Proceedings of the World Congress on Engineering and Computer Science 2008
WCECS 2008, October 22 - 24, 2008, San Francisco, USA

# Piecewise Surface Fitting for 3D Image Extraction.

Mongkol Hunkrajok and Wanrudee Skulpakdee.

***Abstract*— The main aim of this research is to present the method of extracting structure of 3D image datasets captured from a 3D scanner by using a Piecewise Surface Fitting Algorithm that is independent from object structures. Our approach begins by dividing data into small groups and analyzing each group to find an appropriate coordinate system by encoding into tensor, and decomposing it to eigenvectors and eigenvalues. The eigenvector set is used as a transform matrix to transform local data points into a new coordinate system. Then the method has been implemented using approximate mathematical function that describes the data points as a whole. The experimental results are given which shows that the algorithm can produce surfaces of high visual quality and reduce large databases.**

***Index Terms*— 3D image data, Tensor, Transformation, Surface fitting.**

## I. Introduction

For many years 3D image processing has evolved both in hardware and software. Advances in technology, has allowed scientists to accurately capture the high resolution and clarity of 3D images.

To obtain data of a large object using 3D scanning process, we have to scan it in several angles and areas, then combine and build up the obtained data as a 3D image. This primary data are scatter (point cloud data) which contain many unnecessary data points. There are several researchers using various methods and techniques for surface extraction i.e. Marching Cube technique [1], [2] or Tensor Voting technique [3], [4].

Surface Fitting is one among several techniques used to create the surface of an image based on a fairly large number of given data points [5], [6], [7] by using an approximate mathematical function. But to analyze the large databases based on only one function can easily give inaccuracy or error result; therefore, the concept of Piecewise Surface Fitting Algorithm is to divide data into smaller parts and analyze each part before using an approximate function that present local data as a whole. However, some local data may have coordinate system that inappropriate to represent the surface – since the appropriate system has to be one value of domain presenting only one surface point of that local data, thus, we have to transform the system to be correct by definition of function.

Tensor is a generalized linear 'quantity' or 'geometrical entity' that can be expressed as a multi-dimensional array relative to a choice of basis of the particular space on which it is defined. [8] Tensor can be decomposed into 3 orthogonal eigenvectors with 3 eigenvalues describing the array of data in the group [9], [10]. Generally, the direction of eigenvector with smallest eigenvalue is perpendicular to the object's surface. As a result the eigenvector set is used to transform local data points into the new coordinate system appropriated for generating the function, and then use the function to display surface of each selected area.[11]

The remaining of this paper will discuss in details as follows: Section 2 Data Transformation; Section 3 Surface fitting; Section 4 Experimental results and the last Section as conclusion.

## II. Data Transformation

### A. Information capturing from tensor

3D local image features are encoded into Tensor field $F:\Omega \to T_3(\mathbb{R}^3)$ where $\Omega$ is the image domain and $T_3(\mathbb{R}^3)$ denotes the set of symmetric positive semidefinite tensor on $\mathbb{R}^3$

Let $\mathbf{A}$ be a symmetric positive semidefinite $3\times3$ matrix, representing tensor $T_3(\mathbb{R}^3)$. We can decompose such matrix into its orthogonal eigenvectors and eigenvalues

$$\mathbf{A}=\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} = \lambda_1\mathbf{e}_1\mathbf{e}_1^T+\lambda_2\mathbf{e}_2\mathbf{e}_2^T+\lambda_3\mathbf{e}_3\mathbf{e}_3^T \tag{1}$$

where $\lambda_1,\lambda_2$ and $\lambda_3$ are non-negative eigenvalues $(\lambda_1 \ge \lambda_2 \ge \lambda_3)$

$\mathbf{e}_1$ , $\mathbf{e}_2$ and $\mathbf{e}_3$ are orthogonal eigenvectors

We can explain the array of local data as the following:

Let $\boldsymbol{\omega}_p$ is used to be a center for creating tensor matrix. Let $\mathbf{B}_p$ a set of local points surrounding $\boldsymbol{\omega}_p$ which can be written as

Manuscript received June 3, 2008. This work was supported in part by the Chandrakasem Rajabhat University

Mongkol Hunkrajok is with the Department of Mathematics , the Faculty of Science, Chandrakasem Rajabhat University, Bangkok, Thailand 10900. (corresponding author to provide phone: 66-86-793-7949; e-mail: mongkolh@ chandra.ac.th).

