

# **An Automatic Method for Measuring Foot Girths for Custom Footwear Using Local RBF Implicit Surfaces**

Yihua Ding<sup>1</sup>, Jianhui Zhao<sup>1</sup>, Ravindra S. Goonetilleke<sup>2</sup>, Shuping Xiong<sup>3</sup>, Zhiyong Yuan<sup>1</sup>

<sup>1</sup>Computer School, Wuhan University, Wuhan, Hubei, 430079, PR China

<sup>2</sup>Dept. Industrial Engineering and Logistics Management, Hong Kong University of Science and Technology, Hong Kong

<sup>3</sup>Dept. Industrial Engineering and Logistics Management, Shanghai Jiao Tong University, Shanghai, 200240, PR China

## **Abstract:**

Scanned 3D point cloud data of a foot and a last are used in the manufacture of custom footwear. But, automatic and accurate measurement of dimensions, especially girths, is an issue of concern to many designers and footwear developers. Existing methods for measuring girths are primarily based on points or generated triangles, but their accuracy is heavily dependent on the density of the point cloud data. In this paper we present the use of the Radial Basis Function (RBF) surface modeling technique for measuring girths as it has the advantage of being able to operate on unorganized 3D points, allowing the surface that is produced to go through every scanned point, while repairing incomplete meshes. To overcome the high computational expense of the RBF method, local surface recovery, octree division and combination, inverse power method and improved Cholesky factorization are also used. The girth measurements obtained from these approaches are compared against those using purely scan points, and other methods. Experimental results demonstrate that the local RBF implicit surface can provide more stable and accurate measurements, using relatively less time, proving its value in custom footwear manufacture.

**Keywords:** 3D points; RBF; Girth measurement; foot measurement, anthropometry, Custom footwear, scanning, footwear, foot girth, foot model, 3D scan

## 1. Introduction

3D digitalization technologies have paved the way for automatic measurement of critical dimensions from scanned point cloud data. In the shoe making industry, foot and last dimensions are important for the manufacture of custom footwear. Bunch et al. [1] used a 3D digitizer to collect the coordinates of 34 landmarks on the foot surface from which a series of foot dimensions were determined with a computer program. Similarly, Liu et al. [2] used an electromagnetic digitizing device to obtain the coordinates of 26 points on the surface of foot and leg, and then computed 23 variables that included heights, lengths, widths and angles. Many others (e.g. Tsung et al. [3]) have used differing techniques to obtain dimensions such as foot length, foot width, rear foot width, contact area, arch height, and arch angle. Compared with the linear dimensions, girth measures are more difficult to determine as the digitized points are only discrete samples of the continuous real surface. To overcome the discreteness, Xu et al. [4] have constructed triangular meshes from the unorganized point cloud, generated continuous multiple lines, and determined the girth as the sum of all the line lengths. The weakness of the method is that the accuracy is heavily dependent on the density of triangles. Witana et al. [5] used a somewhat different technique projecting the 3D points on the surface of the human foot nearby the plane of measurement onto one 2D plane, and then used the convex hull of the points to determine the girth. This method was aimed at simulating a tape measurement of the foot wherein there are apertures between the tape and the skin surface. The method works on point cloud data but this measurement can be affected by the density of the 3D points as well. Zhao et al. [6] used a Non-Uniform Rational B-Spline (NURBS) to generate smooth surface from point cloud, and then the intersections between the NURBS surface and the tape plane was determined. The length of the intersection curve was taken as the simulated girth measurement. However, the NURBS method has two limitations: control points of NURBS surface should be previously ordered thus it cannot be directly used on unorganized 3D points; the NURBS surface only approximates but not goes through each control point even though the scanned points are on the surface of the foot. Therefore, a more suitable surface modeling approach, without such limitations, should be developed for the determination of girths based on 3D points.

As an important interpolation tool for point cloud data, the implicit representation of objects' shapes with Radial Basis Function (RBF) offers a unified framework for several problems such as surface reconstructing, smoothing and blending, and has attracted a lot of attention. RBF methods are a series of exact interpolation techniques, conceptually similar to fitting a rubber membrane through the measured sample values while minimizing the total curvature of the surface. Carr et al. [7] used thin-plate spline RBF to deal with the problem of interpolating incomplete surfaces derived from 3D medical graphics, and the defects of a skull were repaired. Then they applied polyharmonic RBF to reconstruct smooth, manifold surfaces from point cloud data and to repair incomplete meshes [8]. Ohtake et al. [9] developed an adaptive RBF fitting procedure for high quality approximation of a set of points. The computational time and number of approximation centers depend not only on the size of the dataset but also on the geometric complexity. They proposed a hierarchical approach to 3D scattered data interpolation and approximation with compactly supported RBFs [10]. Their method can integrate the best aspects of scattered data fitting with locally and globally supported basis functions. Dinh et al. [11] presented a method of surface reconstruction that generates smooth and seamless models from sparse, noisy, non-uniform, and low-resolution vision-based data sets. Their method is based on a 3D implicit surface formulated as a sum of weighted RBFs, and the reconstructed surface is locally detailed yet globally smooth because RBFs are used to achieve multiple orders of smoothness. Marie et al. [12] presented an approach for recovering surfaces from scattered 3D points combining generalized RBFs and Voronoi-based surface reconstruction. Since RBF solutions are global in nature, processing millions of points may be beyond the capabilities of most present day personal computers [13]. Therefore, improvements to computational speed and evaluating RBFs are highly desirable especially with large chunks of data.

