Engineering with Computers
DOI 10.1007/s00366-010-0199-1

ORIGINAL ARTICLE

A two-level approach to implicit surface modeling with compactly supported radial basis functions

Rongjiang Pan · Vaclav Skala

Received: 16 August 2009 / Accepted: 19 October 2010

Abstract We describe a two-level method for computing a function whose zero-level set is the surface reconstructed from given points scattered over the surface and associated with surface normal vectors. The function is defined as a linear combination of compactly supported radial basis functions (CSRBFs). The method preserves the simplicity and efficiency of implicit surface interpolation with CSRBFs and the reconstructed implicit surface owns the attributes, which are previously only associated with globally supported or globally regularized radial basis functions, such as exhibiting less extra zero-level sets, suitable for inside and outside tests. First, in the coarse scale approximation, we choose basis function centers on a grid that covers the enlarged bounding box of the given point set and compute their signed distances to the underlying surface using local quadratic approximations of the nearest surface points. Then a fitting to the residual errors on the surface points and additional off-surface points is performed with fine scale basis functions. The final function is the sum of the two intermediate functions and is a good approximation of the signed distance field to the surface in the bounding box. Examples of surface reconstruction and set operations between shapes are provided.

Keywords Implicit surface modeling · Radial basis function · Surface reconstruction · Boolean operation

R. Pan (✉)
School of Computer Science and Technology,
Shandong University, Jinan, China
e-mail: panrj@sdu.edu.cn

V. Skala
Centre of Computer Graphics and Data Visualization,
Department of Computer Science and Engineering,
University of West Bohemia, Plzen, Czech Republic

1 Introduction

Reconstructing a surface from sample points is a powerful 3D modeling method in CAD, reverse engineering, cultural heritage protection and other applications. Recent improvements of shape acquisition techniques have made it easier to digitize a world object into large unorganized point sets. Approaches to surface reconstruction can be roughly divided into two classes: Delaunay-based methods and implicit surface methods [11].

An implicit surface model is given by the zero-level set $f^{-1}(0)$ of a function $f(\mathbf{x})$. The function $f(\mathbf{x})$ has positive values outside the object and negative values inside the object, which is the convention within this paper. It is more suitable for collision detection and set operations between shapes than other surface representations, as the sign of $f(\mathbf{x})$ indicating whether a point $\mathbf{x}$ is interior or exterior to the surface. Among various implicit surface methods, techniques based on radial basis functions (RBFs) [20] have proven to be valid and equally expressive to others. For example, it can produce water-tight surfaces and does not impose constraints on the topological complexity of the model.

The implicit surface representation with RBFs is a linear combination of RBF. In order to find the set of weights, a linear system of equations need to be solved given constraints, centers and type of RBF. There are generally three methods known to overcome its computational problem for large point sets. The first approach is FastRBF algorithm based on globally supported basis functions and fast multipole method [1]. However, it is very difficult to implement correctly. The second approach is to divide a large set of points into a number of small sub-domains and the final function is obtained by blending intermediate functions [6, 15]. Unfortunately, many undesirable artifacts are often

Published online: 04 November 2010

yielded, making it unsuitable for testing whether a given point is interior or exterior to the surface. The third group of methods uses compactly supported RBFs (CSRBFs) to make the linear system sparse. As the method is efficient and easy to implement, it has been widely used recently [7, 8]. Nevertheless, it has some problems easily seen in Fig. 1. As the CSRBF function is only defined in a small neighborhood of the sampled surface, it is sensitive to the quality of input data and often yields surfaces with some unwanted artifacts and spurious zero-level sets in addition to the lack of extrapolation across large holes. The sign of $f(\mathbf{x})$ is not consistently negative inside the surface and positive outside the surface. This causes no serious problems in polygonization [2, 3] and rendering of implicit surfaces since we can skip the cubes or tetrahedra far from the sample points and some extraneous zero-level sets are also obscured by the surface. However, it is not suitable for collision detection and set operations between shapes.

To keep the advantages of CSRBF and overcome its problems, Ohtake et al. [8] use spatial down sampling to construct a coarse-to-fine hierarchy of point sets and fit hierarchical functions to the residual errors of the previous level with basis functions of diminishing support. Although this approach is insensitive to the density of scattered points and can fill large holes in the surface, the sign of the function is still not fully consistent with the underlying signed distance field to the surface. Using centers of basis functions selected among the vertices of the Voronoi diagram of the input data points, an approximation of the signed distance to the surface defined all around the sampled shape is achieved by Samozino et al. [10]. However, it needs a user-defined set of centers and its efficiency and scalability need to be improved. By regularizing multi-scale compactly supported basis functions, Walder et al. [17] propose a framework that leads to the desirable properties of implicit modeling with CSRBFs. To build the regularisers, a complicated expression has to be evaluated, which covers a whole page in the technical report from the group [18].

Compared with implicit surface methods based on globally supported RBFs, partition of unity method [21] gives nice properties concerning processing time and sharp feature reconstruction. However, it results in reconstructions with unwanted spurious surface artifacts because it only constrains the implicit function near the sample points.

