A subdivision-based framework for shape reconstruction

Liu Shaolong[1] · Liu Na[2] · Lv Chenlei[3] (ORCID) · Zhang Dan[3]

© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2024

Abstract

Shape reconstruction from 3D point clouds is one of the most important topic in the field of computer graphics. In this paper, we propose a subdivision-based framework for this topic. The framework includes two parts: distance field optimization and mesh generation. The first part optimizes a point cloud into an approximately isotropic one based on a subdivision structure. The second part is to generate a triangular mesh from the optimized point cloud. The mesh is regarded as the result of shape reconstruction. The advantages of our method includes accurate geometric consistency, improved mesh quality, controllable point number, and fast speed. Experiments indicate that our method has good performance for shape reconstruction (compare to the state-of-the-art, our method achieves five and six times improvement in Hausdorff distance-based measurement and density estimation). The executable file is available: (https://github.com/vvvwo/Parallel-Structure-ShapeReconstruction.)

1 Introduction

Following the development of 3D scanning technology, 3D point clouds have been widely used in different fields such as 3D object modeling and virtual reality, etc. Compared to 2D image or video-based analysis framework [29, 30], Using 3D point clouds with complete geometric features can support more accurate analysis tasks [6]. However, point clouds cannot be used directly in such tasks since some geometric features should be extracted from continuous surface, but not discrete points. Some applications require accurate surface to represent real objects such as face recognition [19], skull reconstruction [31, 32], terrain modeling [18, 28]. Therefore, point cloud-based shape reconstruction is proposed to solve the problem.

Liu Shaolong and Liu Na contributed equally to this work.

✉ Zhang Dan
  danz@mail.bnu.edu.cn

1   Beijing Normal University, School Of Art And Communication, Beijing, 100579, China

2   Beijing Normal University, School of AI, Beijing, 100579, China

3   Qinghai Normal University, The Computer College, Xining, 810008, China

The target of the shape reconstruction is to rebuild a representation of simplicial complex for the point cloud. It can be regarded as a continuous 3D surface, which is represented by a 3D triangular mesh in most cases. To achieve the mesh, connections of different points in a point cloud should be determined. Based on the theory of moving least-squares (MLS) [1], connections can be recovered by local regions defined by points with their neighbors. The performance of the reconstruction depends on two issues: geometric consistency and accuracy of local regions. The geometric consistency means that the reconstructed mesh should be consistent to the MLS surface of original point cloud. It guarantees the reconstructed mesh can be used to represent the original point cloud. The accuracy of local regions determines the quality of the mesh structure. It is important for subsequent mesh-based feature extraction and analysis.

For the first issue, some methods are proposed to implement the reconstruction from a point cloud directly such as Delaunay triangulation [9], Ball pivoting [4], Scale space [13], etc. These methods do not change point positions in reconstruction which keeps the consistency between the reconstructed mesh and the original point cloud. However, the accuracy of local regions can not be assured and promoted. In the view of mathematical model, the quality of local regions can be represented by a distance field that is a collection constructed by distances between all points and their neighbors. It reflects the uniform degree of a point cloud. An optimized distance field means that distances between all points and their neighbors are similar. The aforementioned methods do not optimize the distance field and produces many low quality and error triangles which affect geometric feature extraction in related applications.

A well-known solution is to resample a point cloud into an isotropic one based on Centroidal Voronoi Tessellation (CVT) [8]. The positions of points are iteratively updated to generate a global isotropic point cloud. When the update is completed, the global optimization of distance field is achieved. Based on the optimized distance field, the accuracy of local regions can be assured and the quality of the reconstructed mesh is improved. Considering the time cost of Voronoi cell estimation, some methods [22, 23] attempt to optimize the distance field directly. However, the solution changes the original point cloud in local tangent space excessively. Most of points are moved which break the geometric consistency between the point cloud and the reconstructed mesh in a certain extent.

In this paper, we propose a subdivision-based framework for point cloud-based shape reconstruction. As mentioned before, our framework is constructed by two parts: distance field optimization and mesh generation. The distance field optimization is implemented by a point cloud simplification scheme, which is simplify a point cloud into an approximate isotropic one. It is formulated as a simple and efficient distance field optimization. The simplification is processed in a subdivision structure which improves the efficient by parallel computation. Most of points in the simplified point cloud are kept in their original positions. Combining an up-sampling processing, the point number of simplification can be controlled even without loss of number to the original one. For special geometric feature keeping such as sharp edge, a flexible simplification is used as an optional function. In the part of mesh generation, we reconstruct high quality meshes based on the proposed subdivision structure. It provides the topological constraints by adjacent boxes. According to the constraints, a mesh cropping is designed to remove error connections. Compared to the traditional methods, the accuracy of local regions is improved. In Fig. 1, we show some instances by our method. In summary, our contributions are as follows.

- A subdivision-based framework is proposed for point cloud-based mesh reconstruction. The framework balances the geometric consistency and accuracy of local regions while
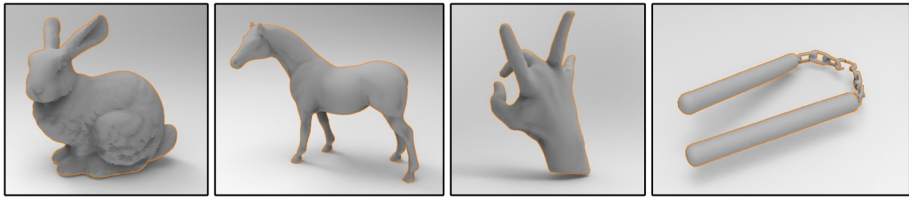
Fig. 1 Four instances of reconstruct meshes based on our method. The vertex numbers of all meshes are equal (10,000)

keeping more geometric features in reconstructed mesh. It improves the practicality of the reconstruction.

- An efficient distance optimization method is proposed, which is constructed by preprocessing, simplification, and up-sampling. The method can be used to efficiently optimize the distance field of a point cloud with a certain point number. It is robust to the input point cloud with non-uniform density.
- Based on the topological constraints provided by the subdivision structure, a mesh cropping process is provided to avoid error connections. It improves the accuracy of local regions.

The rest of the paper is organized as follows. In Section 2, we introduce relevant classical and state of the art works for point cloud-based shape reconstruction. In Section 3, we discuss the fundamental of our framework. In Section 4, we show the details of distance field optimization based on the subdivision structure. In Section 5, we introduce the mesh generation with cropping. The experiments in Section 6 show the effectiveness and efficiency of our method.