In this paper, RBF was used to reconstruct a surface from which the girths can be determined. Considering that only partial surface of the object really needs to be recovered for measurement, we propose an approach to generate the RBF based local implicit surface and thus avoid the whole surface recovery, which is a way to reduce the computing time. To further decrease the computation cost, we present several related methods to improve the efficiency of RBF implementation. The efficacy of the RBF method is thereafter compared

with methods used before. Two published methods are introduced in section 2, RBF based reconstruction of local implicit surface is proposed in section 3, approaches for efficiency improvements are presented in section 4, girth simulation using RBF surface is described in section 5, and then the experimental results and analyses are presented in section 6.

## **2. Existing Methods for Girth Measurement**

Girth measurements can be obtained in many ways. Two methods that have been reported are as follows:

- (1) Project all the nearby 3D points corresponding to a measuring tape plane and then use the 2D convex hull of the projected points to simulate girth measurement;
- (2) compute the intersection lines between the tape plane and 3D triangles generated from scanned points, and then use the line segments to simulate girth measurement.

The results of these two methods are compared with the proposed RBF method.

### **2.1 2D convex hull**

Suppose the 3D tape plane,  $C$ , is defined by three control points in 3D space,  $c1(x1, y1, z1)$ ,  $c2(x2, y2, z2)$  and  $c3(x3, y3, z3)$ . The distance between each point of the scanned point cloud and plane  $C$  is computed. The 3D points on the surface of the foot near the measuring tape are projected onto a 2D plane. Then the convex hull of the projected 2D points is obtained and used to simulate the girth measure.

As illustrated in Figure 1, the convex hull of a set of 2D points is first determined: (1) Find the point with minimum  $Y$  value, and take it as the starting point; (2) Test each of the other points, and find the one which makes the largest right-hand turn (with respect to the start point) as the second point on the hull; (3) Repeat to test each of the other points (including the start point) and find the one which makes the largest right-hand turn (with respect to the previous hull point) as the next point on the hull; (4) Stop when the start point is selected; (5) Save the  $T$  points that were found and set them as the boundary points of the convex hull

$p_1, p_2, \dots, p_T$ . The girth can then be calculated as the sum of the Euclidean distances between two neighboring points on the convex hull:

$$girth = \sum_{i=2}^T \|p_i - p_{i-1}\| + \|p_T - p_1\| \quad (1)$$

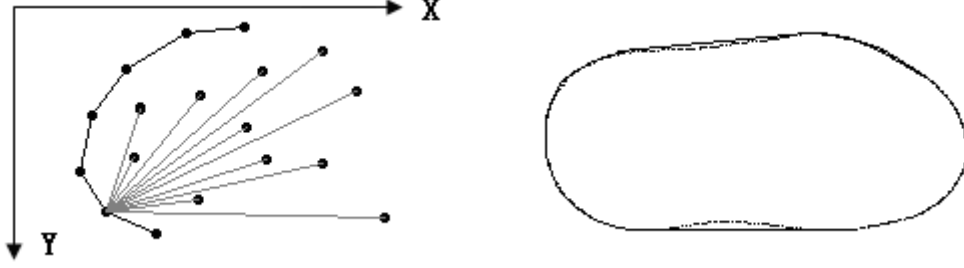


Figure 1. 2D convex hull for projected points

The above method is similar to a manual tape measurement as apertures tend to be present when a tape is stretched over the foot surface as shown in the right of Figure 1. It is clear that the density of the point cloud data is an important determinant of the measure accuracy.

## 2.2 Method with 3D triangular mesh

To generate a triangular mesh from unorganized 3D points, Lin's [14] algorithm can be implemented considering the intrinsic nature of the point cloud. Starting with a seed triangle, the algorithm grows a partially reconstructed triangular mesh by selecting a new point based the degree of sampling uniformity. Thus the recovered mesh is essentially an approximate minimum-weight triangulation to the point cloud constrained to be on a two-dimensional manifold. For any edge of each triangle, suppose its two endpoints are  $M$  and  $N$ , while their distances to tape plane  $C$  are  $M.val$  and  $N.val$ , the intersection point  $p_i$  of the edge and plane  $C$  is calculated as

$$p_i = (N.val * M - M.val * N) / (N.val - M.val) \quad (2)$$

There are four possible cases of intersections between any one triangle and the tape plane, as shown in Figure 2. Only two of them, case 2 and case 4, are useful for girth measurement. After the intersection segments are determined, they are dealt in the following way: (1) Take

one endpoint of a randomly selected segment as point,  $p_1$ , and take the other endpoint of the segment as  $p_2$ ; (2) Check each of the other segments, and find the segment with one endpoint  $p_2$ , then take the other endpoint of the segment as  $p_3$ ; (3) Repeat to test the other segments until the endpoint  $p_1$  is found for the second time; (4) Save the T points found as the boundary points for the girth  $p_1, p_2, \dots, p_T$ .

