IOSR Journal of Mechanical and Civil Engineering (IOSR-JMCE)
ISSN (e): 2278-1684, *ISSN (p):* 2320–334X, *PP: 51-55*
www.iosrjournals.org

Fitting Spline Curves through Set of Unorganized Point Cloud Data

V. N. Chougule, R. N.Nirgude, K. D.Ghag, I.R.Madane, A.S.Deshpande
(Department of Mechanical Engineering, M. E. S. College of Engineering, Pune, India)

ABSTRACT: *In recent days, for fast product development, Reverse Engineering (RE) and Rapid Prototyping (RP) are two widely adopted techniques both by researchers and industries. Due to recent advancements in Computing technology, various techniques like 3D LASER scanner, Co-ordinate Measuring Machine (CMM), Industrial Radiography, etc. can be used to capture data from physical components in form of Point Cloud Data. This Point Cloud Data in turn, can be used for construction of 3D CAD Model by fitting surfaces between points or by constructing curves between points and then fitting surfaces between curves. Prior to solid, it is important to construct a curve model which encompasses all intricate profiles. In current research work, a novel and simplified process of curve generation from unorganised point cloud data is presented. An efficient algorithm using MATLAB to organise the Point Cloud Data acquired from the source images and to fit a B-spline curves satisfying the curve continuity conditions, an efficiently detecting cavities and an abnormilities.*

Keywords – *Computer Aided Graphics Design (CAGD), Point Cloud Data, Nearest Neighboring Point Method, Curve Fitting, B-spline curve.*

I. INTRODUCTION

For rapid product development two emerging technologies, Reverse Engineering (RE) and Rapid Prototyping (RP) have received extensive attention from both research and industries. Reverse Engineering is an important method for reconstructing the Computer Aided Design (CAD) model from a physical part which is in existence. The process starts from digitizing the existing part i.e. capturing a point cloud data of the part by Coordinate Measuring Machine (CMM), 3D LASER Scanners, etc. The measured points are then transformed into a compact CAD model using fitting or interpolation techniques in recent years. The current research of 3D surface reconstruction and software package in reverse engineering do not provide fully automatic reconstruction of complex surface model from the given point cloud data. The process is usually not automatic and requires frequent manual interaction with the user even when well developed software is used.

There are various methods that can be applied to reconstruct a 3D CAD solid model from Point Cloud Data. The comparison and analysis of these methods is most commonly done with CAD model constructed using Curve and surface functions of commercial CAD software reconstruction. CAD software is used for Reverse engineering to obtain a layer-by-layer construction of curve data which is the most herculean task. This paper presents a robust and efficient process to construct B-spline curves through point cloud data points acquired from source images which is vital for 3D solid model construction [1].

II. LITERATURE SURVEY

All existing CAD/CAM systems provide users with curve entities, which can be classified into Analytic and Synthetic entities. Analytic entities are points, lines, arcs, circles, fillets, chamfers and conics. Synthetic entities include Cubic Spline, B-Spline and Bezier Curves. Mathematically, Synthetic curves represent problems of constructing smooth curves that passes through given data points. Therefore the typical form of these curves is a polynomial. Bezier and B-Spline curves can approximate or interpolate the data points. [2] Table 1 shows comparison between different types of curves [1, 2]. The obtained set of unorganized data points in the plane is used to compute a planar B-Spline curve for interpolating the data points. The data point is assumed to represent the shape of some unknown planar curve which can be open or closed. This curve is called a *target curve* or *target shape.*

To achieve automatic curve generation, organizing of points is of huge significance. LASER scanning gains more and more importance in the last few years. LASER scanning produces large sets of multidimensional point data, which demand for an effective and efficient organization and storage. Because of the large data volume several millions of points with increasing tendency, it is not advisable to store the points as conventional points in a CAD program. In general, the approach to maintain the point cloud data in main memory has several disadvantages:

1. This approach requires a long time for loading the data from secondary storage (like hard disks).
2. Main memory storage shows a bad scalability because it swaps (after exceeding a threshold) parts of the memory onto the slow secondary storage [3].

