Contents lists available at ScienceDirect

Computers & Graphics

journal homepage: https://www.journals.elsevier.com/locate/cag

Multi-scale surface reconstruction based on a curvature-adaptive signed distance field

submission ID 185

ARTICLE INFO

Article history:

Keywords: Surface reconstruction, Adaptive signed distance field, Multi-scale B-splines, Principal curvature

ABSTRACT

This paper presents a multi-scale method for surface reconstruction from oriented point sets. The method is based on building and fitting an adaptive signed distance field. The adaptive signed distance field is built on an adaptive octree grid whose local grid interval is determined by the principal curvatures estimated on the input point set. In this way, scale-varying geometric details can be faithfully represented by the adaptive signed distance field. Next, a set of multi-scale B-spline basis functions are adopted to define the implicit function that globally and optimally fits the adaptive signed distance field. Because these basis functions are selected carefully, the fitting problem is reduced to a well-conditioned sparse linear system. As a result, a C^1-continuous field function is generated. The fitted field function is a good approximation of the signed distance field, and thus, its nonzero level sets can also approximate the offsets of the underlying surface well. Experimental results show that the proposed method can faithfully reconstruct crack-free adaptive triangular meshes from oriented point sets. Meanwhile, it is efficient in both running time and memory.

1. Introduction

Reconstructing surfaces from oriented point sets is a problem of great concern in fields such as reverse engineering, cultural heritage protection, and virtual reality. Because of the development of 3D scanning techniques and multi-view stereovision systems, dense oriented point sets have become one of the prevalent surface representations in computer graphics [1]. For the convenience of rendering and editing, point data are usually converted to 2D manifold surface forms, e.g., triangular meshes, via surface reconstruction algorithms.

The challenges in surface reconstruction mainly include the following two aspects: how to deal with imperfect data and how to reconstruct geometric details. In real-world applications, the acquired data are often characterized by noises, non-uniform sampling and missing data. Several approaches have been developed to address these data imperfections, e.g., local fitting for de-noising [2, 3, 4] and global fitting for hole filling [5, 6]. However, most of these methods have either direct or indirect filtering operations, which, without appropriate parameter controls, may lead to the over-smoothing of small details. To reconstruct these scale-varying geometric details, many approaches adapt the reconstruction resolution to the input data in some way. However, many of these decisions are based on the density of the sample points. An increased sample density is often generated by data redundancy rather than by shape complexity [7]. Accordingly, the adaptivity based on point density cannot always guarantee the reconstruction of scale-varying geometric details.

Surface reconstruction methods based on signed distance fields are common. Early methods often used uniform spatial grids for distance sampling and computed signed distance values via local optimization [8, 9]. Thus, these methods may not be robust or efficient for fine-detail reconstruction. Such methods can be improved by implementing adaptive spatial grids and global optimization [10, 11]. However, these methods often limit optimization to the discrete signed distance values on some spatial grids. Although a few methods adopt implicit functions for global fitting of the signed distance fields [5, 12], they usually produce an indicator function whose zero level set approximates the underlying surface while nonzero level sets may not necessarily approximate the offsets.

In this paper, we propose a multi-scale approach for surface reconstruction from an oriented point set. The method first builds an adaptive signed distance field and then fits the field globally using an implicit function. The adaptive signed distance field is built on an adaptive octree grid whose local resolution (or local grid interval) is determined by the principal curvatures estimated on the input point set. Therefore, the adaptive signed distance field can represent scale-varying geometric details by matching its resolution to shape complexity. The implicit function used for fitting is defined by a set of multi-scale B-spline basis functions. Because of the careful selection of these basis functions, the fitting problem is reduced to a well-conditioned sparse linear system. The fitting result is a C^1-continuous field function, which is a good approximation to the signed distance field; thus, its nonzero level sets also approximate the offsets of the underlying surface well. At the end of the proposed method, an octree-based isosurface extraction algorithm is employed to generate a crack-free adaptive triangular mesh. The contributions of our work can be summarized as follows:

- A multi-scale method is proposed for surface reconstruction from oriented point sets; it is based on building and fitting a curvature-adaptive signed distance field. The reconstructed surface is an adaptive crack-free triangular mesh.
- Scale-varying geometric details can be reconstructed by building the adaptive signed distance field on an adaptive octree grid whose local grid interval is determined by the principal curvatures estimated on the input point set.

2. Related Work

There is extensive work related to surface reconstruction from point data. Our discussion in this section covers only some closely related methods. The readers can refer to [13, 14] for a survey on the state of the art or to [15, 16] for comprehensive evaluations on recent representative algorithms.

Combinatorial Algorithms. The methods in this class typically produce an interpolating surface in the form of a triangulation that uses all or a subset of the input points as vertices. Classical computational geometry techniques such as Delaunay triangulation [17], Voronoi diagrams [18] and Alpha Shapes [19] are often adopted for this purpose. One well-known algorithm is the Cocone [20, 21, 22], which computes a piecewise linear approximation to a point sampled surface via a restricted Delaunay triangulation. Recently, a novel method based on dictionary learning was proposed in [23]. A good survey focusing on this class of methods can be found in [24, 25]. Most of these methods are combinatorial in nature and optimize only the topological connections of the points without changing their positional properties. Thus, jagged surfaces are likely to be produced in the presence of noise, and erroneous triangulations often occur in regions with unevenly sampled data.

Implicit Methods. To address data imperfections, implicit surface reconstruction methods generate an approximate surface near the input point set. The general framework is first creating an implicit function and then extracting its zero level set.

One class of popular local implicit methods is that of Moving Least-Squares (MLS) [2, 3, 4, 26], which performs locally weighted least-squares fitting of a point set. MLS methods are also widely used for data de-noising because of their efficient local solution. The proposed method also benefits from an adaptive MLS [27, 28] for pointwise curvature estimation. The implicit function defined by MLS methods are usually local and dependent on the input data, restricting their capacity for hole filling. To overcome this problem, Ohtake et al. [29] defined a global and data-independent implicit function via an octree-based multilevel blending of the local representations, namely, the Multi-level Partition of Unity (MPU) implicit function. However, the blending is heuristic without introducing a global optimization.

To reconstruct a closed surface, it is preferable to use a global implicit method. Many such methods prescribe a function space for the implicit functions, and the possible types of basis functions are radial basis functions [5, 30], wavelets [31], trigonometric polynomials [6], and B-Splines [32, 33]. The Poisson method [34, 35], which formulates the surface reconstruction problem as a Poisson equation, is one of the most widely used methods by the current research community because of its robustness to data imperfections. However, the Poisson method has the possibility to smooth geometric details [11]. One possible reason is that its reconstruction resolution is adaptive to point density rather than shape complexity, and the maximum resolution (i.e., maximum octree depth) is a subjective parameter that must be specified by users. Thus, too-small depths may cause a loss of geometric detail, whereas too-large depths may lead to an excessive computational burden and data redundancy.

Signed Distance Field. Surface reconstruction methods based on signed distance fields can be regarded as a special class of implicit methods, whose implicit functions are usually discrete signed distance fields. Hoppe et al. [8] built a signed distance field on a uniform spatial grid with signed distance values computed by local plane projection. The Volumetric Range Image-Processing (VRIP) method [9], which is extensively used in the Digital Michelangelo project [36], also uses uniform spatial grids, but the signed distance values are calculated by averaging the "ray-casting distances" from multiple overlapping scans. Mullen et al. [10] transformed an unsigned distance field to a signed field. They used an adaptive spatial grid, but the global optimization solution relies on a Delaunay triangulation for domain discretization. Calakli and Taubin [11] constructed a smoothed signed distance field on an adaptive octree grid. Their method needs no explicit signed distance sampling, and the signed distance values are directly derived using global optimization. These methods generally produces discrete representations of signed distance fields. Recently, Sharma et al. [37] created a C^1-continuous signed distance function for surface reconstruction from a set of unorganized planar cross-sections. Pan et al. [12] transformed a signed distance field into a phase field (values are in $[-1, 1]$) on a uniform grid. They used implicit hierarchical B-splines to fit both the phase field and point data, reducing many basis functions.

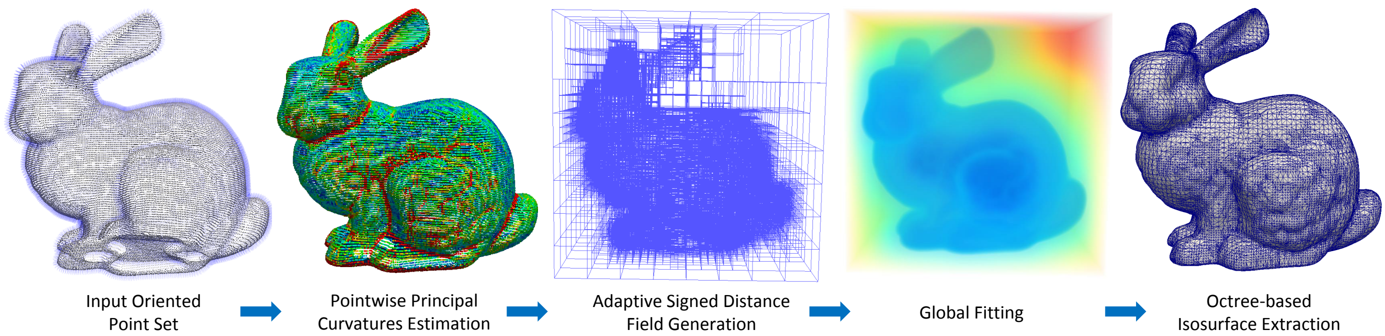


Fig. 1. Algorithm overview.