2 Related works

There have tremendous works for point cloud-based shape reconstruction during the past two decades. In this paper, we focus on single 3D object shape reconstruction without data driven support. Considering the relationship between the exits works and our method, we select the most relevant parts of them to discuss, which are classified into three categories: Approximation-based, Delaunay-based, and Point resampling-based.

Approximation-based methods attempt to rebuild a 2-manifold to fit point cloud. The methods achieve the reconstruction by establishing an objective function, such as Poisson Function [15, 24], Scale Space [11], Subdivision-based fitting [2, 12, 20], and spline-based modeling [26]. Poisson function was a classical method for point cloud based meshing task. The core idea was solving a Poisson equation to achieve an indicator function, which was a piece-wise constant function and signed the different sides of the surface. Scale space meshing provided a smoothing operator for the raw point clouds. According to estimate the mean curvature and solve a mean curvature motion in point clouds, a smooth surface was constructed [11]. In summary, such methods reconstruct smooth surface from a point cloud and are robust to noise. However, the local shape features are broken in a certain degree and the stability are not good (wrong estimation for normal vectors and incorrect approximate region).

Delaunay-based framework is regarded as the mainstream technology in 3D triangular meshing. It provides a simple and efficient point connect scheme for a point cloud without

local surface approximation. There has a lot of works based on Delaunay triangulation and its improved version. Ameta et al. [3] utilized the dual characteristics between Voronoi Diagram and Delaunay triangulation to rebuild the surface. Cohen et al. [9] proposed a greedy Delaunay triangulation to fix the local errors in the reconstruct mesh. Such methods are restricted by the points' positions in general. The quality of the reconstruct mesh is poor when the points' distribution of input point cloud is nonuniform.

Considering the drawback of the Delaunay-based methods, some works attempt to resample the point cloud into an isotropic one. The distance field is optimized by the resampling process. The quality of the reconstruct mesh is improved naturally based on resampling point clouds. The representative method is Centroidal Voronoi Tessellation (CVT) [8] based resampling. Based on the Lloyd's relaxation, the Voronoi Diagram was optimized in local tangent space. The advantages of such works include high quality of the triangulation result and robust to different local points' density. Based on the isotropic remeshing, Lv et al. proposed a point cloud-based reconstruction method [22] to generate isotropic mesh. After that, he extended the framework to output curvature adaptive result [23]. However, some local geometric features are lost during the optimization and the geometric consistency is broken by the point movements. Our reconstruction method is classified into this part and attempt to keep the geometric consistency and more features in the reconstructed mesh.

3 Fundamental

The point cloud-based shape reconstruction task is to construct a representation of simplicial complex to fit the discrete points. According to the two issues for reconstruction, we propose the related mathematical models. For geometric consistency, the mathematical representation of the consistent can be represented by

$$E_{con} = H(M, \ P), \tag{1}$$

where E_{con} is the quantified energy of geometric consistency between reconstructed mesh M and original point cloud P. It is computed by the measurement H which can be computed by the Hausdorff distance [5]. In the reconstruction, the E_{con} should be reduced as much as possible.

It has been discussed that the point-based distance field is used to represent local regions. By optimizing the field, local regions of different points can be accurately detected. An ideal distance field means different points share same distance to their neighbor points. It is called isotropic property. In (2), the quantification of isotropic property is formulated as

$$E_{iso} = \sum_{p_i} \sum_{p_j} \left\| b(p_i, p_j) - \overline{b} \right\|, p_i, p_j \in P, \tag{2}$$

where E_{iso} is the distance field energy for isotropic property measurement. P is the point cloud, p_i and p_j are points in P. The point p_j is the neighbor point for p_i. The parameter b is the distance of two neighbor points and $\overline{b}$ is the average border distance of all border distance b in P. If we want to build the correct connections between different points in a point cloud, the border distance b and the relationship between points and their neighbors should be optimized. Then, the shape reconstruction task can be transferred into the distance field optimization with geometric consistency.

Based on the aforementioned knowledge, we design a simple and efficient framework to optimize the distance field for shape reconstruction while keeping geometric consistency.

The pipeline of our method is shown in Fig. 2. The framework is based on a subdivision structure which is similar to voxelization. Points are divided into different voxel boxes. The distances between the points are optimized in the voxel boxes independently. The optimized point cloud is the combination from the point sets in different voxel boxes. We also provide the mesh generation method to reconstruct the triangular mesh from the optimized point cloud. With a mesh cropping, the accuracy of the reconstructed mesh can be further improved. In following parts, we discuss details of the implementation of our framework.

4 Distance field optimization

As the first part of our framework, the distance field optimization is to optimize E_{iso} of (2). Considering the drawback of resampling [8], we do not want to change many point locations in optimization. The core thought of our method is concise and efficient: we remove some points to adjust the distribution of the point cloud, then the E_{iso} can be optimized and most of reserved points are kept in their original locations. Therefore, our distance field optimization can be regarded as a kind of simplification. Following the core thought, we design the implementation of the distance field optimization.

Before the formal introduction of the implementation, we provide a pre-processing step for input point clouds. In general, a point cloud scanned from laser scanner or other equipment may take some noise and redundant points. Such influence factors affect the efficient in simplification. Therefore, the pre-processing is necessary to reduce the influence of the factors. In our method, the pre-processing includes two functions: denoising and initial simplification. For denoising, we utilize PointCleanNet [27] in pre-processing. For initial simplification, we apply Poisson-disk resampling [10] which can be used to reduce the density of point cloud and uniform the point distribution. The initial simplification also provide a searching radius to control the edge connection in shape reconstruction. In the next section, the radius is used in mesh cropping to improve the quality of reconstructed result.