Wanrudee Skulpakdee is with with the Department of Mathematics, the Faculty of Science , Chandrakasem Rajabhat University, Bangkok, Thailand 10900. (corresponding author to provide phone: 66-86-793-7949; e-mail: wanrudee@ chandra.ac.th).

$$\mathbf{B}_p = \left\{ \boldsymbol{\omega} \in \Omega \,/\, \left\| \boldsymbol{\omega} - \boldsymbol{\omega}_p \right\| \le r \right\} \quad , \; p = 1, 2, ..., k \qquad (2)$$

where $\Omega$ is a set of whole data
$k$ is number of all partitioning data set
$r$ is radius constant

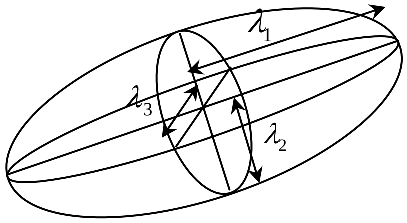



Fig. 1. Graphical representation of symmetric positive semidefinite tensor on $\mathbb{R}^3$.

We consider variation along direction $\mathbf{v}$ among all point in $\mathbf{B}_p$

$$\begin{aligned} \operatorname{var}(\mathbf{v}) &= \sum_{\boldsymbol{\omega} \in \mathbf{B}_p} \left\| \left( \boldsymbol{\omega} - \boldsymbol{\omega}_p \right)^T \cdot \mathbf{v} \right\|^2 \\ &= \sum_{\boldsymbol{\omega} \in \mathbf{B}_p} \mathbf{v}^T \left( \boldsymbol{\omega} - \boldsymbol{\omega}_p \right)\left( \boldsymbol{\omega} - \boldsymbol{\omega}_p \right)^T \mathbf{v} \\ &= \mathbf{v}^T \left[ \sum_{\boldsymbol{\omega} \in \mathbf{B}_p} \left( \boldsymbol{\omega} - \boldsymbol{\omega}_p \right)\left( \boldsymbol{\omega} - \boldsymbol{\omega}_p \right)^T \right] \mathbf{v} \\ &= \mathbf{v}^T \mathbf{A}_p \mathbf{v} \end{aligned}$$

where
$$\mathbf{A}_p = \sum_{\boldsymbol{\omega} \in \mathbf{B}_p} \left( \boldsymbol{\omega} - \boldsymbol{\omega}_p \right)\left( \boldsymbol{\omega} - \boldsymbol{\omega}_p \right)^T \qquad (3)$$

$\mathbf{A}_p$ is a matrix shown data relationship in local area

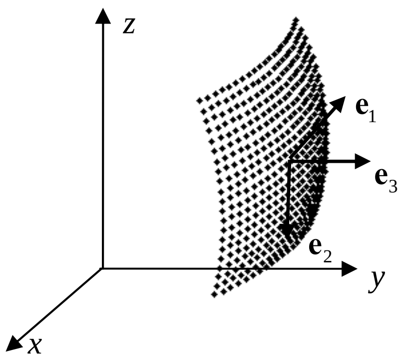



Fig. 2. Eigenvectors of local points in 3 dimensional data.

Figure 2 illustrates eigenvectors with largest eigenvalues capture the most variation $(\mathbf{e}_1)$ and eigenvectors with smallest eigenvalues has the least variation $(\mathbf{e}_3)$ among all of vector $\left( \boldsymbol{\omega} - \boldsymbol{\omega}_p \right)$.

