Incremental Poisson surface reconstruction for large scale three-dimensional modeling

Qiang Yu[1,2][0000-0003-2119-090X], Wei Sui[2], Ying Wang[2], Shiming Xiang[1,2], and Chunhong Pan[2]

[1] School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing 100049, China

[2] National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China

Abstract. A novel Incremental Poisson Surface Reconstruction (IPSR) method based on point clouds and the adaptive octree is proposed in this paper. It solves two problems of the most popular Poisson Surface Reconstruction (PSR) method. First, the PSR is time and memory consuming when treating large scale scenes with millions of points. Second, the PSR can hardly handle the incremental reconstruction for scenes with newly arrived points, unless being restarted on all points. In our method, large scale point clouds are first partitioned into small neighboring blocks. By providing an octree node classification mechanism, the Poisson equation is reformulated with boundary constraints to achieve the seamless reconstruction between adjacent blocks. Solving the Poisson equation with boundary constraints, the indicator function is obtained and the surface mesh is extracted. Experiments on different types of datasets verify the effectiveness and the efficiency of our method.

Keywords: Surface reconstruction · Large scale point cloud · Incremental.

1 Introduction

Surface reconstruction is a widely studied problem in fields of computer graphics, and is significant for applications such as Augmented Reality (AR), City Digitalization and 3D Printing, *etc.*. Surface reconstruction from point clouds is very challenging since the point clouds obtained from scanning or image based methods are usually unorganized, noisy, data-missing or with misregistration.

With the efforts of researchers, numbers of surface reconstruction methods have been proposed in the last two decades, which can be roughly clustered into two categories, *i.e.*, Computational Geometry (CG) based methods [3] [6] [1] and Implicit Function Fitting (IFF) based methods [4] [8] [9]. The CG based methods attempt to directly recover geometry structures from point clouds. These methods are quite fast, but they are essentially local algorithms, and hence can not fill holes and are liable to be affected by noise.

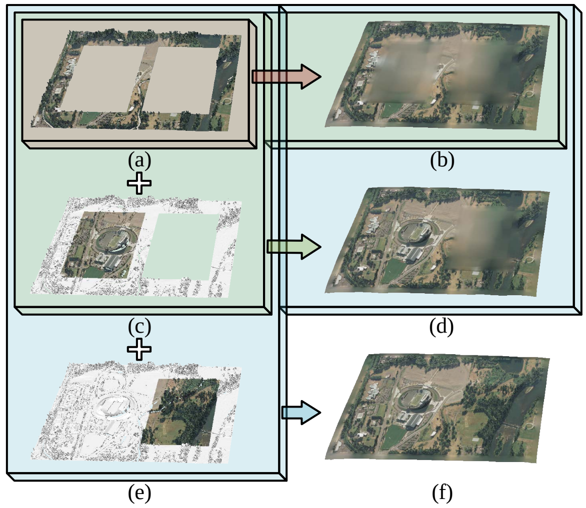

Fig. 1: Our IPSR method can incrementally and seamlessly reconstruct surface meshes from a series of neighboring point cloud blocks. Compared to existent methods (for instance the PSR method), the proposed method does not need to process all points from scratch repeatedly whenever new points are provided. (a) A skeleton point cloud containing two blank regions without points. (b) The reconstruction result of (a). (c) A new inner point cloud block which fills one of the blank regions in (a). (d) The incremental reconstruction result of (c) on the basis of (b). (e) Another new inner point cloud block like (c). (f) The incremental reconstruction result of (e) on the basis of (d).

Despite the effectiveness of the IFF based methods, the low efficiency and heavy computational load prevent their applications for large scale scenes. Besides, these methods can only deal with all points simultaneously.

To address the preceding problems, we propose an improved version of Poisson Surface Reconstruction (PSR) [9], called Incremental Poisson Surface Reconstruction (IPSR). Two main modifications are made compared to PSR: 1) Original octree is replaced by an adaptive octree which can be expanded flexibly when new points are provided; 2) Boundary constraints are integrated to Poisson equation, which guarantee the overall implicit function to be seamless. In our method, instead of repairing surface meshes after mesh fusion [15] [11] [5], the overall implicit function is reconstructed seamlessly in a divided and progressive way (shown in Fig. 1). The underlying mathematical model behind our method is a Poisson equation with well designed boundary constraints.