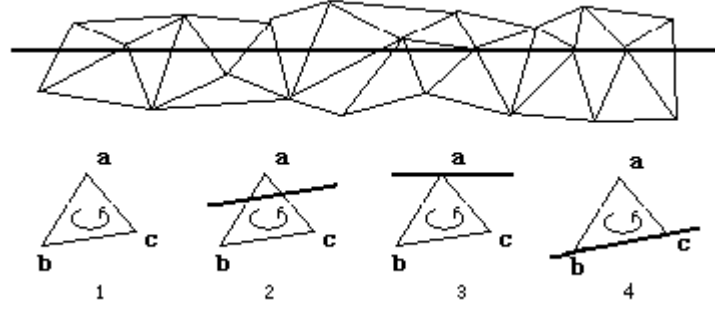


Figure 2. Four cases of intersection

These T points can then be used to calculate the girth value using Equation (1), which simulates the tape measurement with contact along the surface. Or they can be projected onto one 2D plane and then their 2D convex hull can be used to simulate the measurement with the tape stretched around the surface. It is clear that the girth measurement with 3D triangular mesh is also affected by the density of the point cloud data, 3D points or generated triangles.

### 3. RBF Based Local Surface Recovery

Considering that only the local surface of the object is really needed for girth measurement, a constrained partial surface recovery using RBF is proposed. The border or edge of the partial 3D points should be considered for the constrained surface during RBF based reconstruction, somewhat different from the entire surface recovery,. Therefore, the method for constrained surface recovery includes the following steps: transformation of the partial point cloud using PCA, generation of off-surface points, RBF construction for the recovery of points, followed by the output of the local surface using a marching cube method.

#### 3.1 Partial point cloud transformation

When only part of the 3D point cloud needs to be reconstructed for further processing, the edges of these points have to be located to generate the proper shape. For a 3D object with different locations in space, there are differing ranges along X, Y and Z axis respectively in the normal coordinate system. These can result in differing bounding boxes for the 3D object, and thus cause differing precisions in the definition of the border. To define a more precise border for the set of partial point clouds, components in order of significance from the points are computed with the help of the Principal Component Analysis (PCA) method. Suppose there are N points in the partial point cloud, each point is represented by  $p_i(x_i, y_i, z_i)$ ,  $i \in [1, N]$ , and the center  $O(x, y, z)$  is calculated as the average of the N points. Covariance matrix for the partial point cloud is defined as:

$$COV = \frac{1}{N-1} \begin{bmatrix} \sum_{i=1}^N (x_i - x)^2 & \sum_{i=1}^N (x_i - x)(y_i - y) & \sum_{i=1}^N (x_i - x)(z_i - z) \\ \sum_{i=1}^N (y_i - y)(x_i - x) & \sum_{i=1}^N (y_i - y)^2 & \sum_{i=1}^N (y_i - y)(z_i - z) \\ \sum_{i=1}^N (z_i - z)(x_i - x) & \sum_{i=1}^N (z_i - z)(y_i - y) & \sum_{i=1}^N (z_i - z)^2 \end{bmatrix} \quad (3)$$

Eigen values of the covariance matrix are sorted in descending order, and the eigenvector corresponding to the highest eigen value is taken as the principle component of the data set. The ordered eigen vectors are used to set up a local PCA coordinate system with point O as the coordinate origin. Then the PCA coordinate system is transformed to be coincident with the global coordinate system. In this way, the transformed partial point cloud is more significant along both X and Y dimensions while less significant along Z dimension of the global coordinate system.

### 3.2 Generation of off-surface points

Range of the 3D partial point cloud can be defined by its six extreme coordinate values  $x_{\min}, x_{\max}, y_{\min}, y_{\max}, z_{\min}, z_{\max}$ . Size R of the cubic bounding box used to subdivide the partial points is computed from Equation (4). Then the partial point cloud can be divided into a number of bounding boxes. For point  $p_i$ , the bounding box containing it is found first, then the points in the same box and the six boxes that are connected are added together to make a local neighborhood for  $p_i$ .

$$R = \sqrt[3]{\frac{(x_{\max} - x_{\min})(y_{\max} - y_{\min})(z_{\max} - z_{\min})}{N}} \quad (4)$$

Applying Equation (3) on all the points in the local neighborhood of point  $p_i$ , the plane decided by the two eigen vectors corresponding to the two larger eigen values is taken as the fitted surface, and the eigen vector with the smallest eigen value is taken as the surface normal of point  $p_i$ . Since the calculated normals for the partial point cloud may point towards the inside or the outside of the reconstructed surface, an adjustment with the breadth first search method is used to make them consistent. Take the point with maximum  $z$  value as the root point with normal  $n_r$ ; visit each neighbor point of the root with normal  $n_i$ , if  $n_r \cdot n_i < 0$ , change  $n_i$  to its opposite direction; for each visited point, deal with its un-visited neighbor points in the same way until all the points are processed.