The goal of our work is to produce a good approximation of the signed distance to the surface inside the bounding box of the shape as depicted in Fig. 1c, while preserving the simplicity and efficiency of implicit surface interpolation with CSRBFs since it is more suitable for inside and outside tests. We use a two-level fitting approach. In the coarse scale approximation, we choose centers of basis functions by a grid that covers the enlarged bounding box of the given point set and compute their signed distances to the underlying surface using local quadratic approximations of the nearest surface points. Then a fitting to the residual errors on the surface points and additional off-surface points is performed with fine scale basis functions. The final function is the sum of the two intermediate functions and is a good approximation of the signed distance field to the surface inside the bounding box. Moreover, the zero-level set $f(\mathbf{x}) = 0$ of the function $f(\mathbf{x})$ interpolates the input point set rather than approximation in references [10, 17].

The proposed approach constructs an approximate distance field. Freytag et al. [23] computed an approximate distance field fit to samples of exact signed distance to the geometry. In essence, the approach is very close to the current one, but the implementation is quite different. They

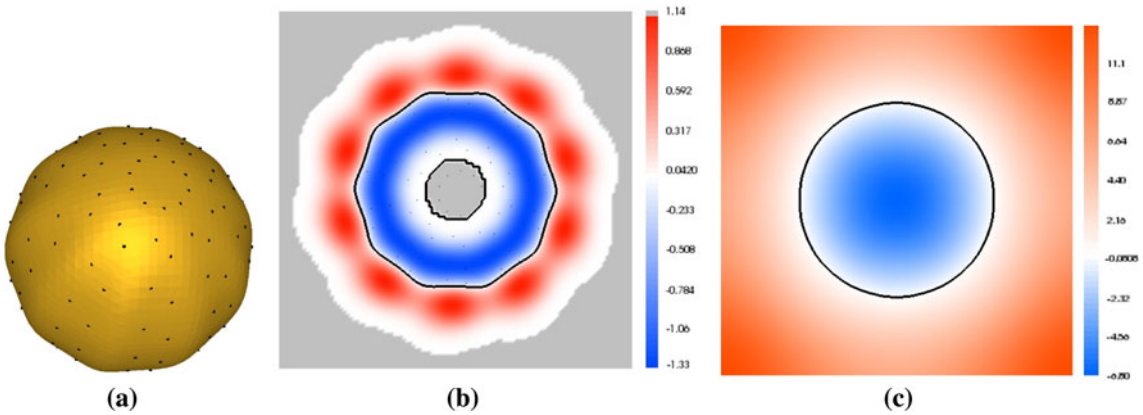

Fig. 1 **a** A surface reconstructed from 180 sample points depicted as *black dots*. **b** A planar slice that cuts the surface—the *colors* represents the function values and the *black curve* highlights its zero level. **c** The desired function values of our methods. In **b** and **c**, positive values are mapped to *red* and negative values to *blue* (color figure online)

chose the trilinear B-splines arranged on a regular Cartesian grid and employed Dirichlet boundary conditions.

The paper is organized as follows: we explain the details of our algorithm in Sect. 2. Examples of surface reconstruction and set operations between shapes are given in Sect. 3. Some future works are discussed in Sect. 4.

2 Algorithm

In the RBF approach, given a set of N 3D points $P = \{p_1, p_2, \ldots, p_N\}$ and N scalar values $\{f_1, f_2, \ldots, f_N\}$, we interpolate the points by a function in the following form

$$f(x) = \sum_{j=1}^{N} w_j \phi_\sigma\left(\left\|x - p_j\right\|\right) + Q(x) \tag{1}$$

such that

$$f(p_i) = f_i, \quad i = 1, \ldots, N \tag{2}$$

where $\phi_\sigma(r)$ is an RBF function, $Q(\mathbf{x}) = q_0 + q_1 x + q_2 y + q_3 z$ is a polynomial accounting for the linear and constant portions of f, and w_j is the weight to be determined.

Let $\mathbf{p}_i = (p_i^x, p_i^y, p_i^z)$ and $\phi_{ij} = \phi_\sigma\left(\left\|\mathbf{p}_i - \mathbf{p}_j\right\|\right)$. With (1), (2) and side conditions given by (3),

$$\sum_{i=1}^{N} w_i = 0, \quad \sum_{i=1}^{N} w_i p_i^x = 0, \quad \sum_{i=1}^{N} w_i p_i^y = 0, \quad \sum_{i=1}^{N} w_i p_i^z = 0 \tag{3}$$

a linear system is built up as follows:

$$\begin{bmatrix} \phi_{11} & \phi_{12} & \cdots & \phi_{1N} & 1 & p_1^x & p_1^y & p_1^z \\ \phi_{21} & \phi_{22} & \cdots & \phi_{2N} & 1 & p_2^x & p_2^y & p_2^z \\ \vdots & \vdots & & \vdots & \vdots & \vdots & \vdots & \vdots \\ \phi_{N1} & \phi_{N2} & \cdots & \phi_{NN} & 1 & p_N^x & p_N^y & p_N^z \\ 1 & 1 & \cdots & 1 & 0 & 0 & 0 & 0 \\ p_1^x & p_2^x & \cdots & p_N^x & 0 & 0 & 0 & 0 \\ p_1^y & p_2^y & \cdots & p_N^y & 0 & 0 & 0 & 0 \\ p_1^z & p_2^z & \cdots & p_N^z & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_N \\ q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_N \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

whose solution gives the unknown weights w_j, $j = 1, \ldots, N$ and q_k, $k = 0, \ldots, 3$.