After pre-processing, we introduce the implementation of our distance field optimization. To adjust the distribution of the point cloud, we utilize the simplification in the subdivision structure. A same implementation is explained in [21]. It divides points into different voxel boxes according to certain rate. In (3), we provide the mathematical model as

$$P_{sim} = \sum_{v \in V} \{p | p \in P_v, P_v \in P\}, \tag{3}$$

where P is the input point cloud after pre-processing, P_v is a point set of P in a voxel box v, V represents the collection of all voxel boxes. The simplification result P_{sim} is constructed by simplified results in different voxel boxes while reducing the energy in (2). Once points
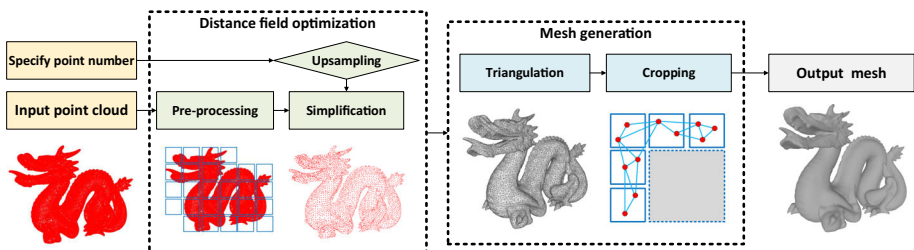

Fig. 2 The pipeline of our method

of P are divided into different voxel boxes, the simplification result can be achieved from parallel simplification in boxes independently. The scale of voxel box in the structure has an obviously influence for simplification. If the scale is too small, the candidate points in single voxel box cannot support simplification. It produces false discontinuities in local region. In contrast, the large scale of voxel box reduces the performance of parallel simplification and the topological constraints are lost. In practise, we provide a default value based on experience, which is shown as

$$L_{scale} = \frac{L_{max}}{\left[\sqrt[3]{|P|/8}\right]},$$ (4)

where L_{scale} is the scale, L_{max} is the longest length of three axes of P, $|P|$ is the point number.

Based on the subdivision structure, we use the farthest point sampling (FPS) [25] to simplify point cloud directly. The FPS is processed in different voxel boxes independently. The start point in each voxel box is selected from the point which has the closest distance to the center of the box. The simplification point number is computed by the simplify ratio computed by the original point number and simplification point number specified by user. The points in each voxel box are simplified by equal ratio. Combining the reserved points from different voxel boxes, the final simplification result is obtained. If the point number of simplification is not equal to user specified one, we just add or reduce some points in different boxes following the FPS order. Finally, the accurate simplification result is achieved and it is an approximate isotropic one. In Fig. 3, we show an instance of simplification.

The FPS achieves the simplification points in different voxel boxes independently. In each voxel box, the distance field of the point subset is optimized. However, the distances between points in borders of adjacent voxel boxes are not adjusted. To solve the problem, we divide the parallel computation for simplification into different rounds. The adjacent voxel boxes are not simplified in same round. In addition, the simplification points of neighbor boxes should be included by FPS for the processing voxel box. In Fig. 4, we show an instance to explain the scheme. The parallel computation is divided into eight rounds which guarantees the global uniform property of simplification. The order of eight rounds for boxes selection is shown in Fig. 4.

As mentioned before, the implementation of the distance field optimization is based on the simplification. A prerequisite is that the point number of reconstructed mesh should be significantly less than the original point cloud, which ensures the simplification has enough candidate points to simplify. Obviously, it reduces the practicality of the method. Once the point number should be kept after reconstruction, the requirement cannot be satisfied by the current implementation. To solve the problem, we add a up-resampling process to increase density of point cloud before simplification. We search a local region represented by Voronoi
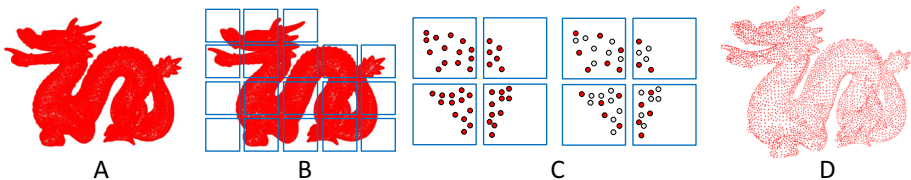
Fig. 3 An instance of simplification for distance field optimization. A: input point cloud; B: subdivision structure; C parallel simplification in adjacent voxel boxes (red points represent original points in point cloud, gray points are redundant points removed by FPS); D: simplification result
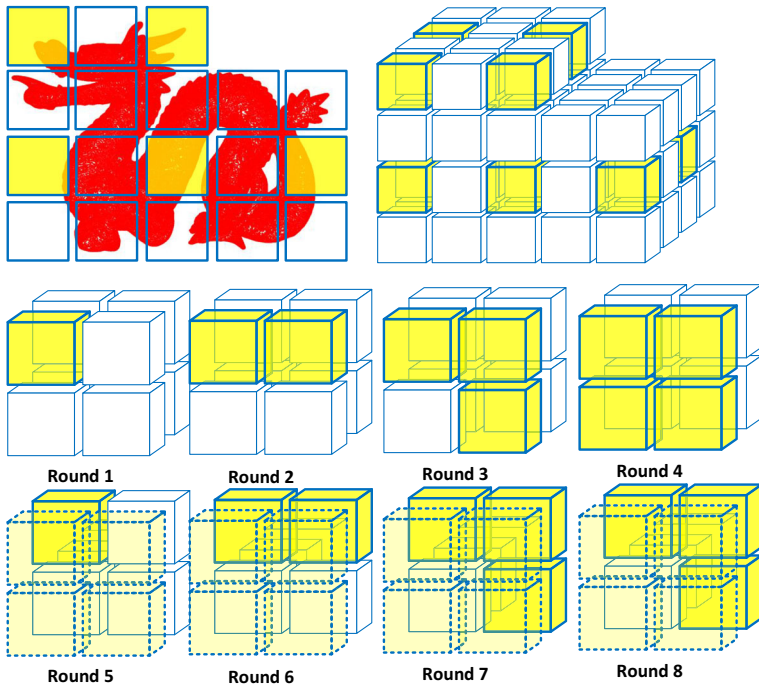
Fig. 4 Parallel simplification scheme. The yellow boxes are arraigned into a same round for parallel simplification. The order of boxes selection in eight rounds. The adjacent boxes are not simplified in a same round

cell for each point and insert new points proportionally. By default, we insert three points into each border of Voronoi cell and one point into each related edge of a triangle. The interpolation add new points as a arithmetic sequence into each intersection area between the triangle area and the Voronoi cell. The new points are mapped into the MLS surface of the point cloud for geometric consistency keeping. Based on the interpolation, the density of point cloud can be increased according to the Voronoi cells and the process can be implemented parallel. Even there have some repeat points inserted into the adjacent area of cells, the simplification can ignore them to obtain final result. An instance is shown in Fig. 5.