To avoid the effect of norm of vector $\left( \boldsymbol{\omega} - \boldsymbol{\omega}_p \right)$ on eigenvectors' direction, we normalize vectors $\left( \boldsymbol{\omega} - \boldsymbol{\omega}_p \right)$, then (3) can be rewritten as follows:

$$\mathbf{A}_p = \sum_{\boldsymbol{\omega} \in \mathbf{B}_p} \frac{\left( \boldsymbol{\omega} - \boldsymbol{\omega}_p \right)\left( \boldsymbol{\omega} - \boldsymbol{\omega}_p \right)^T}{\left\| \boldsymbol{\omega} - \boldsymbol{\omega}_p \right\|^2} \qquad (4)$$

*B. Rotation of 3D points*

Let $\boldsymbol{\omega} \in \mathbf{B}_p$ are any points in $\mathbf{B}_p$, a set of point in local area surrounding $\boldsymbol{\omega}_p$. We transform the coordinates from the original system to the new system by using eigenvectors decomposed from tensor ; therefore, the vectors in the new system have basis as $\mathbf{e}_1$, $\mathbf{e}_2$ and $\mathbf{e}_3$.

Let $\mathbf{E} = \left[ \mathbf{e}_1 \quad \mathbf{e}_2 \quad \mathbf{e}_3 \right] \in \mathbb{R}^{3\times3}$, since $\mathbf{e}_1$, $\mathbf{e}_2$ and $\mathbf{e}_3$ are orthogonal vectors so:

$$\mathbf{e}_m^T \cdot \mathbf{e}_n = \delta_{mn} = \begin{cases} 1 & for \quad m = n \\ 0 & for \quad m \ne n \end{cases} \quad , \quad \forall m, n \in \{1, 2, 3\}$$

Or it can be written in matrix form as

$$\mathbf{E}^T \cdot \mathbf{E} = \mathbf{E} \cdot \mathbf{E}^T = \mathbf{I}$$

so
$$\mathbf{E}^{-1} = \mathbf{E}^T$$

Let $\mathbf{x}^O = \boldsymbol{\omega} - \boldsymbol{\omega}_p = \left[ x^O \quad y^O \quad z^O \right]^T \in \mathbb{R}^3$ are vectors in vector space which consist of 3 basis, which are $\left[1 \quad 0 \quad 0\right]^T$, $\left[0 \quad 1 \quad 0\right]^T$ and $\left[0 \quad 0 \quad 1\right]^T$ and is called " Original system".

Let $\mathbf{x}^N = \left[ x^N \quad y^N \quad z^N \right]^T \in \mathbb{R}^3$ are vectors in vector space which consist of 3 basis, which are $\mathbf{e}_1$, $\mathbf{e}_2$ and $\mathbf{e}_3$ and is called "New system".

$$\mathbf{x}^O = x^N \mathbf{e}_1 + y^N \mathbf{e}_2 + z^N \mathbf{e}_3$$

can be written in matrix form as:

$$\mathbf{x}^O = \mathbf{E} \cdot \mathbf{x}^N \qquad (5)$$

so
$$\mathbf{x}^N = \mathbf{E}^{-1} \cdot \mathbf{x}^O = \mathbf{E}^T \cdot \mathbf{x}^O \qquad (6)$$

so that data coordinate in the old system can be transformed to the new system as shown in (6) and can be inversed to the original system by using (5).[12]

## III. SURFACE FITTING

Let $x, y$ are independent variables and $z$ is a dependent variable. Let $\mathbf{D}_p$ is a set of points in local area after transformation.

Let $\mathbf{x}_i^N = \left[ x_i^N \quad y_i^N \quad z_i^N \right]^T$ such that $\mathbf{x}_i^N \in \mathbf{D}_p, i = 1, 2, ...., l$ when $l$ is a number of points in $\mathbf{D}_p$

We want to find a function which approximates a set of data points in $\mathbf{D}_p$ by using function $f(x, y)$ which represents the relation between $x, y$ and $z$.