Although in the following analysis, we choose Wendland's CSRBF $\phi_\sigma(r) = \phi(r/\sigma)$ with C^2 continuity [19],

$$\phi(r) = \begin{cases} (1-r)^4(4r+1), & 0 \le r \le 1 \\ 0, & r > 1 \end{cases},$$

where σ is the support size, or scale of the basis function, our method is general enough to be applicable to any other kinds of CSRBF functions.

Since we use CSRBFs, the linear system is symmetric, positive semi-definite and sparse, which can be efficiently stored and solved by a direct LU sparse solver or an iterative conjugate gradients stabilized method [9].

In implicit modeling with RBFs, for all the points sampled from a surface, the scalar values are zero, $f_i = 0$. To avoid the trivial solution $w_j = 0$, $j = 1, \ldots, N$ and $q_k = 0$, $k = 0, \ldots, 3$, surface points are usually equipped with oriented normals $\{n_1, n_2, \ldots, n_N\}$ and off-surface points are created along the normals. If the point clouds are exact or preprocessed, we often require the surface interpolates the points. In case of points with noise, it is preferable to approximate rather than to exactly interpolate the data. The matrix equations in 2.1 and 2.2 can be modified to allow the constraint points to be approximated. We refer the reader to [22] for more details.

In the following of this section, we present the main steps of our reconstruction algorithm, namely coarse scale interpolation and fine scale interpolation.

2.1 Coarse scale approximation

First, we must choose the domain of functions in which to define the model. The axis-aligned bounding box is one of the simplest ways for enclosing a shape. To test whether a point is inside or outside a surface, we often first check the point against the bounding box of the surface. If it is interior to the bounding box, we can further decide the sign of the function. Thus, we limit our approximation of the desired signed distance to the surface into the enlarged axis-aligned bounding box of the sampled points, which encloses the whole shape reconstructed from the sample points.

The support size σ of the CSRBF function is estimated heuristically as recommended in [8]. We use an adaptive octree-based subdivision of the bounding box and stop the subdivision if each leaf cell containing no more than eight points, and delete the leaf cells containing no points. We set σ equal to 3/4 of the average diagonal length of the leaf cells.

Then we enlarge the axis-aligned bounding box of the sample points by 4σ along each axis and adjust its side length to multiple of 4σ. We put a grid of spacing 4σ over the entire region of the enlarged axis-aligned bounding box. For each grid cell that contains sample points, we subdivide it into eight subcells by a fine grid of spacing 2σ. If the subcell contains points, we further decompose it into eight smaller subcells of width σ. The centers of cells and subcells are selected as the centers $\{c_1, c_2, \ldots, c_K\}$ of basis functions as illustrated in Fig. 2a. We let the support size σ_1 equal to 3/4 of the diagonal length of the coarsest cell so that $\sigma_1 = 4\sqrt{3}\sigma$.

In order to estimate the function value on these centers, for each center $\mathbf{c}_i$, we find the nearest input sample point $\mathbf{p}_j$ based on a *k–d* tree. If the distance between $\mathbf{c}_i$ and $\mathbf{p}_j$ is zero, the distance from center $\mathbf{c}_i$ to the underlying surface is taken as

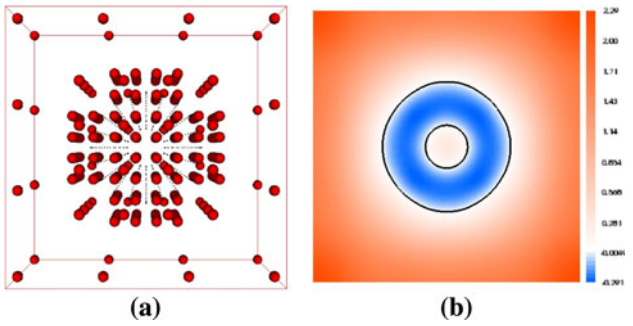

Fig. 2 **a** Selected centers of the coarse scale approximation are visualized as *red spheres*. The input sample points depicted as *black dots*. **b** A planar slice that cuts the enlarged bounding box—the *colors* represent the function values and the *black curve* highlights its zero level. Positive values are mapped to *red* and negative values to *blue* (color figure online)

zero. Otherwise, we fit a local quadratic approximation $g_j(\mathbf{x})$ at $\mathbf{p}_j$ to its neighborhood $N = \{\mathbf{x} \in N : \|\mathbf{x} - \mathbf{p}_j\| < \sigma\}$ as detailed in [8]. Let (u, v, w) be the local orthogonal coordinate system with the origin of coordinates at $\mathbf{p}_j$ such that the plane (u, v) is orthogonal to $\mathbf{n}_j$ and the positive direction of w coincides with the normal $\mathbf{n}_j$ of $\mathbf{p}_j$. The coefficients of the local fitting function $g(u, v, w) = w - (Au^2 + 2Buv + Cv^2 + Du + Ev + F)$ are determined by the following minimization procedure

$$\min \sum_k \phi_\sigma(\|\mathbf{p}_k - \mathbf{p}_j\|) g(u_k, v_k, w_k)^2,$$

where (u_k, v_k, w_k) are the local coordinates of $\mathbf{p}_k \in N$. The solution of this least-squares problem can be obtained by the singular value decomposition approach [9].