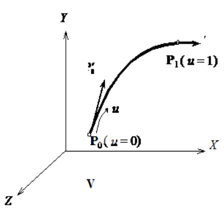
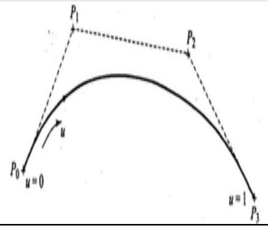
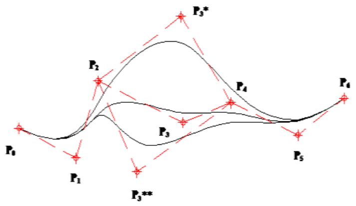
Parameter	Hermite cubic spline curve	Bezier curve	B-spline curve
Definition	It is 3rd degree polynomial with 4 data points and 4 coefficients.	It is curve of nth degree polynomial with n+1 no. of data points.	It is Bezier curve with varying degree.
Formula	$P(u)=\sum_{i=0}^{3} C_i u_i, 0 \le u \le 1$	$P(u)=\sum_{i=0}^{n} P_i B_i, n(u), 0 \le u \le 1$	$P(u)=\sum_{i=0}^{n} P_i N_i, k(u), 0 \le u \le u_{max}$
Advantages	Easy for computation	Curve can pass smoothly through number of data points. Smooth due to high degree continuity.	It has local control over shape of curve. Degree of curve is independent of data points.
Disadvantages	It cannot have control over more than 4 data points. Less smooth than Bezier and B-spline curves	Computation required is more due to higher degree. As it has global control over shape of curve, movement of points will give different shape to the curve.	Computational time increases with complexity of curve.
Figures			

Table 1: Comparison between synthetic curves

III. Nearest Neighboring Point methods [NNP]:

For automatic curve generation, point clouds are arranged in a sequence of nearest point. The most straight forward way to compute the distance between two points is decided by finding the smallest distance between reference points and its neighboring points. The common methods of distance metric are non-parametric which base their retrieval decision directly on the features of the data points. Non-parametric distance metrics method has several important advantages.

1. It enables to naturally handle a huge number of points.
2. It avoids detection of same points repetitively.

Following are some methods used for used to find minimum distance i.e. to find nearest neighboring point.

1. Euclidean Distance: Euclid state that the shortest distance between 2 points on a plane is a straight line and is known as Euclidean distance [4]. If u=(x1,y1) and v=(x2,y2) are two points then Euclidean distance between u and v is given by,

$$EU(u,v) = \sqrt{(x1-x2)^2+(y1-y2)^2} \tag{4}$$

2. Manhattan distance: The distance between points measured along axes at right angles. If u=(x1,y1) and v=(x2,y2) are two points then Manhattan distance between u and v is given by,

$$MH(u,v) = |x1-x2| + |y1-y2| \tag{5}$$

3. Minkowshi distance: This is the generalized metric distance. When t=1 it becomes city block distance and when t=2, it becomes Euclidean distance [4]. Chebyshev distance is special case of Minkowshi distance with t=∞ .The Minkowshi distance of order 't' between two points is defined by P=(x1, x2, x3,…..xn) and Q=(y1,y2,y3,…..yn)

$$(\sum_{i=1}^{n} |xi-yi|^{t})^{1/t} \tag{6}$$

IV. CASE STUDY

The curve reconstruction from noisy point samples is needed for surface reconstruction in medical imaging applications. So, CT images were selected for study purpose. The point cloud data for each of CT image from edge detection techniques was obtained. By using MATLAB software [5, 6, 7], the algorithm was prepared for curve generation and compiled to get appropriate result. Beginning with few simple profiles we came to know that it is important to organize a point cloud data for curve path generation. The curve follows the path by sequence in which data point are arranged. Unorganized data point would give zigzag curves. Setting the data points in a particular fashion is a crucial step to be considered.

Initially the curve fitting was done on sample set of points of circular loop. To fit spline curve, it is important to give input point data in same order in which curve is to be generated .This method below overcome that problem.

Algorithm#1: General algorithm for fitting a B-spline curve P (t) to a point cloud data of sample circular loop.
1. Import Point Cloud Data.
2. Define a suitable start point of the curve.
3. Start an iterative loop, with formula for find nearest point.
4. Get the output point i.e. nearest point to the start point.
5. Again iterate for next nearest point
6. Then, get the output file with point cloud in organized fashion.
7. Then, pass spline curve through arranged points.

This approach was limited only to the single curve generation. But with complex images, curve complexity also increases. That instant, organizing the point cloud data curve-wise became important.

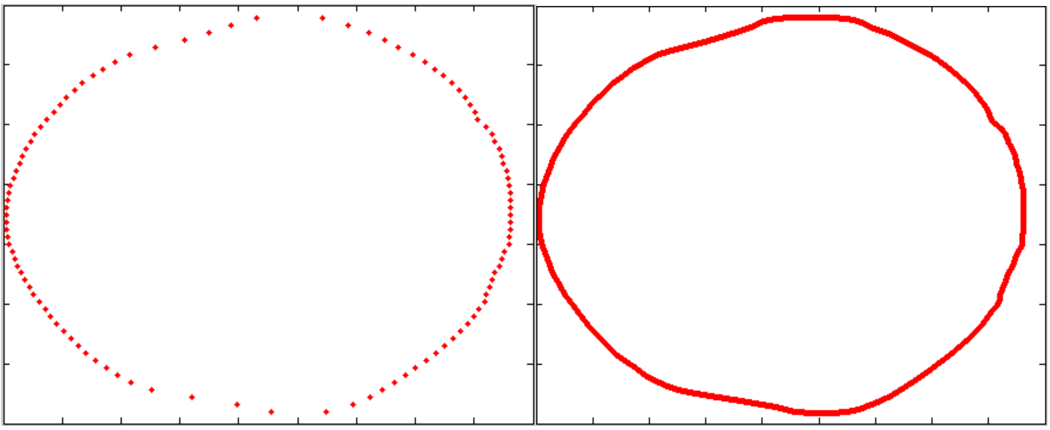


Fig.5 MATLAB output: (a) Set of point cloud data (b) B-spline curve through points