Problem of Geometric Detail Scales. Recently, the scale problem in surface reconstruction has attracted the attentions of researchers. Fuhrmann et al. [38] noted that overlapping depth maps should be fused at compatible scales. In their subsequent work [7], the floating scale implicit function was introduced for reconstructing a surface with different scales of geometric details. Mücke et al. [39] generated a 3D confidence map by splatting a Gaussian function for each input sample into a uniform spatial grid, where the Gaussian standard deviation is set to half of the scale of the sample; the final surface is extracted from the grid via a global graph-cut algorithm. In these methods, the scale information of the geometric details is usually a part of input data, which is obtained from the data acquisition process.

3. Algorithm Overview

Figure 1 shows the overview of the proposed algorithm. The algorithm consists of four major steps. First, we estimate principal curvatures of each input point via an adaptive MLS algorithm. Second, we generate an adaptive signed distance field using an adaptive octree grid whose local resolution is determined by the estimated curvatures. Third, the field is globally fitted by an implicit function, which is defined by a set of multiscale B-spline basis functions. Fourth, an octree-based isosurface extraction algorithm is employed to generate a crack-free adaptive triangular mesh. Note that the proposed algorithm requires the input of oriented points, i.e., points associated with oriented normals. For point data without normal information, we adopt the Principal Component Analysis (PCA) approach and the Minimum Spanning Tree algorithm proposed in [8] for normal estimation and orientation.

3.1. Pointwise Principal Curvatures Estimation

Because there is no parametric or implicit representation of the input point set so far, it is necessary to perform local or global fitting before curvature estimation. Here, the MLS fitting approach [2] is adopted for the input points. Then, the principle curvatures at each point are evaluated analytically based on the locally fitted MLS surface [40]. There are two principal curvatures estimated for each point, but only the one with the maximum absolute value is recorded for that point. Meanwhile, the fitted MLS surface at each point is also recorded for subsequent signed distance evaluations.

Generally, the MLS algorithm requires users to specify a radius h for its smooth kernel function. A fixed radius may lead to poorly-fitted results if the input point set is unevenly sampled. To overcome this problem, we determine the radius adaptively based on local point density [27]. The local point density ρ_i at a point $\mathbf{p}_i$ can be estimated by determining a sphere with minimum radius r_i centred at $\mathbf{p}_i$ that contains the k-nearest neighbours to $\mathbf{p}_i$. By approximating the intersection of this sphere and the underlying surface as a disc, ρ_i can be defined as

$$\rho_i = \frac{k}{\pi r_i^2}. \tag{1}$$

Then, the adaptive radius h_i for point $\mathbf{p}_i$ is defined as

$$h_i = \frac{h}{\rho_i}, \tag{2}$$

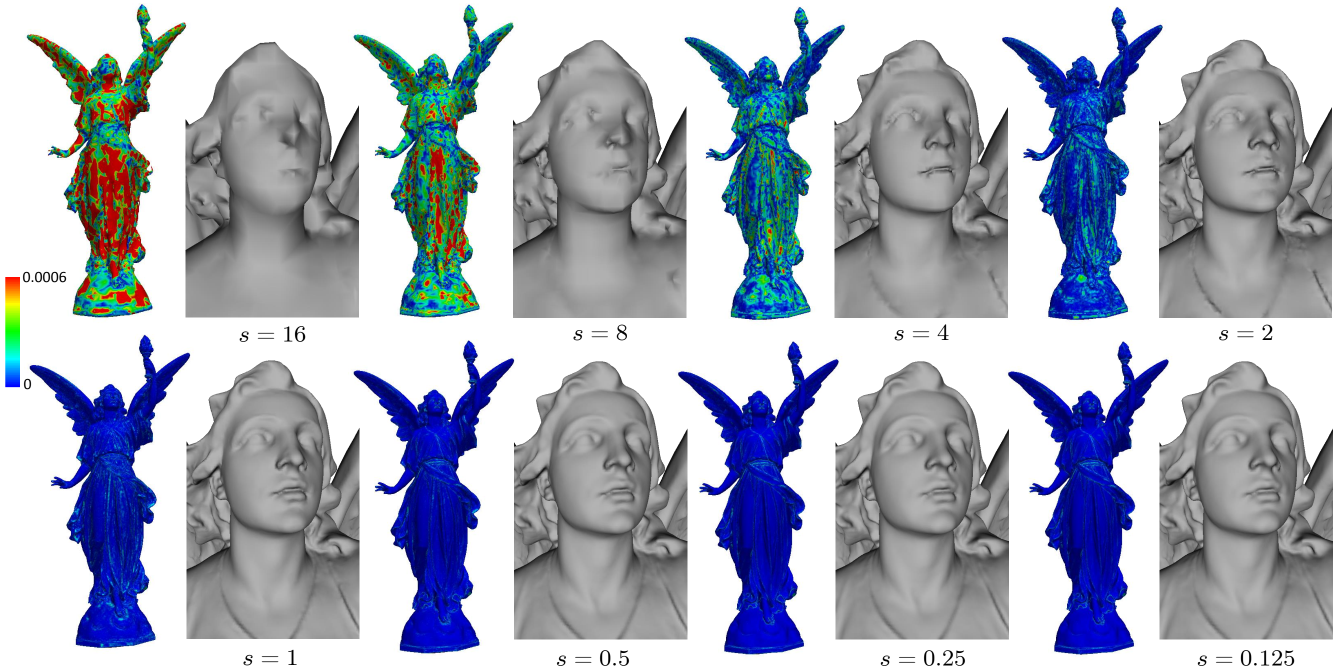
where h is a base value of the radius. In our experiments, we use quadratic polynomial fitting and set $k = 20$ and $h = k/r_{median}$, where r_{median} is the median of radius r_i among all input samples. It seems that the point density-based adaptivity may cause the loss of detail in surface reconstruction as discussed in the related work. In reality, this case will not occur because introducing the adaptivity here solves the problem of non-uniform sampling instead of the problem of reconstructing scale-varying geometric details. Moreover, the MLS fitting is applied only for curvature estimation, not for data de-noising. Therefore, the original point data remain unchanged.

3.2. Adaptive Signed Distance Field Generation

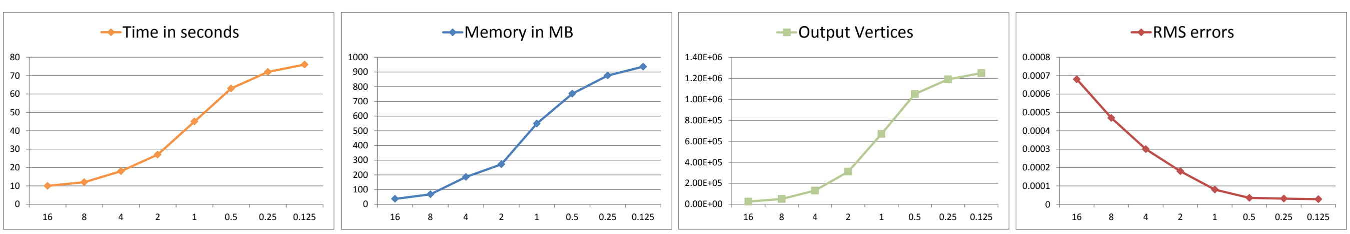
The adaptive signed distance field is generated via two steps. First, an adaptive octree grid is constructed whose local grid interval is determined by the curvatures estimated on the input point set. Second, the signed distance value at each corner of the octree grid is estimated using an MLS surface projection.

3.2.1. Curvature-based Adaptive Octree Construction

Generally, an adaptive octree is initialized as a bounding box (usually a cube) that contains all points and is then recursively subdivided to encompass the point set more and more tightly.



(a) Visualization of the reconstruction results of the Lucy model. The colour-coding visualizations show the reconstruction errors.



(b) Statistics of performance and RMS reconstruction error. The RMS reconstruction error begins to converge when s is smaller than 0.5.

Fig. 2. Reconstruction of the Lucy model with different values of the scale parameter s.

Different adaptive strategies have their respective termination conditions for octree node subdivision. For example, the point density-based adaptive strategy terminates to subdivide an octree node only if the number of points in the node is smaller than a pre-specified threshold.