To keep some import geometric features, we use the flexible simplification [21] in the implementation. The basic idea is to simplify points with different rates to enhance the certain feature. For instance, if we want to keep and reconstruct sharp edges from a point cloud, we detect edge points at first. Then, we simplify edge points with a higher rate in simplification and remove more normal points. The sharp features are enhanced in the simplification which can be inherited into the mesh generation. However, such enhancement should be controlled in a certain degree; otherwise, the E_{iso} is increased in the regions with sharp edges. A suitable ratio for simplification (8:2) is recommended for balance. The formulation of the flexible simplification is represented as

$$\begin{cases} |P_n|R_n + |P_e|R_e = |P_s| \\ R_n/R_e = 2/8 \end{cases} \tag{5}$$

where $|P_n|$ and $|P_e|$ are the point numbers of normal and edge point set, R_n and R_e are simplification rate for $|P_n|$ and $|P_e|$. $|P_s|$ is the simplification point number specified by user.
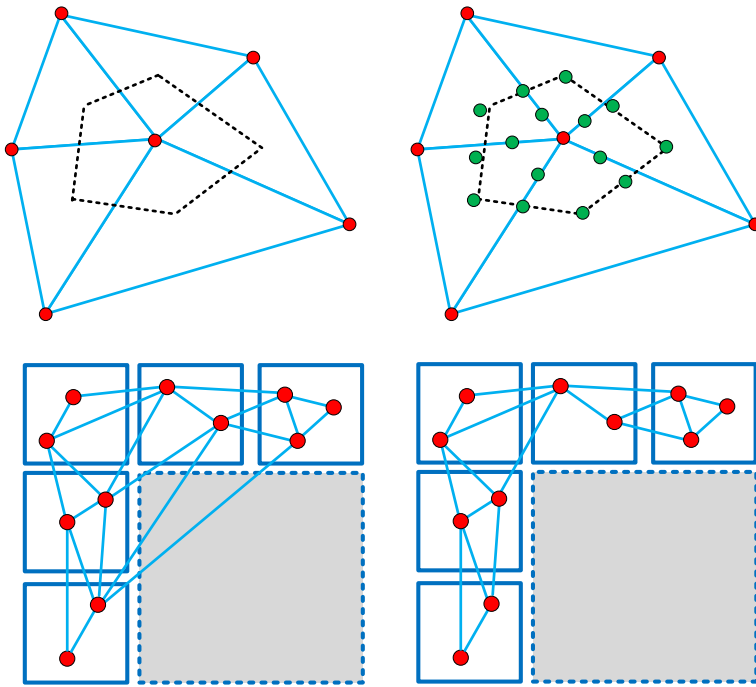
Fig. 5 Two instances of up-sampling and mesh cropping. First row: the new points (green) are inserted into the point cloud according to the Voronoi cell; second: the gray area means there have no reserved points in the voxel boxes. The triangle passing through the gray area should be deleted

According to the (5), we can obtain the simplification result with sharp feature keeping. An instance is shown in Fig. 6.

5 Mesh generation

After the distance field optimization, an approximately isotropic point cloud is obtained from the original one. Based on the point cloud, we provide mesh generation as a second part in our framework. Benefited from the distance field optimization, the local regions can be accurately detected. We utilize a greedy Delaunay triangulation [9] to detect accurate local regions and generate mesh as the shape reconstruction result, which has been realized in CGAL library. The mesh generation inherits the advantage of Delaunay triangulation for geometric consistency. It achieves the balance between E_{con} and E_{iso}.

Even the generated mesh from the approximately isotropic point cloud has a higher quality, the incorrect connections between points are inevitable. The reason is that the greedy Delaunay triangulation does not consider the topological constraints of the original point cloud. The topological constraints in original point cloud are kept in the subdivision structure. The adjacent voxel boxes represent the constraints in the structure. Suppose that a triangle in generated mesh is crossing the region without reserved points, which means that the triangle has incorrect connections. In Fig. 5, an instance shows the situation. According to the constraints in the subdivision structure, we provide a mesh cropping processing to remove error connections.
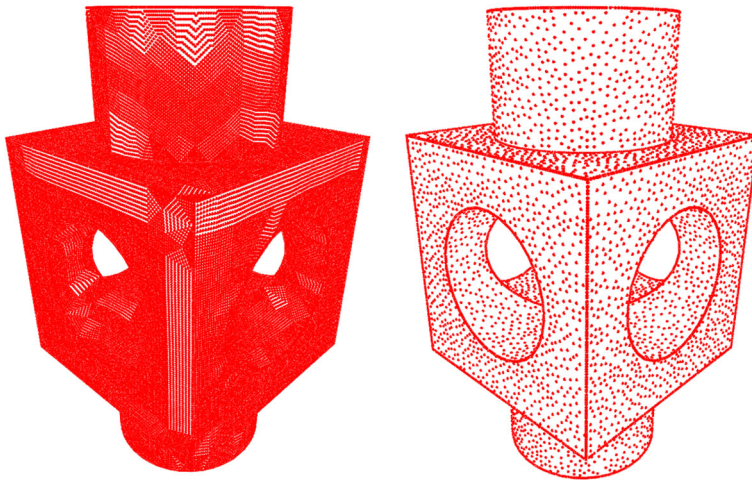
Fig. 6 An instance of flexible simplification for sharp edge keeping. The edges are enhanced in the simplification

The cropping process is used to delete triangles with incorrect connections. A correct triangle of the mesh should follow the adjacent limitation between voxel boxes in the subdivision structure. If the voxel boxes or the local regions have no reserved points of simplification, we call them "gray region" (Fig. 5). A triangle is passing the gray region means that the edge is crossing the discontinuous area. The triangles should be deleted. The mesh cropping is based on the rule to delete such triangles. As mentioned before, the initial simplification in pre-processing computes a searching radius. It can be used to control the edge connection. Once an edge of a triangle is longer than the radius, the cropping is also triggered. After cropping, error edges are deleted which improve the quality of reconstructed mesh.