For one point,  $p_i$ , on the partial point cloud, its off-surface point is defined as the position moved along its normal,  $n_i$ , or the opposite direction,  $-n_i$ , for a predefined distance  $d$ . Therefore, a signed distance function,  $f$ , is defined as in Equation (5). In this work,  $d$  is defined as  $1/10$  of the minimum edge of the minimum spanning tree of the 3D partial point cloud. Such a definition of  $d$  can ensure that the generated off-surface points do not intersect with the other parts of the implicit surface to be produced.

$$\begin{cases} f(p_i) = 0, & p_i \text{ on surface} \\ f(p_i) = d, & p_i \text{ inside surface} \\ f(p_i) = -d, & p_i \text{ outside surface} \end{cases} \quad (5)$$

### 3.3 RBF construction

Suppose the number of 3D points (including original points and their off-surface points) is  $n$ , then the RBF used for surface reconstruction is defined as:

$$f(p_i) = \sum_{j=1}^n w_j \phi(r_{ij}) + P(p_i) \quad (6)$$

where  $P(p_i) = c_0 + c_1 x_i + c_2 y_i + c_3 z_i$ ,  $\phi(r) = r^3$ , and  $r_{ij} = \|p_i - p_j\|$ , the triharmonic spline is chosen as the basis function since it is a good choice for fitting of three variables.



Since the above equation is linear with respect to the unknown weights,  $w_j$ , and coefficients of  $P(p_i)$ , a linear system can be formulated as follows:

$$\begin{bmatrix} \phi(r_{11}) + \lambda_1 & \cdots & \phi(r_{1n}) & 1 & x_1 & y_1 & z_1 \\ \vdots & \cdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \phi(r_{n1}) & \cdots & \phi(r_{nn}) + \lambda_n & 1 & x_n & y_n & z_n \\ 1 & \cdots & 1 & 0 & 0 & 0 & 0 \\ x_1 & \cdots & x_n & 0 & 0 & 0 & 0 \\ y_1 & \cdots & y_n & 0 & 0 & 0 & 0 \\ z_1 & \cdots & z_n & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} w_1 \\ \vdots \\ w_n \\ c_0 \\ c_1 \\ c_2 \\ c_3 \end{bmatrix} = \begin{bmatrix} f(p_1) \\ \vdots \\ f(p_n) \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (7)$$

The computed weights,  $w_j$ , and the coefficients of  $P(p_i)$  are then utilized to construct the RBF.

### 3.4 Output of the constrained surface

The marching cube (MC) method is usually used to generate triangles for the implicit surfaces. Each of the eight vertices of one unit cube is taken as the input for Equation (6) and its result is calculated. From the eight results, using Equation (5), it can be determined whether the implicit surface passes through the unit cube. Since the RBF based implicit surface is unlimited in nature, it is very appropriate for shape reconstruction of the entire object or interpolation of the incomplete mesh together with MC. However, for partial surface recovery, constraints have to be added to define the edges.

As the partial point cloud has been transformed with the help of the PCA method, its plot on the XY plane relates to the most significant of the 3 components. Therefore, when a unit cube is processed in the MC, a cylinder centered with it is defined to help decide whether the cube is within the surface edge along Z axis, as illustrated in Figure 3. The height of the cylinder is more than the range of Z value of the partial point cloud, while the circle radius of the cylinder is adaptively defined in the following way: (1) The maximum edge of the minimum spanning tree of the partial point cloud is taken as the initial radius of the cylinder; (2) If the number of points inside the cylinder is not less than  $N/10$  where N is the number of partial points, go to Step 5; (3) Double the cylinder radius; (4) Go to Step 2; (5) Stop.

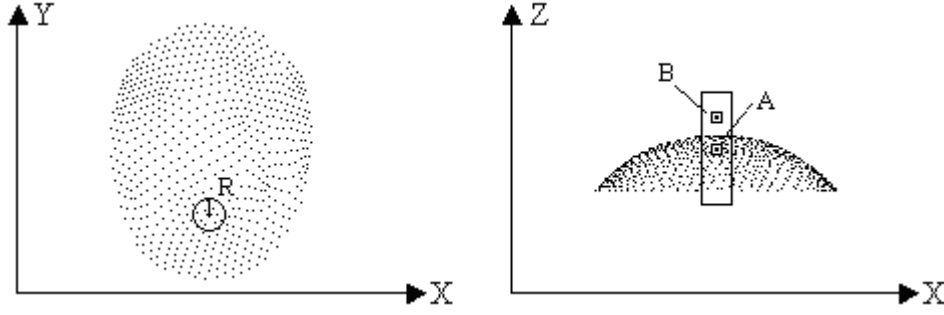


Figure 3. Cylinder centered with one unit cube

Then two extreme coordinate values,  $czmin$  and  $czmax$ , are found from the points within the cylinder, and they are used to define the surface edge along the Z-axis. As shown in the right of Figure 3, unit cubes A and B share the same cylinder, but A is inside the surface edge and thus used for surface generation, while B is discarded as it is outside the surface edge. In the same way, surface edges along X- and Y-axes can be defined as well, and they are used together with surface edge along Z-axis to help produce a constrained shape for the partial point cloud when the implicit surface is generated using the MC method.