We choose to use Taubin's first-order approximation $d_i = \frac{g(u_i,v_i,w_i)}{\|\nabla g(u_i,v_i,w_i)\|}$ of the signed distance from center $\mathbf{c}_i$ to the underlying surface [14], where (u_i, v_i, w_i) are the local coordinates of $\mathbf{c}_i$. Since d_i is only meaningful in the nearby of the local quadratic fitting function $g(u, v, w)$, we take $d_i = (\mathbf{c}_i - \mathbf{p}_j)^{\mathrm{T}}\mathbf{n}_j$ as the signed distance from center $\mathbf{c}_i$ to the underlying surface if $|(\mathbf{c}_i - \mathbf{p}_j)^{\mathrm{T}}\mathbf{n}_j| < |d_i|$.

Thus, we get a coarse scale approximation f^1 of the input points,

$$f^1(\mathbf{x}) = \sum_{j=1}^{K} w_j \phi_{\sigma_1}(\|\mathbf{x} - \mathbf{c}_j\|) + Q(\mathbf{x}),$$

by solving the linear system

$$\begin{bmatrix} \phi_{11} & \phi_{12} & \cdots & \phi_{1K} & 1 & c_1^x & c_1^y & c_1^z \\ \phi_{21} & \phi_{22} & \cdots & \phi_{2K} & 1 & c_2^x & c_2^y & c_2^z \\ \vdots & \vdots & & \vdots & \vdots & \vdots & \vdots & \vdots \\ \phi_{K1} & \phi_{K2} & \cdots & \phi_{KK} & 1 & c_K^x & c_K^y & c_K^z \\ 1 & 1 & \cdots & 1 & 0 & 0 & 0 & 0 \\ c_1^x & c_2^x & \cdots & c_K^x & 0 & 0 & 0 & 0 \\ c_1^y & c_2^y & \cdots & c_K^y & 0 & 0 & 0 & 0 \\ c_1^z & c_2^z & \cdots & c_K^z & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_K \\ q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_K \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix},$$

where $\phi_{ij} = \phi_{\sigma_1}(\|\mathbf{c}_i - \mathbf{c}_j\|)$, $i,j = 1,\ldots,K$ and $\{c_1, c_2, \ldots, c_K\}$ is the set of selected centers.

Figure 2b illustrates the approximate signed distance field of input points in Fig. 2a.

2.2 Fine scale interpolation

After getting the coarse scale approximation of the input points, we interpolate the points by fine scale RBFs with support size σ.

If there is no center selected in some thin parts of the point set surfaces, the approximate signed distance field according to $f^1(\mathbf{x})$ is not correct in these regions, as shown in Fig. 3a, b.

Observation The parts of the model, which cannot be correctly approximated by coarse scale centers, must be included in a subcell of size σ. Residuals of surface points are usually large in these parts than in others.

Therefore, we need more constraints in those parts having large residuals. In addition to the surface point's constraints, we use additional off-surface points to solve this problem.

First, we calculate the residual $r_i = 0 - f^1(\mathbf{p}_i)$ at each input sample point $\mathbf{p}_i$, $i = 1,\ldots,N$. If the absolute error $|r_i|$ exceeds a given threshold $\varepsilon \geq 0$, we add an off-surface point along its normal $\mathbf{n}_i$ according to the following bisection algorithm. In our implementation, we set ε equal to the average absolute residual, $\varepsilon = \frac{1}{N}\sum_{i=1}^{N} |r_i|$

Input: a surface point $\mathbf{p}_i \in P$ whose absolute error $|r_i|$ exceeds a threshold $\varepsilon \geq 0$.
Output: an off-surface point $\mathbf{q}$ and its scalar value r.

```
distHigh ← 0.5*σ
distLow ← 0.0
if (r_i > ε) then
    q ← p_i − distHigh*n_i
else
    q ← p_i + distHigh*n_i
end if
if (p_i is not the nearest point of q) then
  while (distHigh > distLow) do
    distMiddle ← (distHigh + distLow)/2
    if (r_i > ε) then
        q ← p_i − distMiddle*n_i
    else
        q ← p_i + distMiddle*n_i
    end if
    if (p_i is the nearest point of q) then
        distLow ← distMiddle
    else
        distHigh ← distMiddle
    end if
  end while
end if
estimate the signed distance dist from q to p_i using the method in Section 2.1
r ← dist − f^1(q)
```

The off-surface points $\{p_{N+1}, p_{N+2}, \ldots, p_M\}$ are appended to the input surface points such that