The main contributions of our method can be summarized as follows:

– An incremental surface reconstruction method is proposed for large scale scenes where point clouds are provided online and area by area. The proposed method is quite flexible and resource saving with the comparable reconstruc-

tion accuracy to the original PSR method, which is a popular benchmark of surface reconstruction.
- A novel Poisson equation with boundary constraints is formulated based on the adaptive octree, with which neighboring point cloud blocks can be reconstructed incrementally and seamlessly.
- An octree node classification method is designed to classify octree nodes into inner and boundary types. The inner nodes help reconstruct implicit functions while the boundary nodes provide boundary constraints.

2 Related Work

2.1 Surface Reconstruction

The CG based methods, such as Delaunay triangulation [3], Alpha shapes [6] and Voronoi diagram [1], are early but effective ones for surface reconstruction. These methods directly reconstruct 2D triangles or 3D tetrahedrons by interpolating the whole or subset of points. Thus, they are quite fast and easy to implement. Furthermore, they do not need any prior assumption or auxiliary information about the scene. The main drawback of these methods is that noises and outlier points are taken into consideration during the surface reconstruction process, which will result in seriously bad results when the quality of points is poor.

Afterwards, the IFF based methods have been proposed to improve the robustness of surface reconstruction algorithm. These methods are designed to fit a scalar three-dimensional spacial implicit function or calculate signed distance field to points to represent the model, and then extract the surface as a level set of the implicit function. The implicit functions can be represented as the weighted sum of radial basis functions or piece-wise polynomial functions. The final watertight and manifold mesh will be obtained through marching cubes algorithm [10].

2.2 Poisson Surface Reconstruction

The PSR method is developed under the IFF framework . The main idea of the PSR method is that the implicit function can be estimated as the indicator function (whose value is 1 inside the surface and 0 outside the surface) of the model. The smoothed gradient of the indicator function corresponds to the divergence of normal vector field, which can be approximated by a summation over the oriented points.

We begin by reviewing the PSR method concisely. Let $\mathcal{V} : \mathbb{R}^3 \mapsto \mathbb{R}^3$ be the normal vector field of input oriented points, and $\mathcal{X} : \mathbb{R}^3 \mapsto \mathbb{R}^1$ be the indicator function. Then the problem is formed as a Poisson equation:

$$\Delta \mathcal{X} = \nabla \cdot \mathcal{V}, \tag{1}$$

where Δ is the Laplace operator and $\nabla \cdot \mathcal{V}$ represents the divergence of normal vector field $\mathcal{V}$.

The implicit function in the PSR method is represented as the weighted sum of a set of multiresolution Gaussian functions constructed on nodes of the octree. Suppose $\mathcal{F}_i : \mathbb{R}^3 \mapsto \mathbb{R}^1$ is the Gaussian function attached to the i-th node in the octree, and then its value at point $q \in \mathbb{R}^3$ can be represented as

$$\mathcal{F}_i(q) = \mathcal{F}\left(\frac{q - \mathbf{c}_i}{\mathbf{w}_i}\right)\frac{1}{\mathbf{w}_i^3}, \tag{2}$$

where $\mathcal{F} : \mathbb{R}^3 \mapsto \mathbb{R}^1$ is the standardized Gaussian function, $\mathbf{c}_i$ and $\mathbf{w}_i$ are the center and width of the i-th node. Correspondingly, the indicator function and the normal vector field can be represented as

$$\mathcal{X} = \sum_{i}^{n} \mathbf{x}_i \mathcal{F}_i = \mathbf{x}^\mathsf{T}\mathbf{F}, \tag{3}$$

$$\mathcal{V} = \sum_{i}^{n} \mathbf{v}_i \mathcal{F}_i = \mathbf{v}^\mathsf{T}\mathbf{F}, \tag{4}$$

respectively, where n is the number of octree nodes, $\mathbf{x} \in \mathbb{R}^{n\times 1}$ and $\mathbf{v} \in \mathbb{R}^{n\times 3}$ are the coefficients vectors of indicator function and the divergence of normal vector field respectively, and $\mathbf{F}$ is a column vector of all node functions.