Our curvature-based adaptive strategy is inspired by the Nyquist-Shannon sampling theorem. The theorem states that the maximum sampling interval that can recover all components of a signal is half of the period interval of the highest frequency component. Assuming that the geometric details are the signal components of the surface, we conjecture that the maximum spatial sampling interval that can reconstruct all details of a surface is approximately half of the minimal curvature radius estimated on the surface. Since we perform signed distance sampling on an octree grid, the spatial sampling interval is just the grid interval. Because the octree can partition the input point set into a set of local parts, we can adapt the local grid interval to the local minimal curvature radius estimated on the input point set. In this way, an adaptive octree grid can be constructed according to the following adaptive strategy: *An octree node is terminated to be subdivided only if its width (i.e., local grid interval) is not larger than half of the minimal curvature radius of all points inside it.*

Because the adaptive strategy is the major contribution of the proposed method, it is more convincing to verify its correctness via experiments on real-world data. To this end, we conduct an experiment using the Lucy model (30M points). A scale parameter s is introduced to adjust the adaptive strategy. Supposing r_{min} is the minimal curvature radius of all points inside an octree node, the adjusted adaptive strategy terminates to subdivide the octree node only if its width is not larger than sr_{min}. We build adaptive signed distance fields for the Lucy model using different values of s, and then reconstruct the surfaces using the fitting methodology and the isosurface extraction algorithm described in Sections 3.3 and 3.4, respectively. Figure 2 shows the reconstruction results and the statistics of performance and RMS error. As seen in these images and diagrams, the accuracy of the reconstructed surface gradually increases while decreasing the value of s, and the RMS error begins to converge when $s = 0.5$. The experiment indicates that setting the local grid interval to half of the local minimal curvature radius is sufficient to produce results with acceptable precision, and the computational costs in terms of time and space are also moderate. Thus, in the subsequent experiments, half of the local minimal curvature radius, i.e, $s = 0.5$, is always adopted as the criterion for building the adaptive signed distance field.

3.2.2. Signed Distance Computation

We use MLS surface projections to compute the signed distance values. For each corner point $\mathbf{q}_j$ on the adaptive octree grid, we find its nearest point $\mathbf{p}_i$ among the input points. The signed distance value d_j at $\mathbf{q}_j$ is calculated by projecting $\mathbf{q}_j$ to the MLS surface at $\mathbf{p}_i$, which was previously fitted for curvature estimation. The projection direction is parallel to the normal $\mathbf{n}_i$ of $\mathbf{p}_i$. Supposing the projection footprint is $\mathbf{p}'_i$ with normal $\mathbf{n}'_i$ on the MLS surface, d_j can be defined as

$$d_j = (\mathbf{q}_j - \mathbf{p}'_i) \cdot \mathbf{n}'_i, \tag{3}$$

where $\mathbf{n}'_i$ is normalized. In general, the line segment connecting $\mathbf{q}_j$ and $\mathbf{p}_i$ is nearly perpendicular to the MLS surface at $\mathbf{p}_i$. However, for some boundary cases, the orthogonality will not hold. Thus, we introduce a confidence w_j for d_j, which is defined as

$$w_j = \left| \frac{\mathbf{q}_j - \mathbf{p}_i}{||\mathbf{q}_j - \mathbf{p}_i||} \cdot \mathbf{n}'_i \right|, \tag{4}$$

which is simply the absolute value of the cosine of the angle between $\mathbf{q}_j - \mathbf{p}_i$ and $\mathbf{n}'_i$.

3.3. Global Fitting

In this section, an implicit function defined by a set of multi-scale B-spline basis functions is introduced to globally and optimally fit the adaptive signed distance field, the input points and the related normals.

3.3.1. Formulation

The global fitting of the adaptive signed distance field can be formulated as a functional minimization. It seeks to find the implicit function $f(\mathbf{x})$ that minimizes the following energy:

$$E_D(f) = \frac{1}{m} \sum_{j=1}^{m} w_j \left(f(\mathbf{q}_j) - d_j \right)^2, \tag{5}$$

where $\mathbf{q}_j$ is an octree corner point with signed distance value d_j and confidence w_j, and m is the total number of octree corners.

However, a straightforward minimization of E_D may lead to an over-fitting problem such that the resulting implicit function $f(\mathbf{x})$ has high-frequency oscillations. To overcome this problem, we introduce the following regularization term:

$$E_R(f) = \frac{1}{|V|} \int_V ||Hf(\mathbf{x})||^2 d\mathbf{x}, \tag{6}$$

where $Hf(\mathbf{x})$ is the Hessian matrix of $f(\mathbf{x})$, i.e., the 3×3 matrix of second partial derivatives of $f(\mathbf{x})$, and the norm of the matrix is the Frobenius norm. The integral is calculated over the volume V, which encompasses the surface to be reconstructed, and $|V| = \int_V d\mathbf{x}$ is the measure of this volume. In our implementation, V is set to the octree volume that contains the adaptive signed distance field.

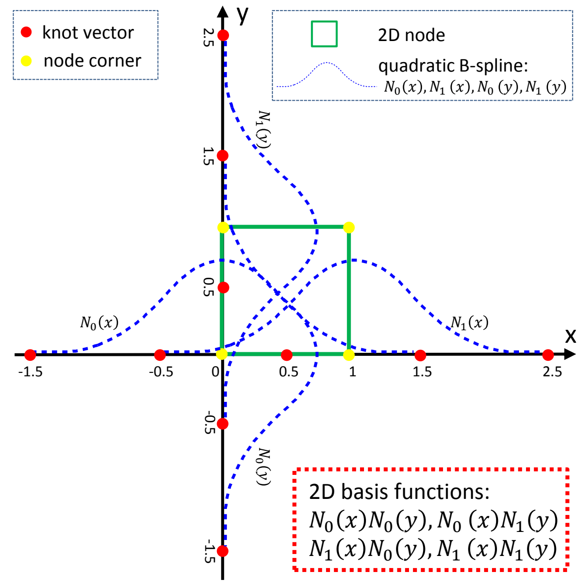


Fig. 3. The layout of a 2D node and the knot vectors of four B-spline basis functions centred at its corners. The width of the node is normalized to 1.

In addition, to improve the fitting precision for the original input data, we add two extra energy terms

$$E_P(f) = \frac{1}{n} \sum_{i=1}^{n} f^2(\mathbf{p}_i) \quad \text{and} \quad E_N(f) = \frac{1}{n} \sum_{i=1}^{n} ||\nabla f(\mathbf{p}_i) - \mathbf{n}_i||^2, \tag{7}$$

where E_P and E_N are used for fitting the input points and their normals, respectively; $\mathbf{p}_i$ is an input point with normal $\mathbf{n}_i$; and n is the total number of input points.

In this way, the total energy functional $E(f)$ used for global fitting becomes a weighted summation of the energy and regularization terms

$$E(f) = E_D(f) + \lambda_R E_R(f) + \lambda_P E_P(f) + \lambda_N E_N(f), \tag{8}$$

where λ_R, λ_P, and λ_N are the parameters that adjust the weights of E_R, E_P, E_N, respectively, relative to E_D. In our experiments, we set λ_R, λ_P, λ_N to 0.01, 1.0, 1.0, respectively.

3.3.2. Function Space

Because the signed distance field is built on an adaptive octree grid, we naturally think that the same spatial structure can be used for defining the function space of $f(\mathbf{x})$. In reality, our definition is somewhat similar to that provided by the Poisson method [34, 35], but differences do exist. In the Poisson method, one octree node corresponds to one basis function, whereas in the proposed method, one octree corner corresponds to one or multiple basis functions.

The definition of the function space is described as follows. For every octree corner c that belongs to an octree node at depth d, a basis function B_c is defined, which is centred at c and whose support is stretched by the width of a depth-d octree node:

$$B_c(\mathbf{x}) \equiv B\left(\frac{\mathbf{x} - c.\mathbf{q}}{w_d} \right), \tag{9}$$

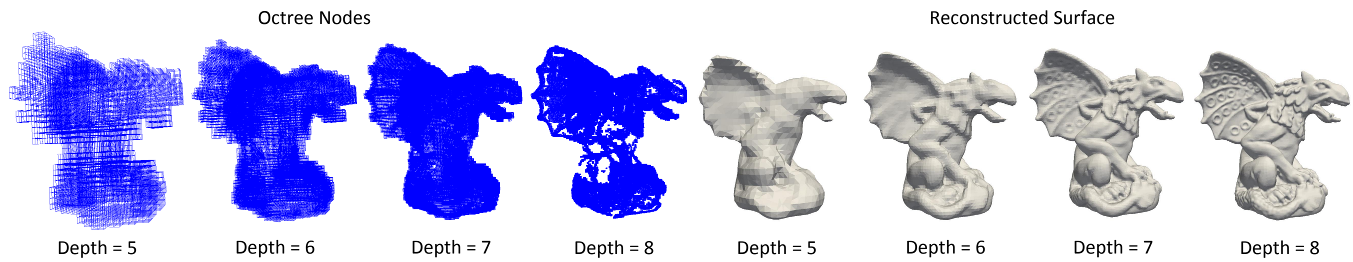


Fig. 4. Multi-scale reconstruction process of the Gargoyle model using a multigrid approach. The left shows the octree nodes at different depths, whose corners corresponds to the B-splines and the signed distance samples used at the corresponding depths. The right shows the reconstructed surfaces.

where $c.\mathbf{q}$ is the position of c and w_d is the width of a depth-d octree node; $B : \mathbb{R}^3 \rightarrow \mathbb{R}$ is a uniform triquadratic B-spline basis, which is supported on the domain $[-1.5, 1.5]^3$. Note that an octree corner may be shared by octree nodes at different depths; thus, it may correspond to multiple basis functions defined at different depths. We declare that corners belonging to depth-d nodes are called depth-d corners. Corners at different depths may coincide in position.