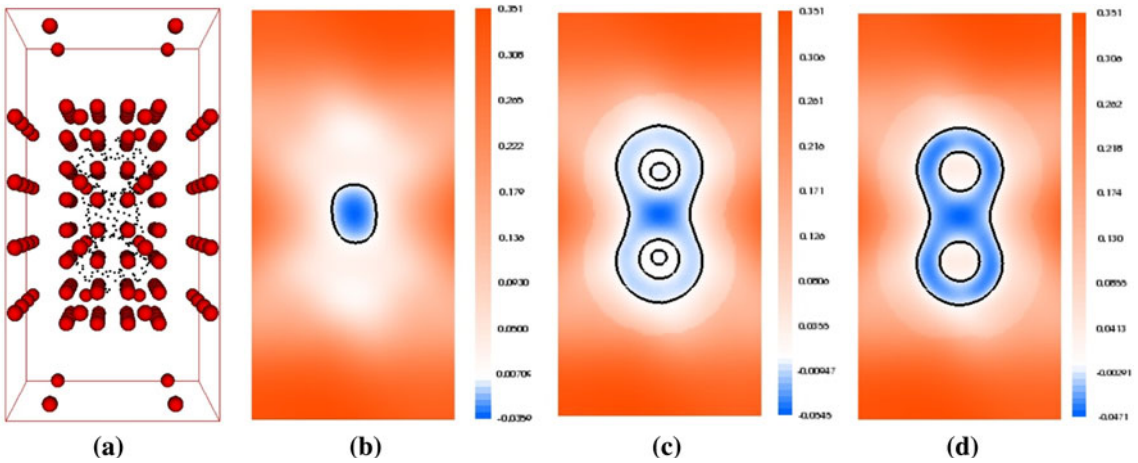

Fig. 3 **a** Selected centers of the coarse scale approximation are visualized as *red spheres*. The input sample points depicted as *black dots*. **b** A planar slice that cuts the coarse scale approximation. Note the distance function is not correct in the thin parts. **c** A planar slice that cuts the fine scale interpolation with surface points. The extra zero-level set is observed. **d** A planar slice that cuts the fine scale interpolation with surface points and off-surface points. In **b**–**d**, the *colors* represent the function values and the *black curve* highlights its zero level. Positive values are mapped to *red* and negative values to *blue* (color figure online)

Table 1 Results of our method for five models

Model	#Points	#Centers 1	#Centers 2	PSNR (dB)	T_{coarse}	T_{fine}	T_{total}
2Torus	4,352	884	6,231	169.09	0.98	2.58	3.56
Knot	28,659	10,640	39,802	123.97	11.84	44.13	55.97
Bunny	34,835	14,903	48,842	189.79	17.86	16.91	34.77
Hand	36,616	11,064	46,258	188.25	14.54	54.06	68.60
Armadillo	165,954	77,595	228,626	139.25	110.14	460.33	570.50

Column two is the number of sample points. Column three and four are numbers of centers in the coarse scale approximation and fine scale interpolation, respectively. PSNR is given in the five column. The remaining columns are timing results (in s) with a 1.80 GHz Pentium 4 processor T_{coarse} the time cost by coarse scale interpolation, T_{fine} is the time cost by fine scale interpolation, T_{total} is the total fitting time

$$f(\mathbf{p}_i) = r_i, \quad i = N+1, \ldots, M.$$

After solving the linear system

$$\begin{bmatrix} \phi_{11} & \phi_{12} & \cdots & \phi_{1M} & 1 & p_1^x & p_1^y & p_1^z \\ \phi_{21} & \phi_{22} & \cdots & \phi_{2M} & 1 & p_2^x & p_2^y & p_2^z \\ \vdots & \vdots & & \vdots & \vdots & \vdots & \vdots & \vdots \\ \phi_{M1} & \phi_{M2} & \cdots & \phi_{MM} & 1 & p_M^x & p_M^y & p_M^z \\ 1 & 1 & \cdots & 1 & 0 & 0 & 0 & 0 \\ p_1^x & p_2^x & \cdots & p_M^x & 0 & 0 & 0 & 0 \\ p_1^y & p_2^y & \cdots & p_M^y & 0 & 0 & 0 & 0 \\ p_1^z & p_2^z & \cdots & p_M^z & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_M \\ q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} = \begin{bmatrix} r_1 \\ r_2 \\ \vdots \\ r_M \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

where $\phi_{ij} = \phi_\sigma(\|\mathbf{p}_i - \mathbf{p}_j\|)$, $i,j = 1,\ldots,M$ and $\{p_1, p_2, \ldots, p_M\}$ is the set of surface points and additional off-surface points, we get the final scale interpolation f of the input points,

$$f(\mathbf{x}) = f^1(\mathbf{x}) + \sum_{j=1}^{M} w_j \phi_\sigma(\|\mathbf{x} - \mathbf{p}_j\|) + Q(\mathbf{x})$$

The effects of fine scale interpolation with and without additional off-surface points are given in Fig. 3c, d, respectively.

3 Results

In our implementation, we used the TAUCS library [16] to solve the linear system. To evaluate the fitting accuracy, we used the peak signal-to-noise ratio (PSNR) as suggested in [4]. The PSNR is defined as $\text{PSNR}[\text{dB}] = 20\log_{10}\frac{\text{peak}}{d}$, where peak is the diagonal length of the model's bounding box and d is the average of algebraic sum of the Taubin distances [14], $d = \frac{1}{N}\sum_{i=1}^{N}\frac{|f(p_i)|}{\|\nabla f(p_i)\|}$.