In the second phase little more complex set of data point image which composes point clouds of two or more circular loops namely outer and inner loop were taken. This data was sorted out into outer loop cloud point from set and stored under label of outer points. Remaining points of set stored under label of inner point. After passing curve it showed outer curve as expected but inner curve was not smooth. This indicated that between outer curve and inner curve, there are some point clouds that represents internal cavities. Thus, more generalized method for curve construction for high complex point cloud set was necessary. The following Algorithm was developed using CT images and some image processing operations.

Algorithm#2:
1. *Import CT images*
2. *Conversion into binary gray scale image*
3. *Extraction of edges by suitable edge detection algorithm like Canny, Sobel, etc.*
4. *Identification of edge pixels*
5. *Study the pattern of pixel strength around corner point and store those patterns as templates.*
6. *Identification of corner pixels by template matching*
7. *Label each curve to differentiate it from other curve*

8. *Storing corner point pixel loop wise / curve wise.*
9. *Fitting B-spline curve through each subset of point cloud.*

Binary Image has only 2 gray levels i.e. black and white. A binary image is composed of pixels, and each pixel with a gray level between 0 (black) to 255 (white) corresponds to one point on the image [1]. For binary images, these techniques can be useful in extracting the vertex pixels directly. Each of the interior pixels has adjacent pixels called its neighbors. Intensity pattern of given pixel and its neighbors was examined to determine whether a point is a corner point or vertex [8]. This can be done by scanning over the image with a set of 3x3 corner point templates which represents all possible types of corners in objects.

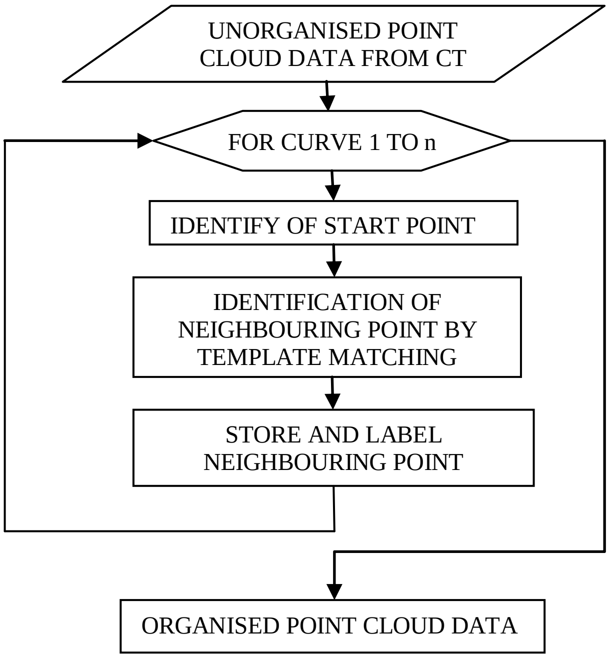


Fig. 2 Flowchart for Algorithm #2

Let us consider binary image S (k, j). S (k, j) is a binary image with 0's representing background pixels and 1's representing foreground pixels and 3 X 3 matrixes are templates. [9,10] To search for corner points, the 512 X 512 matrix obtained from binary image is scanned with templates as shown in fig. Once the templates get matched, consider its interior pixel as start point and store its value (i, m) in new matrix and search for neighboring pixel of value 1. This process continues as long as all the pixel values forming a close loop are obtained. Thus, separate matrixes are formed containing information of different close curves. Thus, point clouds for particular curve in particular matrix. Obtaining an organized cloud point from noisy set of point cloud is achieved.