ISBN: 978-988-98671-0-2 

Bivariate surface function can be written in the form

$$f_{NM}(x,y) = S_N(x) \cdot S_M(y) \qquad (7)$$

when $S_N(x)$ is a function of variable $x$ which has $N$ parameters
$S_M(y)$ is a function of variable $y$ which has $M$ parameters

Assume that the function in (7) has $c$ term and $c$ coefficient, we can calculate coefficient by solving the systems of linear equations with $c$ variables and $l$ equations generated by replacing (7) by the component of $\mathbf{D}_p$ 's data.

$$\begin{aligned} f_{NM}\left(x_1^N, y_1^N\right) &= z_1^N \\ f_{NM}\left(x_2^N, y_2^N\right) &= z_2^N \\ &\ldots\ldots\ldots\ldots \\ &\ldots\ldots\ldots\ldots \\ f_{NM}\left(x_l^N, y_l^N\right) &= z_l^N \end{aligned}$$

or can be written in matrix form as $\mathbf{Fc} = \mathbf{z}$
when $\mathbf{F}$ is a constant matrix of the system of equations after replacing $x_i^N$ and $y_i^N$ for all $i = 1, 2, ...., l$
$\mathbf{c}$ is a column matrix representing coefficient
$\mathbf{z}$ is a column matrix representing $z_i^N$
We try to find least squares error solution minimizing the norm of the error vector

$$\mathbf{e} = \mathbf{Fc} - \mathbf{z}$$

Then, our problem is to minimize the objective function

$$J = \frac{1}{2}\|\mathbf{e}\|^2 = \frac{1}{2}\|\mathbf{Fc}-\mathbf{z}\|^2 = \frac{1}{2}[\mathbf{Fc}-\mathbf{z}]^T[\mathbf{Fc}-\mathbf{z}] \qquad (8)$$

whose solution can be obtained by setting the derivative of (8) with respect to $\mathbf{c}$ to zero

$$\frac{\partial J}{\partial \mathbf{c}} = \mathbf{F}^T[\mathbf{Fc}-\mathbf{z}] = 0$$
$$\mathbf{c} = \left[\mathbf{F}^T\mathbf{F}\right]^{-1}\mathbf{F}^T\mathbf{z} \qquad (9)$$

Note that the matrix **F** having the number of row greater than the number of column $(l > c)$ does not have its left psudo inverse $\left[\mathbf{F}^T\mathbf{F}\right]^{-1}\mathbf{F}^T$ as long as $\mathbf{F}$ is not rank deficient. [13]

## IV. Experimental Result

In this research, we applied the method on the sample data of pagodas in Thailand. Figure 5(a) illustrates pagodas code Pa.01T and Pa.02T with different types of surfaces from smooth, curved to rough areas.

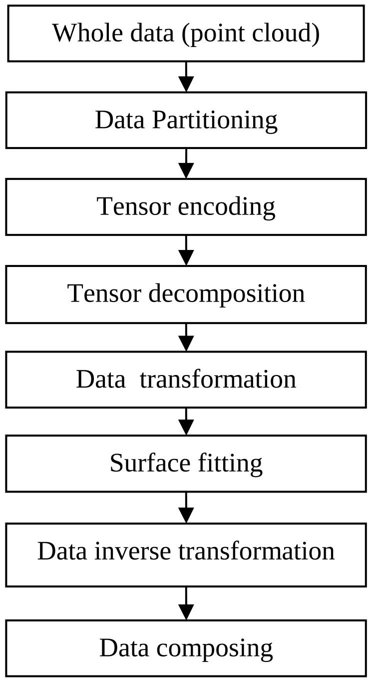



Fig. 3. Piecewise Surface Fitting Algorithm

The sample data have been implemented through the process in Figure3 we begin with dividing data into smaller parts as in (2). Figure 4(a) demonstrates partitioning data of Pa.01T, and then the data will be transformed into a new coordinate system as shown in Figure4 (b). We use the data of the system to generate a function for approximating surface, and we get a structure as shown in Figure 4(c). After that inverse it to the original system as has shown in Figure 4 (d). The last step, we merge all extracted data together and illustrate as a whole image as in Figure 5(b) and 5(c).