Figure 3 is an illustration of the layout of a 2D node and the knot vectors of four B-spline basis functions centred at its corners. The purpose of such a layout is to ensure that there is at least one signed distance sample, i.e., corner point, located in the centre of each B-spline knot box (except for the boundary ones). Empirically, the layout is apt to generate a well-conditioned linear system for the solution of the B-spline fitting.

3.3.3. Solution to the Energy Functional

Having defined the function space, $f(\mathbf{x})$ is expressed as a linear sum of the basis functions $\{B_c\}$, where $f(\mathbf{x}) = \sum v_c B_c(\mathbf{x})$. Then, the energy functional $E(f)$ is converted into a quadratic function of the basis function coefficients $\{v_c\}$. Minimizing the quadratic function yields the following linear system:

$$L\mathbf{v} = \mathbf{b} \quad \text{with} \quad \begin{cases} L = L_D + \lambda_R L_R + \lambda_P L_P + \lambda_N L_N \\ \mathbf{b} = \mathbf{b}_D + \lambda_N \mathbf{b}_N \end{cases}, \qquad (10)$$

where $\mathbf{v}$ is the vector consisting of the coefficients $\{v_c\}$; L_D, L_R, L_P and L_N are the constraint matrices derived from the minimization of E_D, E_R, E_P and E_N respectively, and $\mathbf{b}_D$, $\mathbf{b}_N$ are the corresponding constraint vectors. Note that no constraint vector is generated for E_R and E_P because they introduce only homogeneous constraints. The derivation of L_D, L_R, L_P, L_N, $\mathbf{b}_D$, $\mathbf{b}_N$ are given in Appendix A.

Directly solving the above linear system consumes both time and space. Due to the multi-scale structure of the basis functions, the linear system can be solved using a multigrid approach. A multigrid approach solve a linear system progressively from coarse scale to fine scale, using the solutions at coarser scales to update the residuals at finer scales.

Clustering the basis functions according to their scales, which are related to depths of the octree, and for each depth $d = 0, \ldots, d_{max}$, a linear system is defined as follows:

$$L^d \mathbf{v}^d = \mathbf{b}^d \quad \text{with} \quad \begin{cases} L^d = L_D^d + 2^d \lambda_R L_R^d + \lambda_P L_P^d + \lambda_N L_N^d \\ \mathbf{b}^d = \mathbf{b}_D^d + \lambda_N \mathbf{b}_N^d \end{cases}, \qquad (11)$$

where $\mathbf{v}^d$ is the vector consisting of the coefficients of the basis functions at depth d; L_D^d, L_R^d, L_P^d, L_N^d, $\mathbf{b}_D^d$, $\mathbf{b}_N^d$ are the similar matrices in Equation 10. These matrices can be derived using the same formulas described in Appendix A, where the only modification is that the function space is reduced to the set of basis functions at depth d. Note that λ_R is scaled by 2^d to maintain the scale-independence of the energy minimization at different depths. The reason for this is that E_R^d is a volume integral (cubic in metric), whereas E_D^d, E_P^d and E_N^d are the sums of squared distances (quadratic in metric).

Because the basis functions at coarse scales are smooth, it is unnecessary to constrain them at high-resolution samples. Fortunately, the signed distance values are sampled at the octree corners, which have an inherent multi-resolution structure. Thus, L_D^d and $\mathbf{b}_D^d$ can be derived by considering only the signed distance samples (i.e., octree corners) at depth d, as shown in Figure 4. The input points do not have an inherent multi-resolution structure; therefore, we apply a hierarchical clustering operation. The clustering approach is similar to that employed in [35], which clusters the points inside each octree node at each depth into an averaged position. Specifically, for every octree node o at depth d, the points in the node are averaged to a point $\mathbf{p}_o^d$ with normal $\mathbf{n}_o^d$. The clustered points $\{\mathbf{p}_o^d\}$ and normals $\{\mathbf{n}_o^d\}$ are used to derive L_P^d, L_N^d and $\mathbf{b}_N^d$. In this way, L^d can be derived using both depth-d basis functions and depth-d samples (including signed distances, clustered points and normals).

Algorithm 1 Multigrid Solver

for $d \in \{0, \ldots, d_{max}\}$ **do**	*Iterate from coarse to fine*
for $d' \in \{0, \ldots, d-1\}$ **do**	*Update residuals by*
$\mathbf{b}^d = \mathbf{b}^d - L^{dd'} \mathbf{v}^{d'}$	*solution from coarse levels*
Solve $L^d \mathbf{v}^d = \mathbf{b}^d$	*Solve in current level*

The pseudocode of the multigrid solver for solving $\mathbf{v}$ is given in Algorithm 1. Here, $L^{dd'}$ is a constraint matrix that has a similar derivation to L^d; the only difference is that $L^{dd'}$ is derived using the basis functions at depth d' but the samples at depth d, whereas L^d is derived using both of them at depth d. For each basis function, there are at most $5^3 - 1 = 124$ other basis functions at the same depth that overlap it. Thus, according to the

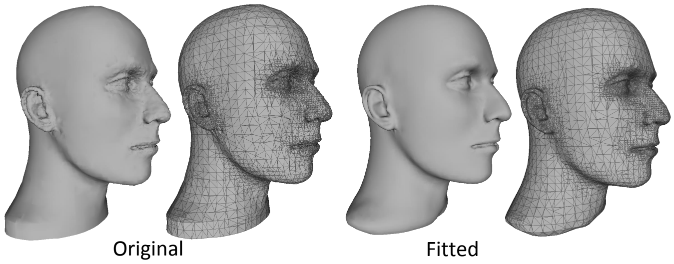


Fig. 5. Octree-based isosurface extraction from the original adaptive signed distance field and the fitted field of the Mannequin model.

formulas given in Appendix A, L^d is a sparse symmetric matrix with at most 125 nonzero entries in each column (row). Figure 4 shows the multi-scale reconstruction process of the Gargoyle model using the multigrid approach.

3.4. Octree-based Isosurface Extraction

The global fitting produces a C^1-continuous field function $f(\mathbf{x})$; therefore, isosurface extraction algorithms such as Marching Cubes can be used straightforwardly. To improve efficiency, we adopt the octree-based isosurface extraction algorithm proposed in [41]. The algorithm can extract crack-free triangle meshes on any octree grid whose corners are assigned scalars. We have already constructed such an octree grid for building the adaptive signed distance filed, and it is very suitable for isosurface extraction because its resolution is adaptive to the curvatures of the underlying surface, which is conducive to the preservation of details in the extracted meshes. To extract smooth meshes, the leaf nodes which intersect the isosurface and whose depths are smaller than 5 are forced to be subdivided to depth 5 before isosurface extraction. The intersection is checked using the scalars recorded on the node corners.

Figure 5 shows the surfaces extracted from the original adaptive signed distance field and the fitted field (whose scalars are updated by $f(\mathbf{x})$) of the Mannequin model. As seen in these images, the extracted meshes have adaptive vertex densities that match the curvatures, and the surface extracted from the fitted field is smoother than the one extracted from the original field.

4. Results and Discussions

The proposed algorithm is implemented in C++ using OpenMP for multi-threaded parallelization. All experiments were run on a desktop with a quad-core Intel Core i7 processor and 16GB of RAM. The API provided by the FLANN library [42] is used for k-nearest point searching in curvature estimation and signed distance computation. The conjugate gradient solver provided by the Eigen library [43] is used to solve the linear system at each multigrid level. Except for the construction of the octree, most of the operations involved in the proposed algorithm can be parallelized.

The proposed algorithm is evaluated from the aspects of accuracy, running time, memory consumption, etc. Several related state-of-the-art methods are also implemented for comparison, including the wavelet reconstruction of Manson et al. [31], the smooth signed distance (SSD) reconstruction of Calakli et al. [11], the screened Poisson (SP) reconstruction of Kazhdan et al. [35], and the Implicit Hierarchical B-splines (IHB) reconstruction of Pan et al [12]. They are all implemented in C++ as executable programs, except for the IHB method, which is implemented in MATLAB. In addition, the proposed approach without the step of global fitting is also compared. This unfitted version can be regarded as a method purely based on an adaptive distance field.