The details of our reconstruction method for five models obtained from the AIM@SHAPE repository are listed in Table 1 and the reconstructed surfaces are shown in Figs. 4, 5, 6, 7, 8. When running the method, we estimated a sample's normal from the positions of its neighbors. Although more sophisticated methods can be used to extract these surfaces, e.g. [2, 3], we use the code published by Lewiner [5] in order to compare the result with others and to be sure that there are no extra zero-level set lying away from the data.

To demonstrate that implicit models reconstructed with our approach are suitable for set operations, we perform the

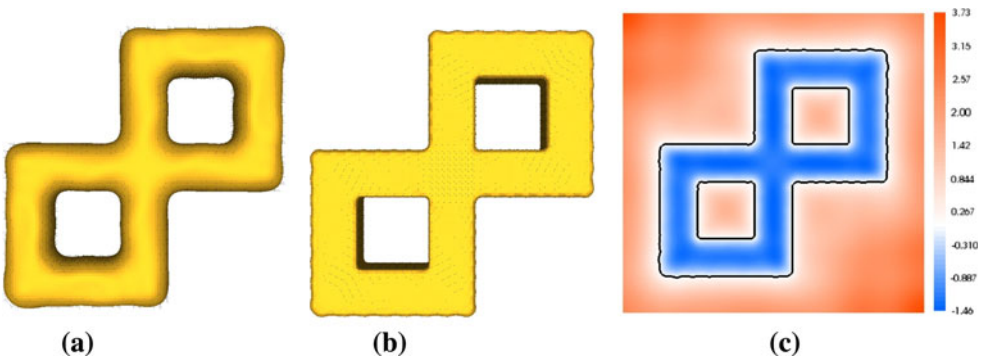

Fig. 4 Reconstruction of 2Torus. *Left* coarse scale approximation, *middle* fine scale interpolation, *right* planar slice that cut the models. The *colors* represent the function values (*red* for positive value, *blue* for negative value) and the *black curve* highlights its zero level, which has the same meaning in the following figures (color figure online)

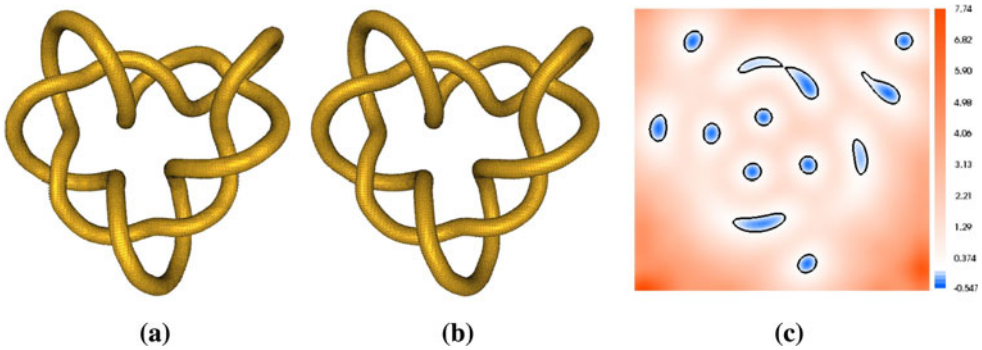

Fig. 5 Reconstruction of knot. *Left* coarse scale approximation, *middle* fine scale interpolation, *right* planar slice that cut the models

Fig. 6 Reconstruction of bunny. *Left* coarse scale approximation, *middle* fine scale interpolation, *right* planar slice that cut the models

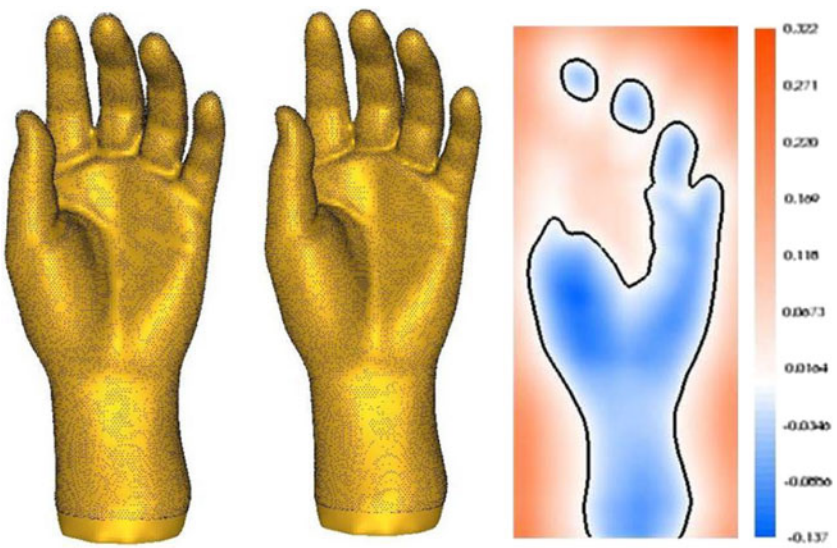

Fig. 7 Reconstruction of hand. *Left* coarse scale approximation, *middle* fine scale interpolation, *right* planar slice that cut the models

Fig. 8 Reconstruction of armadillo. *Left* coarse scale approximation, *middle* fine scale interpolation, *right* planar slice that cut the models