6 Experiments

We show the performance of our method in this section. The experimental point clouds are selected from Stanford [16] and SHREC [7] models. The experimental platform is constructed by Visual Studio 2019 in windows 10 system (X64). The hardware configuration is constructed by a laptop machine, Intel i7 9750H 2.6GHz, 16G RAM, and GeForce GTX 1660Ti. The datasets are constructed from Stanford and SHREC models. Our program can be downloaded from https://github.com/vvvwo/Parallel-Structure-ShapeReconstruction. Using the ".exe" file, users can construct triangular meshes from point clouds directly. The program supports different kinds of data format, including .off, .obj, and .ply. The input parameters include file path and specified point number. The section contains three parts: firstly, we evaluate the accuracy of geometric consistency of different reconstruction methods; secondly, we measure the quality of reconstructed meshes by the methods; finally, we provide a comprehensive analysis for the methods.
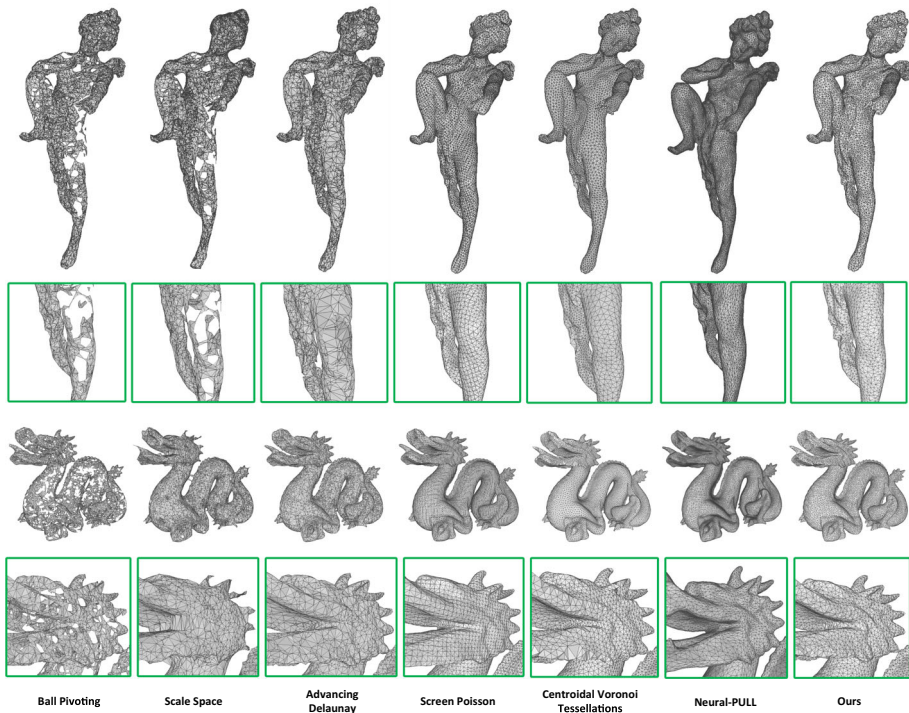
6.1 Evaluation of geometric consistency

The accuracy of geometric consistency can be represented by the Hausdorff distance mentioned in (1). In order to evaluate the influence of triangle with error edges, we sample
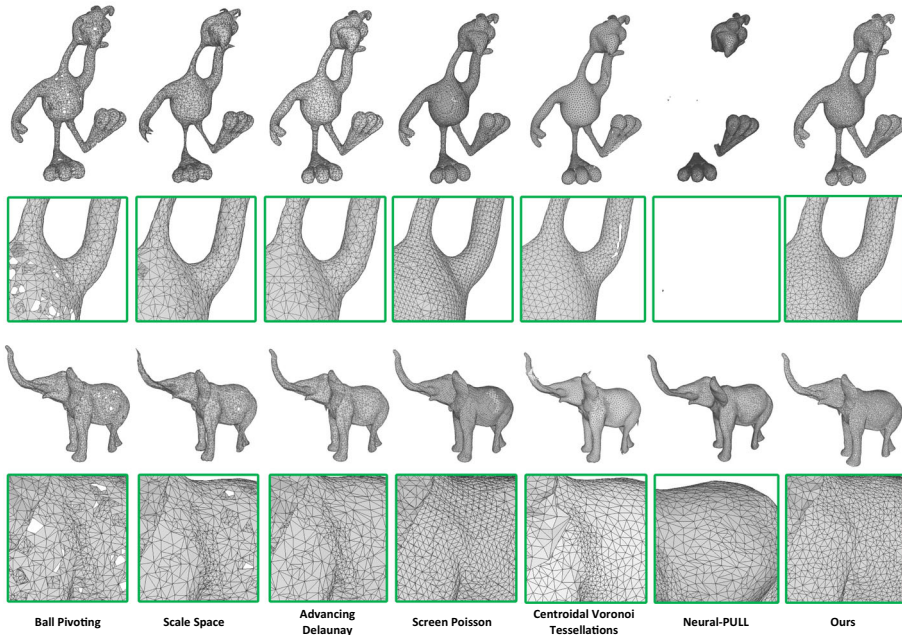
some points from triangles in the reconstructed mesh and add them into the computation of Hausdorff distance. Even the points are not moved in some methods, the Hausdorff distance is increased by the sampled points from error triangles. We compare the distances of different reconstruction methods, including Ball pivoting [4], Scale Space reconstruction [11], Screen Poisson [15], Advancing Delaunay reconstruction [9, 17], CVT-based reconstruction [8], Neural-Pull [24] and our method. Some methods cannot control the point in the reconstruction, including Ball pivoting, Scale Space, and Advancing Delaunay. We use resampling method [14] implemented by CGAL library to sample the original point cloud with same point number for fair. In Fig. 7, the results are shown from Stanford models. We also test the reconstruction without change of point number. In Fig. 8, reconstructed meshes by different methods are shown from SHREC models. It can be used to explain the function of up-sampling in our framework. In Table 1, we show the Hausdorff distances of different methods. Our method achieves better geometric consistency in the reconstruction.