The proposed method can automatically determine the maximum depth d_{max} of the basis functions according to the minimal curvature radius of all input points. The other four algorithms treat d_{max} as a parameter to be specified. For fairness of comparison, the parameters d_{max} of these algorithms are set to be the same as that of the proposed method, except for the IHB reconstruction, which can support depth 6 at most because of the limitation of memory. Other parameters of these methods are set to the default values provided by their implementation instructions.

4.1. Accuracy

Real Scanning Data. To evaluate the accuracy of the different algorithms on real-world data, we gathered several well-known scanned datasets, which are the Asian Dragon model (3.6M points), the Lion model (0.6M points), the Neptune model (2.0M points) and the Ramesses model (0.8M points). All of the scanned models contain scale-varying geometric details.

Figure 6 shows the reconstruction results for these models. The cross-section visualizations show the deviation between the reconstructed surfaces and the input point data. Overall, all six methods can reconstruct the models faithfully. After carefully observing the reconstructed surfaces and the cross-section visualizations, we found that the wavelet reconstruction of the Lion model has apparent derivative discontinuities, and the IHB reconstruction of the Asian Dragon and Neptune model smoothly filtered the geometric details. For the Lion model, weak noises are introduced into the surfaces reconstructed via the SP method, the SSD method and the unfitted method, which can be observed by zooming into the head of the Lion model. The proposed method can produce a relatively faithful surface reconstruction for all four models.

Figure 7 shows quantitative comparisons across all datasets, in the form of RMS errors, measured by the distances from the input points to the reconstructed surfaces. As the figure indicates, the RMS error of the proposed method are always the smallest among five methods for the Dragon, Neptune and Ramesses model. For the Lion model, the proposed method can produce comparable results with those of the SSD method, the SP method and the unfitted method.

Synthetic Point Data. To check the ability to address noise, non-uniform sampling and missing data, we evaluate the proposed method on two sets of synthetic point data, which are generated from two ground-truth models: the Armadillo model and the Horse model. Different approaches are adopted to generate point data for these two models. For the Armadillo model, we first randomly sample 4×10^5 points on the model and then cut these points into four parts as shown in Figure 9; the final

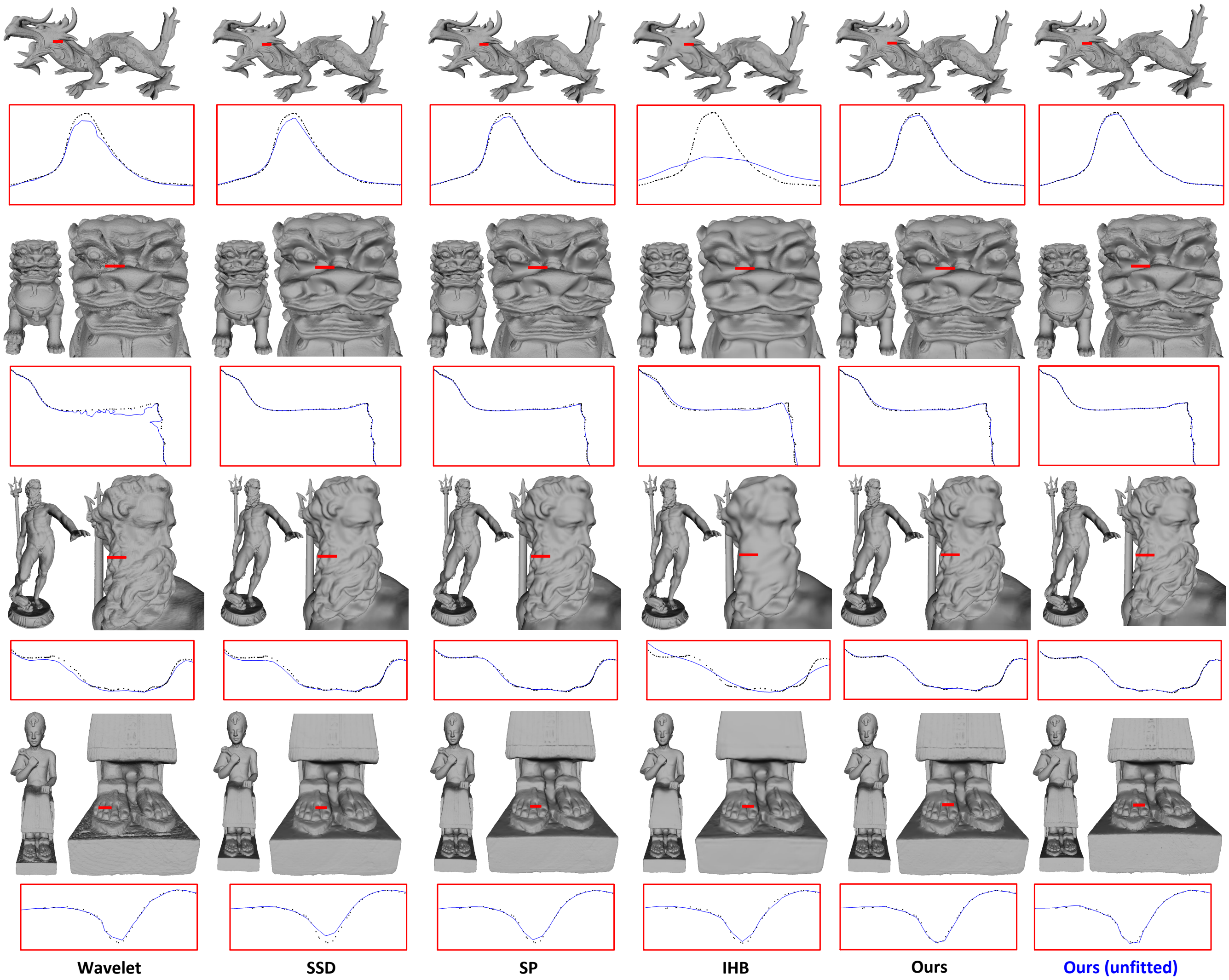


Fig. 6. Reconstruction of the real scanned data. Top to bottom with the automatically determined maximum depth d_{max}: Asian Dragon (11), Lion (10), Neptune (11), and Ramesses (10), except for d_{max} = 6 in the IHB method for all models. The 2D visualizations under the reconstructed models show cross-sections of the reconstructed surfaces and the input point data.

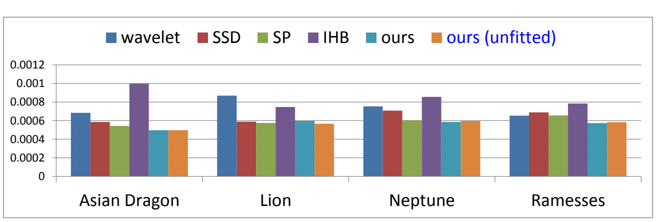


Fig. 7. RMS errors of the reconstruction of the real scanning data. The errors are measured by the distances from the input data to the reconstructed surfaces and normalized by the bounding box diagonal of the input models.

point set is produced by randomly selecting 1, 1/2, 1/4 and 1/8 of the points from each of the four parts. For the Horse model, we first randomly sample 2×10^5 points on the model and then remove some of points surrounding the waist and on the leg of the horse. Finally, we add Gaussian random noise to both point sets with a standard deviation $\sigma = 0.002$.

Figure 8 shows the reconstruction results of the synthetic point datasets. It also shows the reconstruction errors relative to the ground-truth models using the colour-coded visualizations. Observing these results, we find that the wavelet method is somewhat sensitive to non-uniform sampling such that discontinuous surface sheets are generated in the 1/4 and 1/8 sampled regions of the Armadillo model. Furthermore, the reconstruction of the Horse model by the wavelet method is also unacceptable because of the unsmoothness in the part of the missing data. The other methods are insensitive to non-uniform sampling and can generate smooth surface transitions to fill in the incomplete regions. For the Armadillo model, the IHB method produces a relatively larger reconstruction error in regions with large curvatures, such as regions near the fingers and ears. For the Horse model, the surface reconstructed by the SP method has a very slight shrink in the waist part, where there is no point data. The unfitted method introduces apparent noises for both models.