#### 4. Efficiency Improvements for RBF

Since computation time is a setback of RBF based surface reconstruction, several methods are explored for efficiency improvements. These determine how to decide the proper number of points in one group and take them as the unit for surface recovery, how to find the suitable approaches for the solutions of covariance matrix and linear system.

##### 4.1 Octree based division and combination

Computational cost of RBF surface reconstruction is related with the number of processing points [8]. It will take a very long time to process a large number of points at the same time, but much time will be saved if the points are divided into many blocks and then dealt with one group at a time. Therefore, space subdivision with octree-based self-adaptive method is adopted to further reduce the time during RBF recovery.

The minimum cubic bounding box for the partial point cloud and their off-surface points is defined as the root of the octree. Then the root is subdivided into 8 smaller cubes with the

same size as its children cells. The subdivision is operated recursively on the cells until each cell has satisfied the requirements. The rules for octree-based adaptive subdivision are as follows:

- If the number of points in the current cell is more than the predefined maximum value  $n_{\max}$ , the cell is a non-leaf cell and subdivided into 8 smaller cells;
- if the number of points in the current cell is no more than  $n_{\max}$ , the cell is taken as a leaf cell without further subdivision;
- if the number of points in the current cell is zero, the cell is marked as NULL and no further subdivision is needed.

A suitable value for  $n_{\max}$  is around 300 based on a series of experiments.

Besides the upper limit of the number of points in the leaf cells of octree, there should be a lower limit as well. If the number of points in one leaf cell is too few, its linear system may be homogeneous, and the surface to be reconstructed will be uncertain. Even if the linear system consists of non-homogeneous linear equations, the leaf cell with too few points may not have a unique solution. The linear system with non-homogeneous linear equations can be represented by the following equation

$$Ax = b \quad (8)$$

where  $A$  is the  $(n+4) \times (n+4)$  matrix, and the sufficient and necessary conditions that Equation (8) has unique solution are that  $b$  can be represented by the column vectors of matrix  $A$ , and the column vectors in matrix  $A$  are linear independent with each other, i.e.

$$r((A, b)) = r(A) = \text{num of columns in } A \quad (9)$$

Relationship between the number of points in one leaf cell and the rank of matrix  $A$  is listed in Table 1. From Table 1, we can find that Equation (8) has the unique solution only if the number of points,  $n \geq 4$ , i.e. one certain surface can be reconstructed from the points in the leaf cell.

Table 1. Relationship between number of points and rank of matrix

Number of points	$r(A)$	$r(A, b)$	Full rank	Equation (9)
1	2	2	5	False

2	4	4	6	False
3	6	6	7	False
4	8	8	8	True
$n (n > 4)$	$n+4$	$n+4$	$n+4$	True

The computation precision has to be considered when two neighbor points are very near, especially for each sampling point and its off-surface point used for RBF. If one sampling point lies in one leaf cell, its off-surface point usually lies in the same leaf cell. As shown in Figure 4, normally, there are pairs of points in the leaf cell. Therefore, it is more suitable to replace the condition  $n \geq 4$  with  $n \geq 2 * 4$  in RBF recovery. In our paper, the predefined minimum threshold value for the number of points in each leaf cell is  $n_{\min} = 10$ .

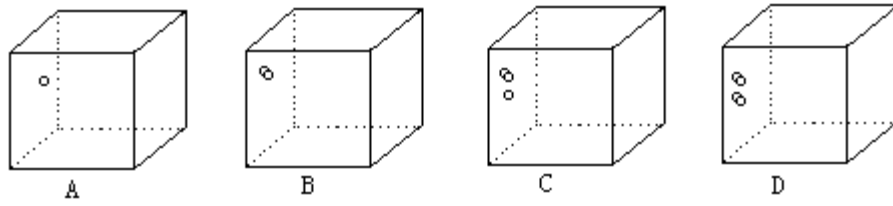


Figure 4. The usual situations of 1~4 points in one leaf cell

After division of octree, the number of points in some leaf cells may be less than  $n_{\min}$ , thus a combination is used to deal with the problem. The leaf cells are checked one by one, and the current leaf cell whose number of points less than  $n_{\min}$  needs to be combined with one of its neighbor leaf cells. Based on the topology relationships, the one with the least number of points is found from all the neighbor leaf cells, and then be combined with the current leaf cell. The combination operation does not really turn these two leaf cells into one, instead the points in both are used during RBF reconstruction. If the number of points in the combined two leaf cells is still less than the predefined  $n_{\min}$ , the combination operation will be performed recursively.

After the above self-adaptive operations of division and combination, there should be a suitable number (no more than  $n_{\max}$  and no less than  $n_{\min}$ ) of points in leaf cells of the

octree. Then the 3D points in each leaf cell are used for the RBF reconstruction, i.e. calculation of unknown weights and coefficients from the linear equations of the points.