6.2 Evaluation of mesh quality

To measure the quality of reconstructed meshes, the average value of minimum inner angles which reflects the isotropic property is used. The minimum inner angle means the smallest angle in a triangle. The average value of the angles represents the quality of the triangles in the reconstructed mesh. We generate color maps for the visualization of angles. In Fig. 9, we show the color maps. We also provide the density parameter D_r computed by the average distances between all points and their k-neighbors to represent the quality of distance field

| Ball Pivoting | Scale Space | Advancing Delaunay | Screen Poisson | Centroidal Voronoi Tessellations | Neural-PULL | Ours |

Fig. 7 Comparisons of reconstructed meshes from Stanford models by different methods

Fig. 8 Comparisons of reconstructed meshes from SHREC models by different methods. With the up-sampling, our method does not reduce point number in final result

of reconstructed mesh. The computation is shown as:

$$D_r = Max\{A(M)\} - Min\{A(M)\}, \tag{6}$$

where A represents the average distances set of the mesh M, and the average distance is computed between a point and its adjacent points according to the mesh. We select the maximum and minimum values from $A(M)$ and compute the difference. Then, the D_r represents the degree of uniform density in the reconstructed meshes. In Table 2, we compare the global average value of minimum inner angles and density parameters in reconstructed meshed by different methods. Our method achieves similar performance with CVT-based method according to the angle values. The uniform density of our reconstructed mesh is better than others, which is benefited from the distance field optimization.

We have discussed that the flexible simplification is used to enhance sharp edges with distance field optimization. With the enhanced simplification result, the sharp edges can be kept in the reconstructed mesh. We compare the reconstructed results with sharp edges by different methods in Fig. 10. The results show that our method retains the sharp edges with fewer discontinuities and smoothness.

6.3 Comprehensive analysis

Combined with previous experiments, we provide a comprehensive analysis for facilitate understanding of the advantages of our method. For geometric consistency, our method achieves better result which is benefited from the distance field optimization in our framework. The simplification for the optimization dose not change the positions of most point. On the contrary, the Screen Poisson, Ball pivoting and CVT-based methods move points in the reconstructed mesh which reduce the accuracy of geometric consistency. For mesh

Table 1 Comparisons of Hausdorff distances by different methods. The models (Child to WoodMan) are reconstructed without change of point number, the up-sampling is used in our method; other models are resampled into 10,000 or 20,000 (Dragon) points in the reconstruction

Models methods	Ball pivoting	ScaleSpace	Advancing Delaunay	Screen Poisson	CVT	Neural-Pull	Ours
Child	0.02019	0.03034	0.03265	0.03639	0.20681	0.09721	**0.01941**
Deer	0.01836	0.02396	0.02656	0.03494	0.18631	0.08623	**0.01625**
Dog	0.02098	0.02804	0.02591	0.03245	0.13151	0.04782	**0.01151**
Elephant	**0.02008**	0.02504	0.02772	0.03715	0.23331	0.05762	0.02639
Girl	0.02863	0.03795	0.02461	0.02599	0.10562	0.07861	**0.02653**
Orangutan	0.02177	0.04112	0.03898	0.04472	0.07475	0.09533	**0.02009**
Weedle	**0.02098**	0.03921	0.03141	0.03572	0.03888	0.08719	0.03544
WoodMan	0.01721	0.02299	0.02156	0.02407	0.03366	0.04792	**0.01505**
Angle	**0.00241**	0.00314	0.00569	0.00631	0.00544	0.00407	0.00259
Armadillo	0.00287	0.00506	0.00433	0.00489	0.00564	0.00751	**0.00236**
Bunny	0.00313	0.00546	0.00939	0.00837	0.00822	0.00891	**0.00253**
Dragon	0.00278	0.00329	0.00418	0.00546	0.00769	0.00467	**0.00281**
Hourse	0.00285	0.00331	0.00458	0.00328	0.00486	0.00672	**0.00269**

The bold emphasis means that the quantitative results of our method are better than other solutions according to the geometric consistency

quality estimation, the CVT-based method achieves better performance for inner angle optimization. However, the time cost of the optimization in Voronoi diagram is huge and the edges are crashed in the regions of sharp features in a high probability (instance shown in Fig. 10). Without distance field optimization, the error connections between different points can not be avoided in the reconstructed mesh, which influences the performance of methods, including Ball pivoting, Scale space, and Advancing Delaunay.

For time cost report, we compare the different methods in Tables 3 and 4. In SHREC models, the up-sampling is used to keep the point number in our method which increases
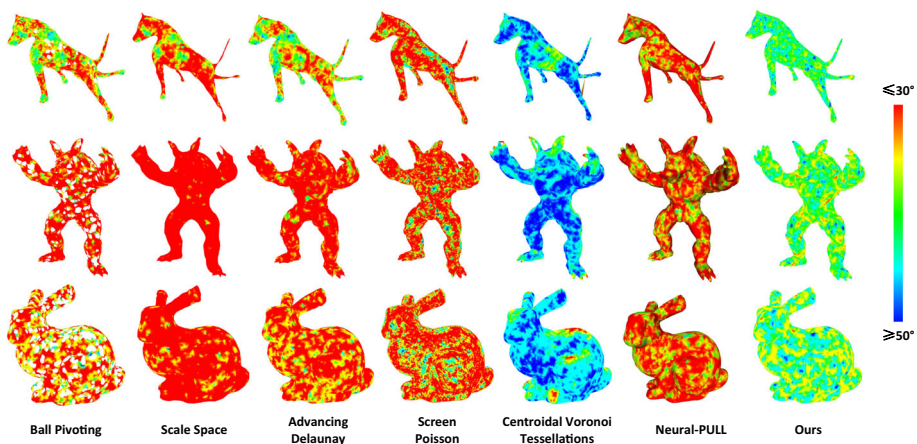

Fig. 9 Color maps for average values of inner angles by different method

Table 2 Comparisons of average values of inner angles and density parameters in the reconstructed meshes by different methods