Figure 9 plots the RMS errors and the maximum errors of the reconstruction results of the synthetic point data. The errors are measured bidirectionally between the ground truth model and

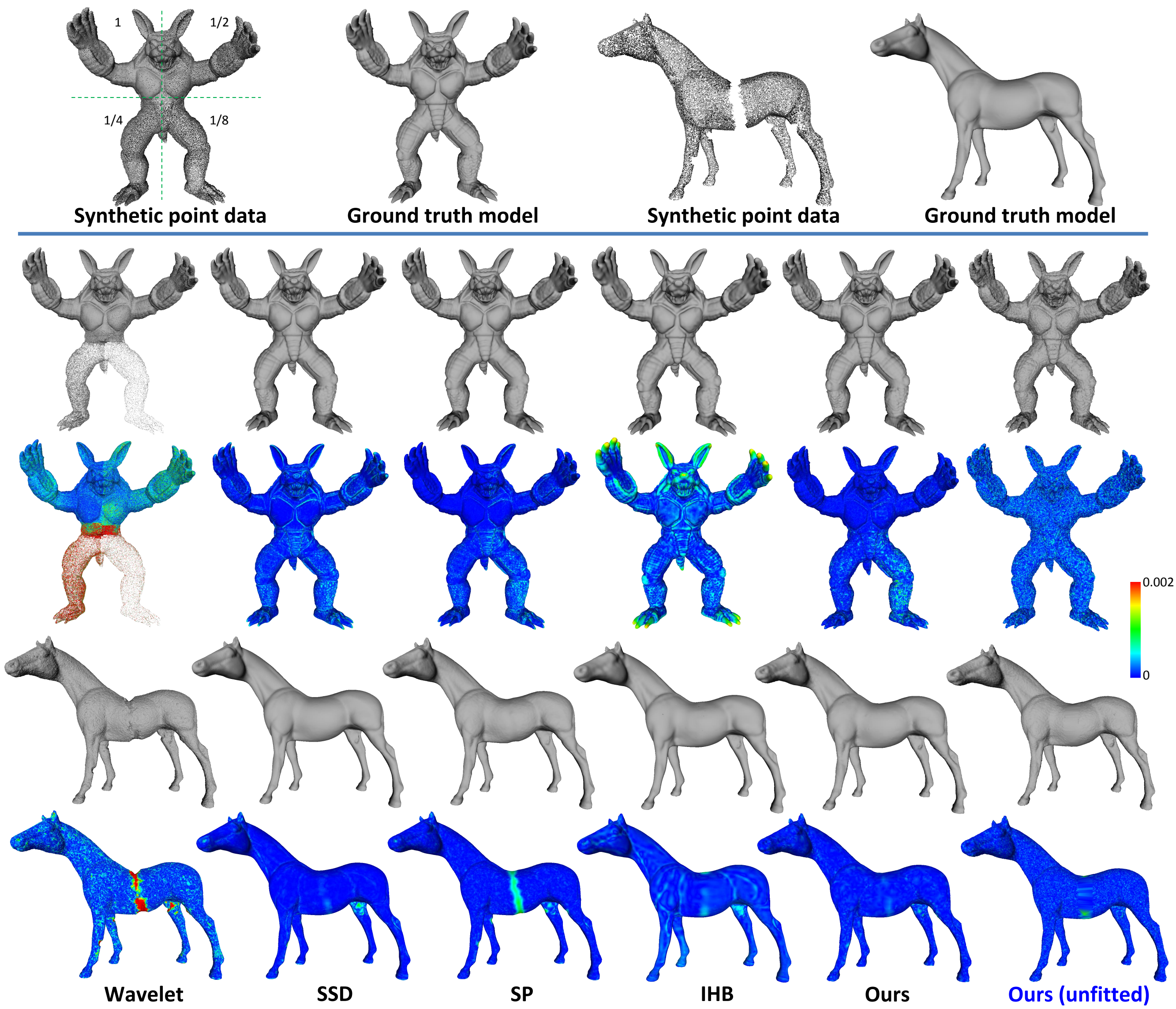


Fig. 8. Reconstruction of the synthetic point data generated from the Armadillo model (d_{max} = 9) and the Horse model (d_{max} = 9), except for d_{max} = 6 in the IHB method. The colour-coding visualizations indicate the reconstruction errors, which are measured by the distances from the reconstructed meshes to the ground truth models.

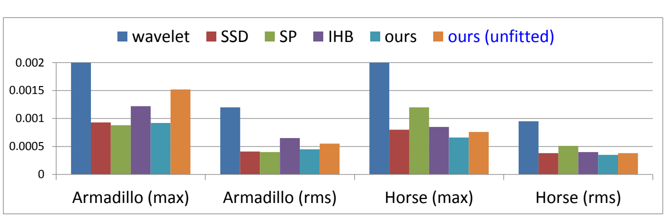


Fig. 9. Maximum errors and RMS errors of the reconstruction of the synthetic point data. The errors are measured bidirectionally between the reconstructed surfaces and the ground truth models.

the reconstructed surface using the Metro tool [44]. As in the case of real scanning data, the accuracy of the proposed method is comparable to or better than the other five methods.

4.2. *Computational Efficiency*

Table 1 shows the running time, memory usage and number of output vertices of all of the algorithms in the previous experiments. The unfitted version of the proposed method is the fastest one because it does not need to compute an implicit function. The wavelet method is the second fastest and requires the least memory because it uses compactly supported orthogonal basis functions, making the algorithm compute the implicit function coefficients directly through integration without an explicit solution for a linear system. In contrast, the other four algorithms require the solution for a global linear system because of the use of non-orthogonal basis functions. Both the proposed method and the SP method benefit from a similar multigrid framework, which can speed up the solution of the global linear system. However, the SSD method and the IHB method solve the global linear system directly; thus, their costs

Table 1. Runtime performance for the reconstruction of the real scanning data and the synthetic point data.

Model	Time in seconds					Memory in MB					Output Vertices $\times 10^6$				
	Wavelet	SSD	SP	IHB	ours/unfitted	Wavelet	SSD	SP	IHB	ours/unfitted	Wavelet	SSD	SP	IHB	ours/unfitted
Asian Dragon	23	279	102	329	55/15	54	2112	1054	2540	731/257	2.22	1.23	1.25	1.73	0.91/0.91
Lion	29	286	123	168	38/12	60	2202	1160	1520	431/144	2.82	1.56	1.57	1.16	0.56/0.57
Neptune	20	160	77	289	28/8	38	1871	937	1873	386/103	3.21	0.94	0.94	0.69	0.40/0.40
Ramesses	15	278	128	210	35/11	56	2080	1046	2071	484/152	4.41	1.25	1.25	1.06	0.69/0.69
Armadillo	6	45	24	55	13/4	27	811	393	973	295/93	1.41	0.31	0.31	0.32	0.25/0.26
Horse	4	20	8	28	5/2	11	693	281	491	194/67	0.8	0.13	0.13	0.13	0.08/0.08

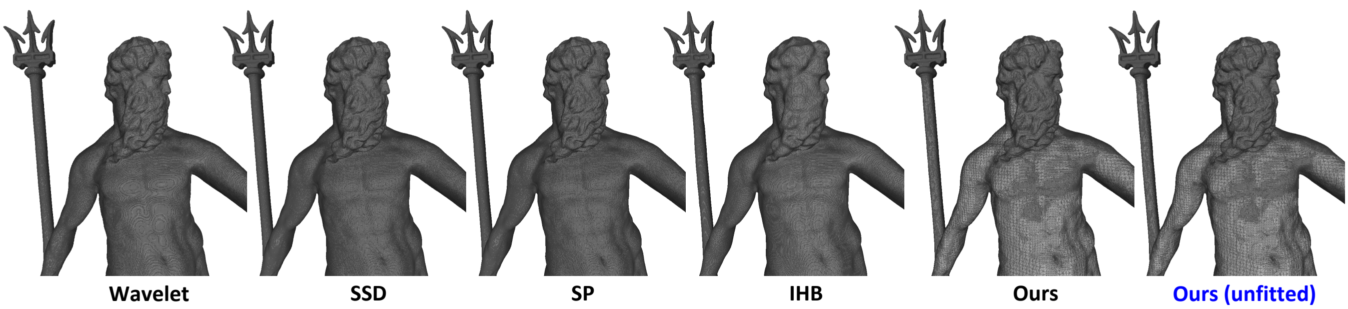


Fig. 10. Visualizations of the mesh wireframe of the reconstructed surfaces of the Neptune model. The proposed method generates an adaptive mesh that has a relatively smaller number of vertices.

Table 2. Number of octree nodes constructed in the experiments.

$\times 10^6$	Asian Dragon	Lion	Neptune	Ramesses	Armadillo	Horse
SSD/SP	7.64	8.90	5.78	7.74	1.67	1.97
ours	1.89	0.98	0.88	1.35	0.25	0.14

in terms of running time and memory are both larger than those of the proposed method and the SP method.

Except for the wavelet method and the unfitted method, the proposed method has a shorter running time and a smaller memory usage than the other three methods. The reason for this is that the proposed method constructs a sparser adaptive octree. Table 2 shows the number of the octree nodes constructed by the SSD method (the same as the SP method) and the proposed method. It is apparent that the proposed method constructs octrees with less nodes and therefore, fewer basis functions are required and fewer coefficients must be solved. The sparsity is attributable to the curvature-based adaptivity for constructing the octree. The adaptivity allows the proposed method to use fewer basis functions in regions with small curvature and more basis functions in regions with large curvature. Moreover, the octree-based isosurface extraction allows the generation of adaptive meshes, which have fewer vertices than those obtained by the other methods (see Table 1 and Figure 6). Although the number of vertices is reduced, the accuracy of our reconstructed meshes is comparable to those generated by the other methods, as demonstrated in Section 4.1.