## 4.2 Solutions of covariance matrix and linear system

For RBF reconstruction, the surface normal of each point  $p_i$  needs to be computed and its off-surface point is obtained. Therefore, the covariance matrix for the points in the local neighborhood of  $p_i$  is set up, and the eigen vector corresponding to the smallest eigen value of the matrix is taken as the surface normal of point  $p_i$ . As the covariance matrix is a symmetric positive semi-definite matrix and only the smallest eigenvalue needs to be calculated, the inverse power method is adopted.

For the unknown weights and coefficients of RBF, the linear system including a set of linear equations also needs to be computed. LU decomposition is usually used and its time complexity is  $\frac{2}{3}N^3 + O(N^2)$ , while space complexity is  $N^2 + O(N)$ . When the triharmonic spline, a good choice for fitting functions of three variables, is selected as the basic function of RBF reconstruction, the matrix for the linear system is real symmetric and positive definite. Thus the method of improved Cholesky factorization is employed to calculate the solution of the matrix with time complexity  $\frac{1}{6}N^3 + O(N^2)$  and space complexity  $\frac{1}{2}N^2 + O(N)$ . In this way, time and space is saved.

## 5. Girth Measurement Using RBF Surface

Anthropometrical measurement usually performs on the local surface, and global surface reconstruction from the whole point cloud is not necessary. Thus it is one way to reduce the computation time by determining the required 3D points only and then reconstruct the partial surface. As shown in the left of Figure 5, only the black points covered with tape is useful in the girth measurement of human foot. To find the useful points, for the tape plane C defined by three 3D control points  $c_1$ ,  $c_2$  and  $c_3$ , the distance between each point of the point cloud and plane C is calculated. The points whose distance is no more than threshold value D are selected, and they are used to make up point set S. To facilitate the following computation, a local coordinate system is set up with control points  $c_1$ ,  $c_2$  and  $c_3$ , and then it is transformed

together with point set S to be coincident with the global coordinate system, as illustrated in the right of Figure 5.

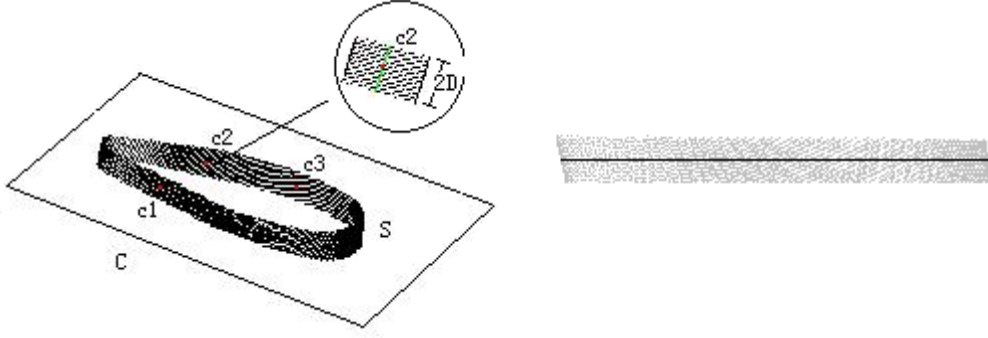


Figure 5. The points used for girth measurement

With our proposed approach for RBF based surface reconstruction, the constrained implicit surface for point set S is generated. To calculate the girth value, the method for measurement simulation is performed in the following way:

- Step 1. For each point of point set S, calculate its 3D distance to tape plane C;
- Step 2. Compare the distance with the threshold value d, which is the maximum edge of the minimum spanning tree of point set S;
- Step 3. Take the points whose distance no more than the threshold value d as the initial intersection points (point set I), illustrated as the black points in the right of Figure 5;
- Step 4. For any point  $p_i$  of point set I, find its K neighbors, i.e. repeatedly increase the radius r of the circle centered with  $p_i$  until the number of the points in the circle is no less than K, as shown in Figure 6;
- Step 5. Fit one straight line m for point  $p_i$  and its K neighbors using least square method;
- Step 6. Generate two off-line points for point  $p_i$ ,  $p_M$  and  $p_N$ , along the opposite directions of the normal of the fitted line;
- Step 7. Calculate the exact intersection point  $p_i'$  between tape plane C and implicit surface of point set S corresponding to the initial intersection point  $p_i$  by

$$p_i' = p_M + (isovalue - V_M)(p_N - p_M)/(V_N - V_M) \quad (10)$$

where  $p_M$  and  $p_N$  are two off-surface points for  $p_i$ ,  $V_M$  and  $V_N$  are their RBF values using Equation (6), isovalue is used to control the relative position (inside, outside or on) of  $p_i$  with

the implicit surface and in our case isovalue is assigned as 0 to make sure that point  $p_i$  lies exactly on the implicit surface;

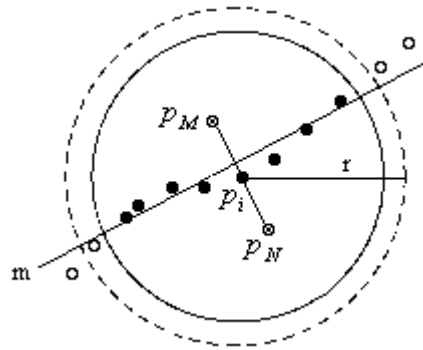


Figure 6. Off-surface points of point  $p_i$

## 6. Experimental Results

Performances of the following three methods for girth measurements are compared with experiments: Method 1, the existing method to project the 3D points nearby the tape plane, and then use 2D convex hull of them to simulate the girth measuring; Method 2, the existing method to calculate the intersection points between tape plane and the generated triangular mesh from 3D points, and then use the intersected segments to simulate the girth measuring; Method 3, our proposed method to calculate the intersection points between tape plane and the generated local RBF implicit surface from 3D points, and then use the intersected points to simulate the girth measuring.