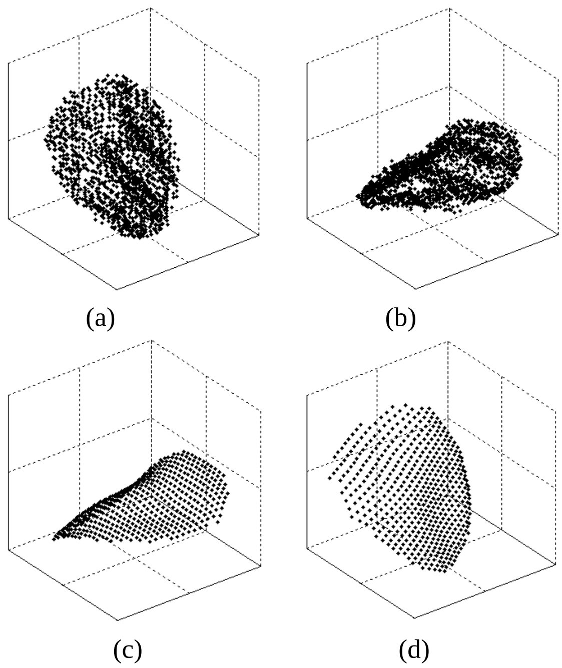


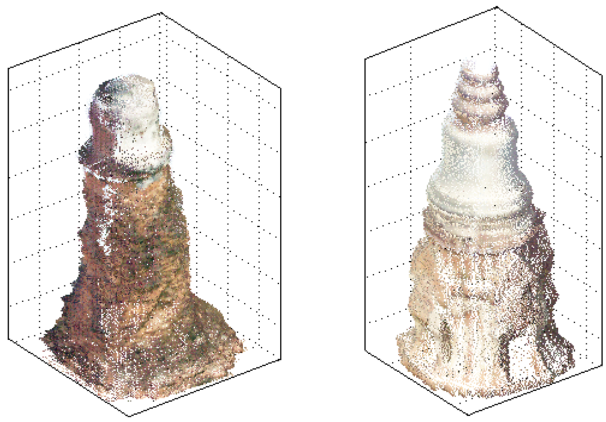

Fig. 4. (a) Point cloud partitioned data
(b) Transformed data into new space
(c) Extracted surface data
(d) Transformed back into real space

ISBN: 978-988-98671-0-2

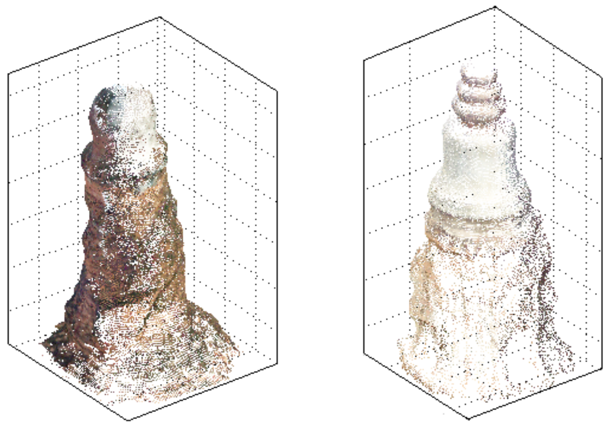
The experimental results are shown in Table 1.

Table 1 : Tested data details and PSF algorithm Outputs.

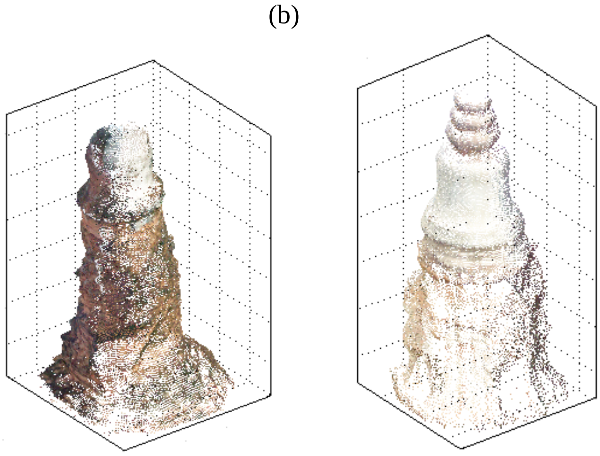
| SAMPLE SET | PA.01T | PA.02T |
|---|---|---|
| No. of original data points | 187,592 | 54,537 |
| No. of extracted data points | 59,101 | 17,024 |
| Radius constant | 0.6 | 0.5 |
| Root mean square error | Fitting by Polynomial function degree 7x7 | |
| | 12.04 | 12.86 |
| | Fitting by Fourier function order 12x12 | |
| | 10.40 | 10.86 |