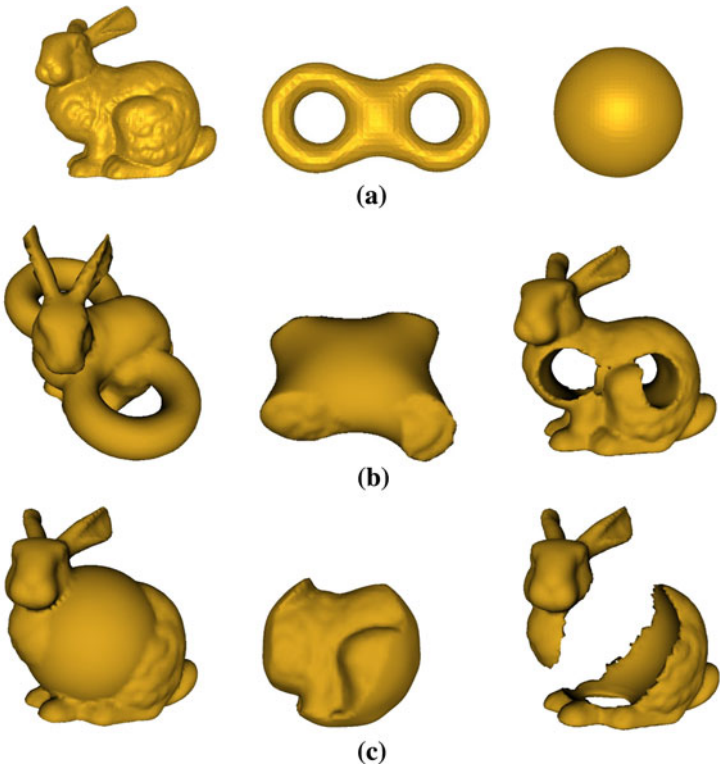

Fig. 9 **a** Three models used for Boolean operations. The three columns in **b** and **c** are the results of union, intersection and difference operations, respectively

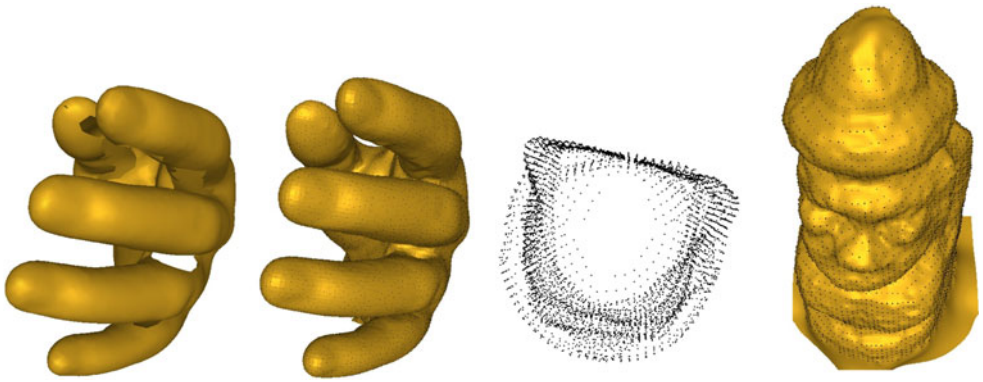

Fig. 10 The *left column* is original models with large holes, the *right column* shows models reconstructed from the sampling points

Boolean operations union, intersection and difference implemented in Visualization Toolkit (VTK) [12, 13]. The results of the set operations between two models are shown in Fig. 9.

As demonstrated in Fig. 10, our approach also repairs large holes in the input data gracefully.

Additionally, we compared the results of our method and the MPU method [21] to a data set (50,000 points) in Fig. 11. Although the reconstruction was computed in 11.1 s with the MPU method and 36.8 s with our method, the MPU method resulted in reconstructions with some spurious surface sheets.

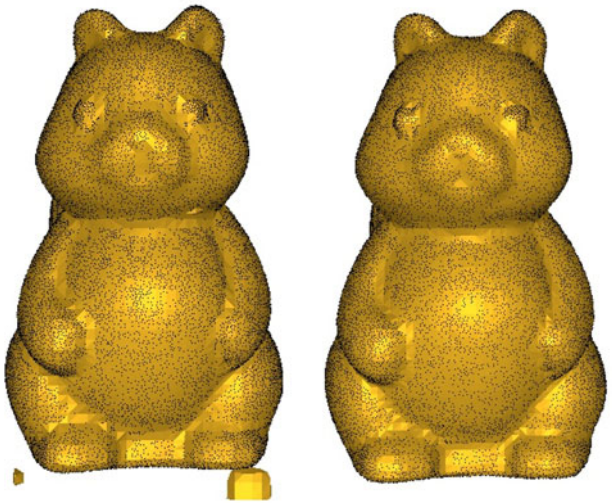

Fig. 11 Comparison with the MPU method for a simple shape. The *left figure* is reconstructed with MPU and the *right figure* is reconstructed with the proposed method

4 Summary

We have presented a fast and simple approach to reconstruct implicit surface with CSRBFs. The reconstructed surface exhibits less unwanted artifacts, which is previously obtained only by globally supported or globally regularized RBFs. By combining a coarse scale approximation and a fine scale interpolation, the signed distance field is correctly approximated inside the enlarged bounding box of the model. The running time and memory usage are significant favorable with respect to previous approaches with globally supported or globally regularized RBFs [1, 10, 17], since we only have to solve two sparse linear systems and do not require to assemble matrices like $\mathbf{A}^T\mathbf{A}$ for a large sparse matrix $\mathbf{A}$, which is a highly time consuming task as the timing results shown in [10, 17].