## 6.1 Experiments on measurement accuracy

To test the accuracy of our method, computer generated 3D points of a standard cylinder are used as experimental data since the correct circumference of the cylinder can be computed previously. As shown in Figure 7, surface points for the cylinder with radius value of 40 are generated with different densities, i.e. 1200 points in Model 1, 900 points in Model 2, 450 points in Model 3. Based on the equation  $C = 2 * \pi * r$ , exact circumference of the cylinder is calculated as  $2 * 3.1415926 * 40 = 251.327408$ .

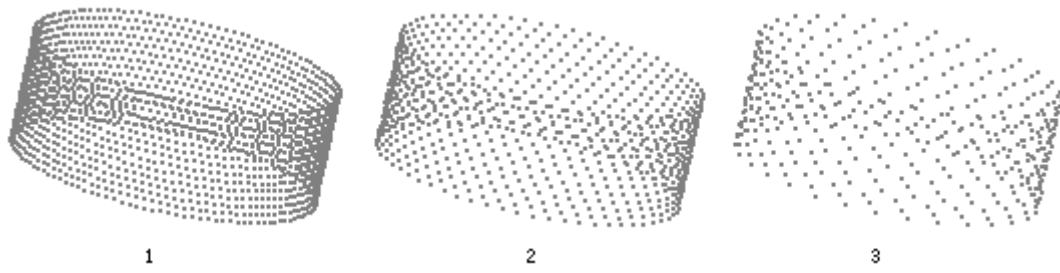


Figure 7. Points of cylinder with different densities

Measurement accuracy is described by the deviation between accurate girth and measured girth, and the deviations of cylinder girth measurements from different methods on different densities of data are listed in Table 2. From Table 2, it can be found that the deviation value increases with the decrease of data density. But for any kind of model, our method can obtain the measured result with the minimum deviation.

Table 2. Deviations of cylinder girth from different methods (Unit: mm)

Algorithm	Deviation (accurate girth – measured girth)		
	Model 1	Model 2	Model 3
Method 1	0.028709	0.114822	0.459100
Method 2	0.018406	0.074521	0.228097
Method 3	0.014857	0.028504	0.033265

Scanned points of real last with different densities are used as experimental data. As shown in Figure 8, Model 1 has 17040 points, Model 2 has 4260 points, while Model 3 has 1080 points. Different with the cylinder, theoretically accurate girth value of the last with irregular surface is not available. Thus we compare the measured girths based on the same tape plane using different methods and different models. From the experimental results listed in Table 3, it can be found that our method has the most stable performance. For example, compared with Model 1, about 93.66% points are missed in Model 3, but the measured girth value from our method has only about 0.51% change.



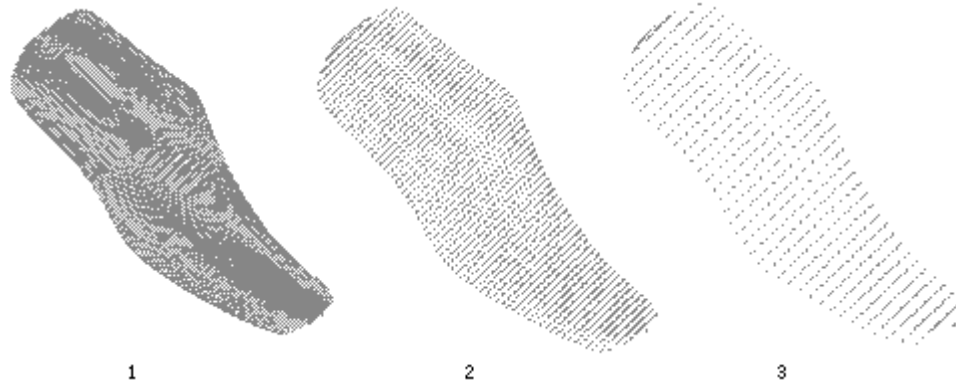


Figure 8. Points of last with different densities

Table 3. Measured results of last girth from different methods (Unit: mm)

Algorithm	Measured girth value		
	Model 1	Model 2	Model 3
Method 1	248.640348	243.796843	239.850287
Method 2	248.749485	247.414452	244.912462
Method 3	248.904068	248.476340	247.630655

## 6.2 Experiments on computing expense

RBF surface modeling has the disadvantage of high computational expense, therefore its computing cost is also tested to make sure that the required time of our proposed approach can be tolerated in the practical applications. Our method is tested on the measurements for short heel girth, instep girth and ball girth of human foot, which are often used in shoe making industry. The measurement period is regarded as the whole procedure from the time that the points are read from file to the time that the calculated girth value is provided. Number of partial points and time (with the unit of milliseconds) for measurements of the three kinds of foot girths are listed in Table 4. It can be found that the computational time of any girth measurement is less than 1 second, which satisfies the requirements of custom footwear in shoe making industry.