Methods	Ball pivoting		ScaleSpace		Advancing Delaunay		Screen Poisson		CVT		Neural-Pull		Ours	
Models	Angle	D_r	Angle	D_r	Angle	D_r	Angle	D_r	Angle	D_r	Angle	D_r	Angle	D_r
Child	33.44°	0.033	31.84°	0.044	35.34°	0.043	32.41°	0.029	**47.21°**	0.149	33.01°	0.025	42.08°	**0.016**
Deer	34.41°	0.031	33.39°	0.034	36.56°	0.036	32.42°	0.019	**47.29°**	0.179	32.91°	0.021	43.86°	**0.011**
Dog	37.73°	0.035	35.62°	0.037	39.61°	0.037	32.34°	0.024	**47.95°**	0.291	32.82°	0.023	43.38°	**0.011**
Elephant	34.11°	0.032	33.06°	0.037	35.91°	0.045	32.87°	0.031	**46.97°**	0.542	31.93°	0.025	41.92°	**0.018**
Girl	36.58°	0.032	34.56°	0.041	38.73°	0.037	32.41°	0.024	**47.62°**	0.071	32.48°	0.025	42.56°	**0.015**
Orangutan	37.53°	0.042	34.85°	0.059	38.42°	0.059	32.07°	0.029	**47.16°**	0.054	33.26°	0.028	41.27°	**0.019**
Weedle	33.55°	0.036	32.44°	0.043	34.38°	0.042	32.07°	0.031	**47.52°**	0.031	32.12°	0.029	41.44°	**0.016**
WoodMan	33.11°	0.024	33.45°	0.031	36.61°	0.026	32.58°	0.014	**48.69°**	0.059	31.29°	0.021	43.91°	**0.008**
Angle	29.05°	0.041	30.65°	0.048	30.83°	0.071	32.18°	0.054	**46.85°**	0.058	32.31°	0.048	42.69°	**0.018**
Armadillo	31.05°	0.051	30.06°	0.057	31.74°	0.058	32.24°	0.035	**47.02°**	0.034	33.28°	0.034	41.59°	**0.016**
Bunny	33.94°	0.052	32.86°	0.071	33.85°	0.096	32.15°	0.032	**47.22°**	0.048	32.57°	0.041	41.47°	**0.031**
Dragon	30.51°	0.034	28.99°	0.049	31.09°	0.065	32.64°	0.046	**46.45°**	0.079	33.26°	0.035	41.68°	**0.022**
Hourse	32.81°	**0.039**	31.95°	0.051	33.08°	0.059	32.52°	0.048	**45.71°**	0.058	33.12°	0.041	42.25°	0.041

The results show that our method achieves similar isotropic properties to CVT in an efficient way (Table 4)
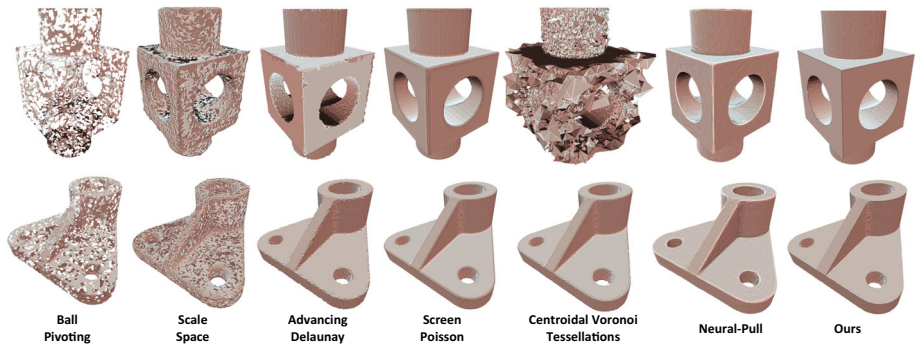
Fig. 10 Comparisons of reconstructed meshes with sharp edges by different methods

the time cost. However, our method still achieves better performance than the CVT-based method. For Stanford models, the point cloud with different point number are reported separately. The point number is specified to 10,000 in the reconstruction. It is clear that the time cost of our method is reduced. Even the time cost of resampling and simplification is not considered, our method is still fast than Advancing Delaunay with same triangulation strategy. It means that the optimized point cloud improved the reconstruction process. The Screen Poisson achieves fastest speed in the reconstruction. However, it requires the computation of normal vectors. The point number is controlled in a rough range which limits the scope in related applications. The Neural-PULL uses similar reconnection strategy with Screen Poisson while considering prior data-based experience. It can reconstruct more accurate geometric details. However, the processing requires more iterations that reduce efficiency.

Although there have many advantages of our method, some limitations still exist. The simplification in our framework achieves an approximate isotropic result from the input point cloud. It is not a strict isotropic one. The quality of the reconstructed mesh from our optimized point cloud is worse than CVT-based method (some results are shown in Table 2). For geometric consistency, the up-sampling process adds some new points into the point cloud which reduce the accuracy. In Table 1, the Hausdorff distances of our method in some models are worse than the results of Ball pivoting. Nonetheless, our method still has excellent utility in practice. It is a balance strategy for quality and speed, while keeping the geometric feature as much as possible.

7 Conclusions

We have proposed a Subdivision-based framework for point cloud-based shape reconstruction. The method includes distance field optimization and mesh generation. For distance

Table 3 Comparisons of average time cost for SHREC models by different methods. For Screen Poisson, the time cost of normal detection is added (green); for our method, the time cost of simplification with up-sampling is added (red)

Methods	Ball pivoting	ScaleSpace	Advancing Delaunay	Screen Poisson	CVT	Neural-Pull	Ours
Time	11.54 s	23.48 s	9.19 s	0.564 s + 2.69 s	591.31 s	1103.21 s	30.12 s + 8.46 s

Table 4 Comparisons of average time cost for Stanford models by different methods. For Ball Pivoting, ScaleSpace and Advancing Delaunay, the time cost of resampling is also reported (blue); for Screen Poisson, the the time cost of normal detection is added (green); for our method, the time cost of simplification is added (red)

Methods	Ball pivoting	ScaleSpace	Advancing Delaunay	Screen Poisson	CVT	Neural-Pull	Ours
50,000	62.35 s + 15.89 s	62.35 s + 24.27 s	62.35 s + 8.78 s	1.11 s + 1.26 s	623.56 s	1081.12 s	4.56 s + 7.56 s
100,000	346.15 s + 17.35 s	346.15 s + 24.88 s	346.15 s + 8.86 s	5.38 s + 3.56 s	756.26 s	1089.22 s	5.12 s + 7.81 s
200,000	455.72 s + 19.22 s	455.72 s + 24.52 s	455.72 s + 8.96 s	7.19 s + 3.98 s	636.15 s	1091.23 s	5.36 s + 8.54 s
500,000	1382.35 s + 78.28 s	1382.35 s + 48.34 s	1382.35 s + 20.28 s	12.78 s + 4.26 s	865.36 s	1109.35 s	12.56 s + 22.15 s

field optimization, the subdivision structure is established to divides a point cloud into different voxel boxes, while keeps the topological constraints by adjacent voxel boxes. Based on the structure, the simplification scheme is implemented to optimize the point cloud. The sparse and nonuniform point clouds can be processed by up-sampling before simplification. The sharp features can be enhanced by the flexible simplification. The mesh generation establishes the mesh based on the optimized point cloud. With the mesh cropping, the accuracy of the reconstruction is improved. The experimental data show that our method is better than classical methods for the balance between geometric consistency and accuracy of local regions. The point number is controlled and sharp edges can be rebuild in the final result.