COLUMN NUMBER

ROW NUMBER	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0
3	0	0	0	1	1	0	0	0	0	0	1	0	0	0	0
4	0	0	1	1	0	0	2	2	2	0	0	1	1	0	0
5	0	0	1	0	0	2	0	0	0	2	0	0	1	0	0
6	0	1	0	0	2	0	0	0	0	0	2	0	0	1	0
7	0	1	0	0	2	0	0	0	0	0	2	0	0	1	0
8	0	1	0	2	0	0	0	0	0	0	0	2	0	1	0
9	0	1	0	0	2	0	0	0	0	0	2	0	0	1	0
10	0	1	0	0	2	0	0	0	0	0	2	0	0	1	0
11	0	0	1	0	0	2	0	0	0	2	0	0	1	0	0
12	0	0	1	1	0	0	2	2	2	0	0	1	1	0	0
13	0	0	0	1	1	0	0	0	0	0	1	1	0	0	0
14	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

CORNER POINT TEMPLATE

0	1	1		1	1	1		1	1	0
0	1	1		0	1	0		1	1	0
0	0	0		0	0	0		0	0	0

0	0	0		0	0	0		0	0	1
0	1	0		0	1	1		0	1	1
1	1	1		0	1	1		0	0	1

NEAREST POINT TEMPLATE

0	0	1		0	0	0		1	0	1
1	1	0		1	1	0		0	1	0
0	0	0		0	0	1		0	0	0

Fig.3 a) Image with Binary Pixel Strength b) Templates with Different Pattern

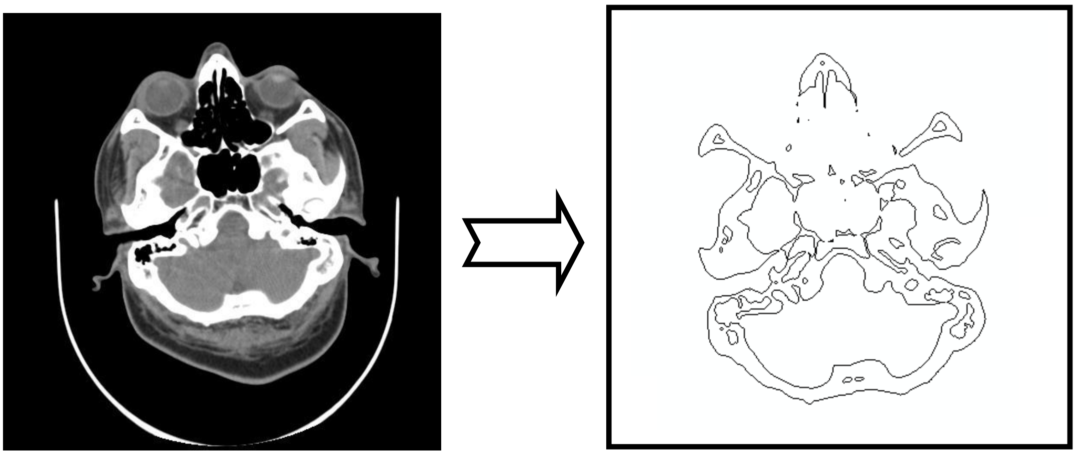


Fig.4 a) Input CT image, b) Output Result

V. Conclusion and Future Work:

We come across the obstacles to the curve fitting problem. We faced two major kinds of obstacles i.e. finding nearest neighboring point and organized form of large set of cloud points curve-wisely. By mentioned algorithm, problems were solved. Result we obtain from this method is very effective and efficient with satisfactory accuracy. This method is prepared for bio-CAD application, we are looking for. Ongoing studies are being undertaken to determine more general method, useful for various application with various approach.

REFERENCES

[1] Chung-Shing Wang, Wie-Hua A. Wang, Man-Ching Lin, *STL Rapid Prototyping Bio-CAD model for CT medical segmentation*, Elsevier journal, 3(61), 2010, 187-197.

[2] Ibrahim Zeid's *Mastering CAD/CAM*, first edition (Tata McGraw-Hill, New Delhi, 2004).

[3] T. Brinkoff, *Spatial Access Methods For Organizing LASER scanner Data*, ISPRS Journal of Photogrammetry & Remote Sensing 54(1), 2004, 64-67.

[4] R.Gonzalez, R.Woods, S. Eddins, *Digital Image Processing Using MATLAB 2nd Edition* (Tata McGraw-Hill Education, New Delhi, 2010)

[5] S.J. Chapman, E. Herniter, *MATLAB Programming for Engineers 2nd Revised edition Edition* (Cl-engineering,2001)

[6] B. R. Hunt, Ronal L. Lipsman ,*A guide to MATLAB,* (Cambridge University Press, New Delhi, 2011)

[7] L. M. Mestetskiy, N. F. Dyshkant, D. Gordev, M. S. Kumari, B.H. Shekar, *Surface measures for accuracy evaluation in 3D face reconstruction*, 10(45), 2012, 3583-3589.

[8] R. Schilling's *Fundamentals of robotics* (Pearson education, New Delhi,2007)

[9] W. Wang, H. Pottmann, Y. Liu, *Fitting B-Spline Curves to Point Clouds by Curvature-Based Squared Distance Minimization*, ACM Transactions on Graphics, *25(2), 2006, 214-238*.