4.3. Nonzero Level Sets

To check the ability of the nonzero level sets of the fitted field function $f(\mathbf{x})$ to approximate offset surfaces, we generated a set of nonzero level sets using the fitted field function of the Dragon model, as shown in Figure 11. As seen in these images, these nonzero level sets well approximate the real offsets with the corresponding signed distance values. This also indicates that $f(\mathbf{x})$ is a good approximation of the signed distance field; this property can be useful in applications such as collision detection, volume rendering and sculpting [45].

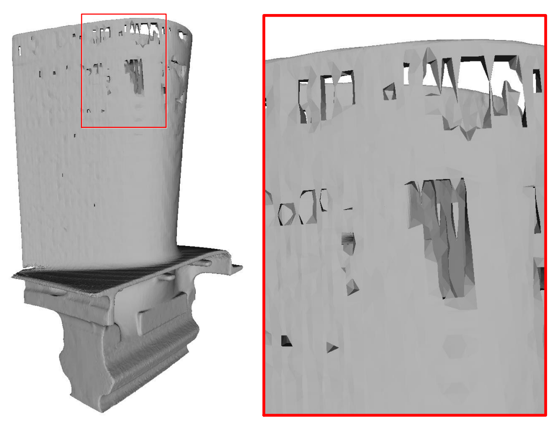


Fig. 12. A failure example of reconstructing the Blade model. Holes appears in the very thin and nearly planar parts of the model.

4.4. Limitations

Because of global fitting, our approach can handle a certain degree of noise, non-uniform sampling and missing data. However, if the input data contain severe noise, outliers or misalignment, the reconstruction quality may decrease noticeably. The robustness of our approach is mainly bottlenecked by the estimation of curvatures, which is a differential operation on the underlying surface. Differential operators often become unstable in the presence of severe data imperfections.

The curvature-based adaptivity is not suitable for reconstructing extremely thin shapes, e.g., a thin plate. Because the points at each side of the plate, except those on the boundary, are nearly on a plane with very small curvatures, the subdivisions of certain octree nodes may terminate too early such that both

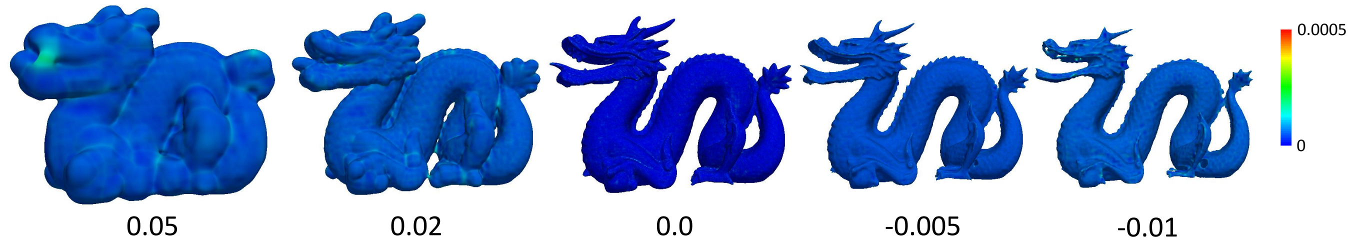


Fig. 11. Generation of nonzero/zero level sets for the Dragon model. These level sets correspond to the signed distance values in the below, which are normalized by the diagonal of the bounding box of the model. The colors are mapped by the deviations of the level sets relative to the real offset surfaces.

sides of the points remain in the same node. For these nodes, the signed distances computed at the corners are always positive, and therefore no face will be extracted. Figure 12 shows a failure example of reconstructing the Blade model, where holes appears in the very thin and nearly planar parts of the model. Reconstruction of extremely thin shapes remains a challenging problem. In terms of our approach, the most effective solution is to recursively subdivide all nonempty octree nodes until a sufficient depth is reached.

5. Conclusion and Future Work

We propose a multi-scale approach for surface reconstruction from oriented point sets. The basic idea is to build an adaptive signed distance filed on an octree grid and to then fit it using an implicit function. We introduce a curvature-based strategy for the adaptive octree construction. Therefore, scale-varying geometric details can be reconstructed. A set of multi-scale B-spline basis functions are adopted to define the implicit function; thus, a C^1-continuous field function is obtained by solving a well-conditioned sparse linear system. The signed distance field is well approximated by the field function; thus, its nonzero level sets can also well approximate the offset of the underlying surface. The produced mesh surfaces are crack-free and adaptive to the shape complexities of the underlying surfaces.

There are some possible directions to improve our approach in the future. First, the multi-thread implementation can be further accelerated using GPU computation [46, 47, 48]. Second, an out-of-core version of our approach can be adapted to address very large data sets. Additionally, we believe that our approach can be useful in many other applications, such as mesh simplification, level of details mesh compression and transmission, and constructive solid geometric modelling, where the ability of multi-scale surface representation or signed distance evaluation is crucial.

Acknowledgment

We would like to thank XXX and his colleagues for their help in preparing the comparison experiments. Thanks also go to the Stanford 3D repository, the Aim@Shape project, the Large Geometric Models Archive at Georgia Tech and the surface reconstruction benchmark [15] for providing the models used in this paper. The work is support by XXX. (The terms "XXX" are temporarily omitted for review.)

References

[1] Kobbelt, L, Botsch, M. A survey of point-based techniques in computer graphics. Computers & Graphics 2004;28(6):801–814.
[2] Alexa, M, Behr, J, Cohen-Or, D, Fleishman, S, Levin, D, Silva, CT. Point set surfaces. In: IEEE Visualization 2001, October 24-26, 2001, San Diego, CA, USA, Proceedings. 2001, p. 21–28.
[3] Amenta, N, Kil, YJ. Defining point-set surfaces. ACM Trans Graph 2004;23(3):264–270.
[4] Dey, TK, Sun, J. An adaptive MLS surface for reconstruction with guarantees. In: Third Eurographics Symposium on Geometry Processing, Vienna, Austria, July 4-6, 2005. 2005, p. 43–52.
[5] Carr, JC, Beatson, RK, Cherrie, JB, Mitchell, TJ, Fright, WR, McCallum, BC, et al. Reconstruction and representation of 3d objects with radial basis functions. In: Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 2001, Los Angeles, California, USA, August 12-17, 2001. 2001, p. 67–76.
[6] Kazhdan, MM. Reconstruction of solid models from oriented point sets. In: Third Eurographics Symposium on Geometry Processing, Vienna, Austria, July 4-6, 2005. 2005, p. 73–82.
[7] Fuhrmann, S, Goesele, M. Floating scale surface reconstruction. ACM Trans Graph 2014;33(4):46:1–46:11.
[8] Hoppe, H, DeRose, T, Duchamp, T, McDonald, JA, Stuetzle, W. Surface reconstruction from unorganized points. In: Proceedings of the 19th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1992, Chicago, IL, USA, July 27-31, 1992. 1992, p. 71–78.
[9] Curless, B, Levoy, M. A volumetric method for building complex models from range images. In: Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1996, New Orleans, LA, USA, August 4-9, 1996. 1996, p. 303–312.
[10] Mullen, P, de Goes, F, Desbrun, M, Cohen-Steiner, D, Alliez, P. Signing the unsigned: Robust surface reconstruction from raw pointsets. Comput Graph Forum 2010;29(5):1733–1741.
[11] Calakli, F, Taubin, G. SSD: smooth signed distance surface reconstruction. Comput Graph Forum 2011;30(7):1993–2002.
[12] Pan, M, Tong, W, Chen, F. Phase-field guided surface reconstruction based on implicit hierarchical b-splines. Computer Aided Geometric Design 2017;52-53:154–169.
[13] Berger, M, Tagliasacchi, A, Seversky, LM, Alliez, P, Levine, JA, Sharf, A, et al. State of the art in surface reconstruction from point clouds. In: Eurographics 2014 - State of the Art Reports, Strasbourg, France, April 7-11, 2014. 2014, p. 161–185.
[14] Berger, M, Tagliasacchi, A, Seversky, LM, Alliez, P, Guennebaud, G, Levine, JA, et al. A survey of surface reconstruction from point clouds. Comput Graph Forum 2017;36(1):301–329.
[15] Berger, M, Levine, JA, Nonato, LG, Taubin, G, Silva, CT. A benchmark for surface reconstruction. ACM Trans Graph 2013;32(2):20:1–20:17.
[16] Berger, M, Levine, J, Nonato, LG, Taubin, G, Silva, CT. An end-to-end framework for evaluating surface reconstruction. In: Sci technical report, University of Utah, Jan. 2011. 2011,.
[17] Kolluri, RK, Shewchuk, JR, O'Brien, JF. Spectral surface reconstruction from noisy point clouds. In: Second Eurographics Symposium on Geometry Processing, Nice, France, July 8-10, 2004. 2004, p. 11–21.
[18] Amenta, N, Choi, S, Kolluri, RK. The power crust. In: Sixth ACM Symposium on Solid Modeling and Applications, Sheraton Inn, Ann Arbor, Michigan, USA, June 4-8, 2001. 2001, p. 249–266.