Compared with the down sampling multi-scale method [8], our approach builds a more accurate coarse scale approximation based on local quadratic approximations of surface points and eliminates small extra zero-level sets by additional off-surface points.

In our future work, we will investigate how to employ fewer centers in the coarse scale approximation and fine scale interpolation based on the geometric features and local sampling density of the model.

Acknowledgments The authors thank their colleagues at Shandong University and University of West Bohemia for their critical comments and suggestions. This work has been supported by the international exchange scholarship between China and Czech governments, the project VIRTUAL 2C06002 Ministry of Education of the Czech Republic and the Key Project in the National Science & Technology Pillar Program of China under grant No.2008BAH29B02.

References

1. Carr JC, Beatson RK, Cherrie JB, Mitchell TJ, Fright WR, McCallum BC, Evans TR (2001) Reconstruction and representation of 3d objects with radial basis functions. In: ACM SIGGRAPH 2001. ACM Press, New York, pp 67–76
2. Cermak M, Skala V (2005) Polygonization of implicit surfaces with sharp features by the edge spinning, the visual computer, vol 21, no 4. Springer, Berlin, pp 252–264. ISSN 0178-2789
3. Cermak M, Skala V (2004) Edge spinning algorithm for implicit surfaces, applied numerical mathematics, vol 49, no 3–4. Elsevier, Amsterdam, pp 331–342. ISSN 0168-9274
4. Kitago M, Gopi M (2006) Efficient and prioritized point subsampling for CSRBF compression. EUROGRAPHICS symposium on point based graphics, pp 121–128
5. Lewiner T, Lopes H, Vieira A, Tavares G (2003) Efficient implementation of marching cubes cases with topological guarantees. J Graph Tools 8(2):1–15
6. Li Q, Wills D, Phillips R, Viant WJ, Griffiths JG, Ward J (2004) Implicit fitting using radial basis functions with ellipsoid constraint. Comput Graph Forum 23(1):55–70
7. Morse BS, Yoo TS, Chen DT, Rheringans P, Subramanian KR (2001) Interpolating implicit surfaces from scattered surface data using compactly supported radial basis functions. In: SMI '01: Proceedings of international conference on shape modeling and applications. IEEE Computer Society, Washington
8. Ohtake Y, Belyaev A, Seidel H-P (2003) A multi-scale approach to 3d scattered data interpolation with compactly supported basis functions. In: Proceedings of international conference shape modeling. IEEE Computer Society, Washington, pp 153–161
9. Press WH, Teukolsky SA, Vetterling WT, Flannery BP (1993) Numerical recipes in C: the art of scientific computing. Cambridge University Press, New York
10. Samozino M, Alexa M, Alliez P, Yvinec M (2006) Reconstruction with Voronoi centered radial basis functions. In: Eurographics symposium on geometry processing. ACM Press, Cagliari, pp 51–60
11. Schall O, Samozino M (2005) Surface from scattered points: a brief survey of recent developments. In: 1st international workshop on semantic virtual environments. MIRALab, Villars-sur-Ollon, pp 138–147
12. Schroeder W, Martin K, Lorensen W (1998) Visualization toolkit, 2nd edn. Prentice Hall, Englewood Cliffs
13. Skala V (2003) VTK for .NET platform, http://herakles.zcu.cz/research/vtk.net/
14. Taubin G (1991) Estimation of planar curves, surfaces, and nonplanar space curves defined by implicit equations with applications to edge and range image segmentation. IEEE Trans Pattern Anal Mach Intell 13(11):1115–1138
15. Tobor I, Reuter P, Schlick C (2004) Multi-scale reconstruction of implicit surfaces with attributes from large unorganized point sets. In: Proceedings of SMI, pp 19–30
16. Toledo S (2003) Taucs Version 2.2. http://www.tau.ac.il/~stoledo/taucs/
17. Walder C, Schölkopf B, Chapelle O (2006) Implicit surface modelling with a globally regularised basis of compact support. In: Proceedings of eurographics. Blackwell, Oxford, pp 635–644
18. Walder C, Schölkopf B, Chapelle O (2006) Implicit surface modelling with a globally regularised basis of compact support. Technical report. Max Planck Institute for Biological Cybernetics, Department of Empirical Inference, Tübingen, Germany, April 2006
19. Wendland H (1995) Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree. AICM 4:389–396

20. Savchenko V, Pasko A, Okunev O, Kunii TL (1995) Function representation of solids reconstructed from scattered surface points and contours. Comput Graph Forum 14(4):181–188
21. Ohtake Y, Belyaev A, Alexa M, Turk G, Seidel H-P (2003) Multi-level partition of unity implicits. ACM Trans Graph 22(3): 463–470. In: Proceedings of SIGGRAPH
22. Turk G, O'Brien JF (2002) Modelling with implicit surfaces that interpolate. ACM Trans Graph 21(4):855–873
23. Freytag M, Shapiro V, Tsukanov I (2006) Field modeling with sampled distances. Comput Aided Des 38(2):87–100