanco, G. Brunnett, "Direct segmentation of algebraic model for reverse engineering", Computing 72, 1-2, pp. 207-220, 2004.
- [24] F. P.Ferrie, J. Lagardeand, P.Whait,, "Darboux frames, snakes, and super-quadrics: geometry from the bottom up", IEEE Transaction on Pattern Analysis and Machine Intelligence, Vol. 15, No. 8, pp. 771-783, 1993.
- [25] R. J. Cripps, X. Li, "Data Segmentation using implicit surfaces", Proceedings of the 34th International MATADOR Conference, S. Hinduja(Ed.), Manchester, pp.75-80, 2004.
- [26]D. S. Angel and R. Mohammad " Efficient distance estimation for fitting implicit quadric surfaces", IEEE Transaction, pp. 3521-3524, 2009.
- [27] Lukacs G., Marshall A. D. and Martin, R. R., "Faithful least- squares fitting of spheres,cylinders, cones and torifor reliable segmentation", Proc. of the FifthEuropeanConference on Computer Vision, Freiburg, 1998.