(a)



(b)



(c)

Fig. 5 (a) Point cloud data object
(b) Surface extracted object using polynomial fit
(c) Surface extracted object using fourier fit

## V. CONCLUSION

From the experimental results, we see that Piecewise Surface Fitting Algorithm is an effective method to extract the structure of 3D object from scatter data that contain many unnecessary data points, cause images unclear, and are not suitable for creating 3D models. The algorithm gives an output of new dataset which has structure similar to the old one, but with smaller data points.

From the experiment, we find that the error of algorithm depends upon following factors:

1) Size of study area (ball's radius) should be appropriate with surface's characteristics. To set a small study area, the numbers of data points maybe not enough for calculating an appropriate coefficient of the function.
2) If the study areas have too many noises -- when use the data to calculate eigenvectors and eigenvalues, the vectors maybe not in proper direction for axis rotation.
3) Types of functions using in Surface Fitting, including orders (or degrees) of functions, are appropriate for surface's characteristics or not.

In this research, we experiment mainly based on objects with pagodas' surface; therefore, we would experiment our future work with different types of objects.

## REFERENCES

[1] C.Montani,R.Scateni and R.Scopigno, "Decreasing isosurface complexity via discrete fitting," *Computer Aided Design*.,vol.17,Elsevier,2000,pp.207-232.
[2] F. Allamandri,P. Cignoni,C. Montani and R.Scopigno, "Adaptively adjusting Marching Cubes output to fit a trilinear reconstruction filter," *Visualization in Scientific Computing.*(Proceeding of the Eurographics Workshop in German),1998.
[3] Gerard Medioni, Chi-Keung Tang and Mi-Suen Lee, "Tensor Voting:Theory and Applications,", *in 12eme Congres Francophone AFRIF-AFIA de Reconnaissance des Formes et Intelligence Artificielle (RFIA)*, Feb 2000.
[4] Mi -Suen Lee and Gerard Medioni , "Grouping .,-, $\longrightarrow$ , 0,into Regions , Curves and Junctions," *Computer Vision and Image Understanding* , vol.76 , no.1 , 1999 , pp.54-69.
[5] G.Taubin,F.Cukierman,S.Sullivan,J.Ponce and D.J.Kriegman, "Parameterized Families to polynomials for Bounded Algebraic Curve and Surface fitting," *IEEE Trans. On Pattern Analysis and Machine Intelligence*.,vol.16,Issue3,March 1994,pp.287-303.
[6] Martin Peternall, "Developable surface fitting to point clouds," *Computer Aided Geometric Design.*,vol.21,Issue8,2004,pp.785-803.
[7] B.I.Loch,J.A.Belward and J.S.Hanan, "Application of surface fitting techniques for the representation of leaf surfaces," *Modeling and Simulation Society of Australia and New Zealand.*,2005.
[8] Wikipedia the free encyclopedia(2007, Nov.)., Tensor, Available: http://en.wikipedia.org/wiki/Tensor
[9] Gerard Medioni , Chi-Keung Tang and Mi -Suen Lee , *A Computer Framework for Segmentation and Grouping* , Elsevier , 2000.
[10] M.Hunkrajok and W.Skulpakdee, "Noisy data reduction using tensor and fuzzy c-means algorithm," *ISCGAV'07*(Proceeding of the7th WSEAS International Conference),2007,pp.50-53.
[11] The Numerical Algorithms Group Ltd, Oxford UK. (2007, Nov.). Curve and Surface Fitting, Available: http://hlrwm.rhrk.uni-kl.de/home/super/nagc/cl/html/E02/e02conts.html
[12] Yi Ma,S.Soatto,J.Kosecka and S.S.Sastry, *An Invitation to 3D Vision.* NEW YORK:Springer-Verlag,2004.
[13] Won Y.Yang,W.Cao,Tae S.Chung and J.Morris , *Applied Numerical Methods Using MATLAB* , New Jersey:A John Wiley and Sons,2005.