In future works, we will improve the quality of isotropic density in distance field optimization. The triangulation also should be enhanced to fit more strict manifold property while keeping the sharp edges and regions with significant curvature changes.

Declarations

Conflict of interest No potential conflict of interest was reported by the authors.

References

1. Alexa M, Behr J, Cohen D, Fleishman S, Levin D (2001) Point set surfaces. In: Visualization, pp 21–28
2. Alsnayyan A, Shanker B (2022) Laplace-beltrami based multi-resolution shape reconstruction on subdivision surfaces. J Acoust Soc Am 151(3):2207–2222
3. Amenta N, Bern M, Kamvysselis M (1998) A new voronoi-based surface reconstruction algorithm. In: Computer graphics and interactive techniques, pp 415–421
4. Bernardini F, Mittleman J, Rushmeier H, Silva C, Taubin G (1999) The ball-pivoting algorithm for surface reconstruction. IEEE Trans Vis Comput Graph 5(4):349–359
5. Berger M, Levine JA, Nonato LG, Taubin G, Silva CT (2013) A benchmark for surface reconstruction. ACM Trans Graph (TOG) 32(2):1–17
6. Bhople AR, Shrivastava AM, Prakash S (2021) Point cloud based deep convolutional neural network for 3d face recognition. Multimed Tools Appl 80(20):30237–30259
7. Bronstein A, Bronstein M, Castellani U, Falcidieno B, Fusiello A, Godil A, Guibas L, Kokkinos I, Lian Z, Ovsjanikov M et al (2010) Shrec 2010: robust large-scale shape retrieval benchmark. Proc 3DOR 5(4)
8. Chen Z, Zhang T, Cao J, Zhang Y, Wang C (2018) Point cloud resampling using centroidal voronoi tessellation methods. Comput-Aided Des 102:12–21
9. Cohen-Steiner D, Da F (2004) A greedy delaunay-based surface reconstruction algorithm. Vis Comput 20(1):4–16
10. Corsini M, Cignoni P, Scopigno R (2012) Efficient and flexible sampling with blue noise properties of triangular meshes. IEEE Trans Vis Comput Graph 18(6):914–924

11. Digne J, Morel J, Souzani C, Lartigue C (2011) Scale space meshing of raw data point sets. Comput Graph Forum 30(6):1630–1642
12. Duan Y, Qin H (2004) A subdivision-based deformable model for surface reconstruction of unknown topology. Graph Model 66(4):181–202
13. Fuhrmann S, Goesele M (2014) Floating scale surface reconstruction. ACM Trans Graph (ToG) 33(4):1–11
14. Huang H, Li D, Zhang H, Ascher U, Cohen-Or D (2009) Consolidation of unorganized point clouds for surface reconstruction. ACM Trans Graph 28(5):1–7
15. Kazhdan M, Hoppe H (2013) Screened poisson surface reconstruction. ACM Trans Graph 32(3):1–13
16. Krishnamurthy V, Levoy M (1996) Fitting smooth surfaces to dense polygon meshes. In: Proceedings of the 23rd annual conference on computer graphics and interactive techniques, pp 313–324
17. Lafarge F, Alliez P (2013) Surface reconstruction through point set structuring. Comput Graph Forum 32:225–234
18. Li W, Han D, Li H, Wang X, Zhu J (2018) Extraction of digital terrain model based on regular mesh generation in mountainous areas. Multimed Tools Appl 77(5):6267–6286
19. Lv C, Wu Z, Wang X, Zhou M, Toh K-A (2019) Nasal similarity measure of 3d faces based on curve shape space. Pattern Recogn 88:458–469
20. Lv C, Lin W, Zhao B (2021) Voxel structure-based mesh reconstruction from a 3d point cloud. IEEE Trans Multimed 24:1815–1829
21. Lv C, Lin W, Zhao B (2021) Approximate intrinsic voxel structure for point cloud simplification. IEEE Trans Image Process 30:7241–7255
22. Lv C, Lin W, Zhao B (2022) Intrinsic and isotropic resampling for 3d point clouds. IEEE Trans Pattern Anal Mach Intell. https://doi.org/10.1109/TPAMI.2022.3185644
23. Lv C, Lin W, Zheng J (2022) Adaptively isotropic remeshing based on curvature smoothed field. IEEE Trans Visual Comput Graph. https://doi.org/10.1109/TVCG.2022.3227970
24. Ma B, Han Z, Liu Y-S, Zwicker M (2021) Neural-pull: learning signed distance function from point clouds by learning to pull space onto surface. In: International conference on machine learning, pp 7246–7257. PMLR
25. Moenning C, Dodgson A (2003) Fast marching farthest point sampling for implicit surfaces and point clouds. Comput Lab Tech Rep 565:1–12
26. Pal P, Ballav R (2007) Object shape reconstruction through nurbs surface interpolation. Int J Prod Res 45(2):287–307
27. Rakotosaona M-J, La Barbera V, Guerrero P, Mitra NJ, Ovsjanikov M (2020) Pointcleannet: learning to denoise and remove outliers from dense point clouds. In: Computer graphics forum, vol 39. Wiley Online Library, pp 185–203
28. Verykokou S, Ioannidis C, Athanasiou G, Doulamis N, Amditis A (2018) 3d reconstruction of disaster scenes for urban search and rescue. Multimed Tools Appl 77(8):9691–9717
29. Zhang G, Ge Y, Dong Z, Wang H, Zheng Y, Chen S (2021) Deep high-resolution representation learning for cross-resolution person re-identification. IEEE Trans Image Process 30:8913–8925
30. Zhang G, Yang J, Zheng Y, Wang Y, Wu Y, Chen S (2021) Hybrid-attention guided network with multiple resolution features for person re-identification. Inf Sci 578:525–538
31. Zhang N, Zhao J, Duan F, Pan Z, Wu Z, Zhou M, Gu X (2022) An end-to-end conditional generative adversarial network based on depth map for 3d craniofacial reconstruction. In: Proceedings of the 30th ACM international conference on multimedia, pp 759–768
32. Zheng X, Zhao J, Lv Z, Duan F, Pan Z (2020) Skull similarity comparison based on spca. Multimed Tools Appl 79:22423–22446