[19] Edelsbrunner, H, Mücke, EP. Three-dimensional alpha shapes. ACM Trans Graph 1994;13(1):43–72.
[20] Dey, TK, Giesen, J. Detecting undersampling in surface reconstruction. In: Proceedings of the Seventeenth Annual Symposium on Computational Geometry, Medford, MA, USA, June 3-5, 2001. 2001, p. 257–263.
[21] Amenta, N, Choi, S, Dey, TK, Leekha, N. A simple algorithm for homeomorphic surface reconstruction. Int J Comput Geometry Appl 2002;12(1-2):125–141.
[22] Dey, TK, Wang, L. Voronoi-based feature curves extraction for sampled singular surfaces. Computers & Graphics 2013;37(6):659–668.
[23] Xiong, S, Zhang, J, Zheng, J, Cai, J, Liu, L. Robust surface reconstruction via dictionary learning. ACM Trans Graph 2014;33(6):201:1–201:12.
[24] Dey, TK. Curve and Surface Reconstruction: Algorithms with Mathematical Analysis (Cambridge Monographs on Applied and Computational Mathematics). New York, NY, USA: Cambridge University Press; 2006. ISBN 0521863708.
[25] Cazals, F, Giesen, J. Delaunay triangulation based surface reconstruction: Ideas and algorithms. In: EFFECTIVE COMPUTATIONAL GEOMETRY FOR CURVES AND SURFACES. Springer; 2006, p. 231–273.
[26] Guennebaud, G, Gross, MH. Algebraic point set surfaces. ACM Trans Graph 2007;26(3):23.
[27] Pauly, M. Point primitives for interactive modeling and processing of 3d-geometry. Ph.D. thesis; ETH Zurich; 2003.
[28] Yang, P, Qian, X. Direct computing of surface curvatures for point-set surfaces. In: Symposium on Point Based Graphics, Prague, Czech Republic, 2007. Proceedings. 2007, p. 29–36.
[29] Ohtake, Y, Belyaev, AG, Alexa, M, Turk, G, Seidel, H. Multi-level partition of unity implicits. ACM Trans Graph 2003;22(3):463–470.
[30] Süßmuth, J, Meyer, Q, Greiner, G. Surface reconstruction based on hierarchical floating radial basis functions. Comput Graph Forum 2010;29(6):1854–1864.
[31] Manson, J, Petrova, G, Schaefer, S. Streaming surface reconstruction using wavelets. Comput Graph Forum 2008;27(5):1411–1420.
[32] Jüttler, B, Felis, A. Least-squares fitting of algebraic spline surfaces. Adv Comput Math 2002;17(1-2):135–152.
[33] Pan, M, Tong, W, Chen, F. Compact implicit surface reconstruction via low-rank tensor approximation. Computer-Aided Design 2016;78:158–167.
[34] Kazhdan, MM, Bolitho, M, Hoppe, H. Poisson surface reconstruction. In: Proceedings of the Fourth Eurographics Symposium on Geometry Processing, Cagliari, Sardinia, Italy, June 26-28, 2006. 2006, p. 61–70.
[35] Kazhdan, MM, Hoppe, H. Screened poisson surface reconstruction. ACM Trans Graph 2013;32(3):29:1–29:13.
[36] Levoy, M, Pulli, K, Curless, B, Rusinkiewicz, S, Koller, D, Pereira, L, et al. The digital michelangelo project: 3d scanning of large statues. In: Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 2000, New Orleans, LA, USA, July 23-28, 2000. 2000, p. 131–144.
[37] Sharma, O, Agarwal, N. Signed distance based 3d surface reconstruction from unorganized planar cross-sections. Computers & Graphics 2017;62:67–76.
[38] Fuhrmann, S, Goesele, M. Fusion of depth maps with multiple scales. ACM Trans Graph 2011;30(6):1–8.
[39] Mücke, P, Klowsky, R, Goesele, M. Surface reconstruction from multi-resolution sample points. In: Proceedings of the Vision, Modeling, and Visualization Workshop 2011, Berlin, Germany, 4-6 October, 2011. 2011, p. 105–112.
[40] Goldman, R. Curvature formulas for implicit curves and surfaces. Computer Aided Geometric Design 2005;22(7):632–658.
[41] Kazhdan, MM, Klein, AW, Dalal, K, Hoppe, H. Unconstrained isosurface extraction on arbitrary octrees. In: Proceedings of the Fifth Eurographics Symposium on Geometry Processing, Barcelona, Spain, July 4-6, 2007. 2007, p. 125–133.
[42] FLANN - Fast Library for Approximate Nearest Neighbors. `http://www.cs.ubc.ca/research/flann/`; 2013. Accessed: 2017-04-01.
[43] Eigen. `http://eigen.tuxfamily.org/index.php?title=Main_Page`; 2013. Accessed: 2017-04-01.
[44] Cignoni, P, Rocchini, C, Scopigno, R. Metro: Measuring error on simplified surfaces. Comput Graph Forum 1998;17(2):167–174.
[45] Gibson, SFF, Perry, RN, Rockwood, AP, Jones, TR. Adaptively sampled distance fields: a general representation of shape for computer graphics. In: Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 2000, New Orleans, LA, USA, July 23-28, 2000. 2000, p. 249–254.
[46] Zhou, K, Hou, Q, Wang, R, Guo, B. Real-time kd-tree construction on graphics hardware. ACM Trans Graph 2008;27(5):126:1–126:11. URL: `http://doi.acm.org/10.1145/1457515.1409079`. doi:`10.1145/1457515.1409079`.
[47] Bastos, T, Filho, WC. Gpu-accelerated adaptively sampled distance fields. In: 2008 International Conference on Shape Modeling and Applications (SMI 2008), June 4-6, 2008, Stony Brook, NY, USA. 2008, p. 171–178.
[48] Liu, F, Kim, YJ. Exact and adaptive signed distance fieldscomputation for rigid and deformablemodels on gpus. IEEE transactions on visualization and computer graphics 2014;20(5):714–725.

Appendix A.

By computing the derivatives of $E_D(f)$, $E_R(f)$, $E_P(f)$, $E_N(f)$ relative to each basis function coefficient v_c and equating them to zero, we respectively obtain the constraint matrices L_D, L_R, L_P, L_N and the constraint vectors $\mathbf{b}_D$, $\mathbf{b}_N$ as follows

$$
\begin{aligned}
&L_D = \frac{1}{m} {K_D}^T K_D \quad \text{with} \quad (K_D)_{jc} = B_c(\mathbf{q}_j) \\
&L_P = \frac{1}{n} {K_P}^T K_P \quad \text{with} \quad (K_P)_{ic} = B_c(\mathbf{p}_i) \\
&L_N = \frac{1}{n} \left({K_{Nx}}^T K_{Nx} + {K_{Ny}}^T K_{Ny} + {K_{Nz}}^T K_{Nz} \right) \quad \text{with} \\
&(K_{Nx})_{ic} = (B_c)_x(\mathbf{p}_i) \quad (K_{Ny})_{ic} = (B_c)_y(\mathbf{p}_i) \quad (K_{Nz})_{ic} = (B_c)_z(\mathbf{p}_i) \\
&\mathbf{b}_D = \frac{1}{m} {K_D}^T \mathbf{d} \\
&\mathbf{b}_N = \frac{1}{n} \left({K_{Nx}}^T \mathbf{n}_x + {K_{Ny}}^T \mathbf{n}_y + {K_{Nz}}^T \mathbf{n}_z \right) \\
&(L_R)_{cc'} = \frac{1}{|V|} \int_V \Big((B_c)_{xx}(B_{c'})_{xx} + (B_c)_{yy}(B_{c'})_{yy} + (B_c)_{zz}(B_{c'})_{zz} + \\
&\qquad 2(B_c)_{xy}(B_{c'})_{xy} + 2(B_c)_{xz}(B_{c'})_{xy} + 2(B_c)_{yz}(B_{c'})_{xy} \Big)
\end{aligned} \tag{A.1}
$$

where $\mathbf{d}$ is the vector consisting of the signed distance values $\{d_j\}$, and $\mathbf{n}_x$, $\mathbf{n}_y$, $\mathbf{n}_z$ are the vectors consisting of the x, y, z components respectively of the normals $\{\mathbf{n}_i\}$; $(B_c)_x$, $(B_c)_y$, $(B_c)_z$ are the partial derivatives of B_c relative to x, y, z, respectively; $(B_c)_{xx}$ and other similar terms are the second order partial derivatives of B_c. By analysing the above formulas, we find that L_D, L_R, L_P, L_N are all symmetric matrices.