Table 4. Computational cost for girth measurements

	Short heel girth	Instep girth	Ball girth
Number of partial points	9090	7112	7044
Time for measurement	943	720	693

## 7. Conclusion

The existing girth measurement algorithms from point cloud or triangular mesh can obtain satisfactory results if the data to be processed has high density. But their measuring accuracy decreases obviously with low data density since points and triangles are just relative simple descriptions of the object's surface. Parametric surface such as NURBS has the ability to describe the continuous surface, but it cannot be directly used on the unorganized 3D points and the generated surface only approximates the scanned points. Thus the better method for automatic girth measurement is needed since it is very valuable for custom manufacture such as custom shoemaking in footwear industry.

Different with the existing methods, RBF based implicit surface modeling is employed for girth measurement from 3D point cloud. Our approach can operate on the unorganized 3D points without previous processing and make the produced surface go through each sample point. Therefore it provides more stable and accurate measurements, which has been proved with experiments on computer generated surface points of regular cylinder and scanned points of real irregular last surface.

To reduce the computational expense of our method, only the needed local surface for girth measurement is reconstructed to avoid unnecessary computation, octree based division and combination are used to speed the process of RBF reconstruction, inverse power method and improved Cholesky factorization are presented in the solution of covariance matrix and linear system. Our approach is tested in practical girth measurements for human foot, and the girth value can be computed in relative short time less than 1 second.

## Acknowledgement

The work was supported by NSFC (No. 60603079) and the Research Grants Council of Hong Kong (613008).

## References

- [1] Bunch RP, Foot measurement strategies for fitting athletes, *Journal of Testing and Evaluation*, 16(4), 1988, pp. 407-411.
- [2] Liu W, Miller J, Stefanyshyn D, Nigg BM, Accuracy and reliability of a technique for quantifying foot shape, dimensions and structural characteristics, *Ergonomics*, 42(2), 1999, pp. 346-358.
- [3] Tsung BYS, Zhang M, Fan YBF, Boone DA, Quantitative comparison of plantar foot shapes under different weight bearing conditions, *Journal of Rehabilitation Research and Development*, 40(6), 2003, pp. 517-526.
- [4] Xu C, Liu Y, Jiang Y, Pan Y, The design and implementation for personalized shoe last CAD system, *Journal of Computer-Aided Design & Computer Graphics*, 16(10), 2004, pp. 1437-1441.
- [5] Witana CP, Xiong S, Zhao J, Goonetilleke RS, Foot measurements from 3-dimensional scans: A comparison and evaluation of different methods, *International Journal of Industrial Ergonomics*, 36(9), 2006, pp. 789-807.
- [6] Zhao J, Xiong S, Bu Y, Goonetilleke RS, Computerized girth determination for custom footwear manufacture, *Computers & Industrial Engineering*, 54(3), 2008, pp. 359-373.
- [7] Carr JC, Fright WR, Beatson RK, Surface Interpolation with Radial Basis Functions for Medical Imaging, *IEEE Transactions on Medical Imaging*, 16(1), 1997, pp. 96-107.
- [8] Carr JC, Beatson RK, Cherrie JB, Mitchell TJ, Fright WR, McCallum BC, Reconstruction and representation of 3D objects with radial basis functions, *Computer Graphics (Proceedings of SIGGRAPH 2001)*, August, 2001, pp. 67-76.
- [9] Ohtake Y, Belyaev A, Seidel HP, 3D Scattered Data Approximation with Adaptive Compactly Supported Radial Basis Functions, *Proceedings of the Shape Modeling International 2004 (SMI'04)*, pp. 31-39.
- [10] Ohtake Y, Belyaev A, Seidel HP, 3D scattered data interpolation and approximation with multilevel compactly supported RBFs, *Graphical Models*, 67(3), 2005, pp. 150-165.
- [11] Dinh HQ, Turk G, Slabaugh G, Reconstruction surfaces by volumetric regularization using radial basis functions, *IEEE transactions on pattern analysis and machine intelligence*, 24(10), 2002, pp. 1358-1371.

- [12] Marie S, Marc A, Pierre A, Mariette Y, Reconstruction with Voronoi Centered Radial Basis Functions, Eurographics Symposium on Geometry Processing, 2006, pp. 51-60.
- [13] Ohtake Y, Belyaev A, Alexa M, Turk G, Seidel H, Multi-level Partition of Unity Implicits, Proceedings of Siggraph 2003, pp. 463-470.
- [14] Lin H, Tai C, Wang G, A Mesh Reconstruction Algorithm Driven By Intrinsic Property of Point Cloud, Computer-Aided Design, 26(1), 2004, pp. 1-9.