In Eq. (4), $\mathbf{v}_i \in \mathbb{R}^3$ is the vector held by the i-th node in the octree. It is calculated using the normal vectors of the points in the point cloud by the following function

$$\mathbf{v}_i = \sum_{p \in \mathsf{Ng}(o_i)} \frac{1}{\mathsf{Ds}(p)}\ \alpha_{p,o_i}\ p.\mathbf{n}, \tag{5}$$

where p is a point in the point cloud, o_i is the i-th node, $\mathsf{Ng}(o_i)$ is the point set bounded in the $3\times3\times3 = 27$ neighboring nodes of o_i, $\mathsf{Ds}(p)$ is the point density at p which is estimated as the number of points in its neighborhood [9], α_{p,o_i} is the trilinear interpolation weight of o_i among $2\times2\times2 = 8$ nearest nodes around p and $p.\mathbf{n}$ is the normal vector held by the point p.

Combined with Eq. (2), Eq. (3) and Eq. (4), Eq. (1) can be reformulated as an linear expression as follows

$$\mathbf{L}\mathbf{x} = \mathbf{v}, \tag{6}$$

where $\mathbf{L} \in \mathbb{R}^{n\times n}$ is the Laplace matrix defined as

$$\mathbf{L}_{i,j} = \left\langle \frac{\partial^2 \mathcal{F}_i}{\partial x^2}, \mathcal{F}_j \right\rangle + \left\langle \frac{\partial^2 \mathcal{F}_i}{\partial y^2}, \mathcal{F}_j \right\rangle + \left\langle \frac{\partial^2 \mathcal{F}_i}{\partial z^2}, \mathcal{F}_j \right\rangle. \tag{7}$$

After obtaining the coefficients vector $\mathbf{x}$, the indicator function can be easily calculated.

However, based on global Poisson equation, the PSR method easily results in over-smoothing of the input points. Another problem is that the PSR method can not deal with out-of-core reconstruction.

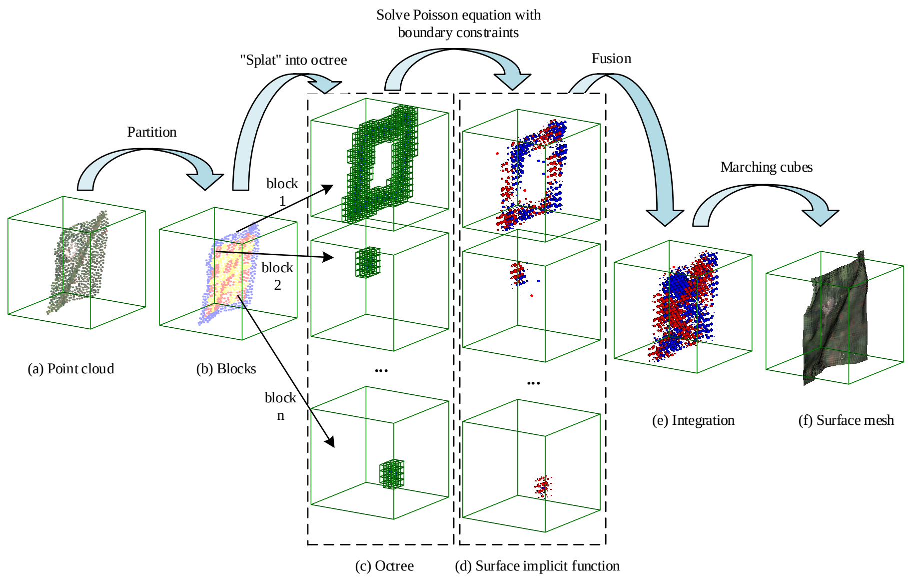

Fig. 2: The pipeline of our IPSR method. (a) The whole point cloud. (b) The partitioned blocks shown in different colors. (c) The parts of octree, on which the Poisson equations with boundary constraints are constructed and then solved. (d) The solved indicator function. We illustrate the indicator function by a series of spheres in 3D space. (e) The integration of the indicator function. (f) The surface mesh extracted from the octree using the marching cubes algorithm.

2.3 Incremental Surface Reconstruction

Although the PSR method does well in handling noisy data, it suffers from a limitation that all points should be present before performing the surface reconstruction process. Hence, some methods have been proposed to address this problem.

Newcombe *et al.* [12] incrementally constructed a truncated signed distance function (TSDF) using the input points of each scan. Nico *et al.* [14] incrementally refined the coarse base mesh using field-aligned method. These existing methods have a similar application of reconstructing small models from RGB-D images by indoor scanners.

In contrast, we propose the IPSR method based on the PSR method (and compatible with the SPSR method), which aims at incrementally reconstructing large scale scenes and accepts a series of neighboring but no-overlap-required point clouds.

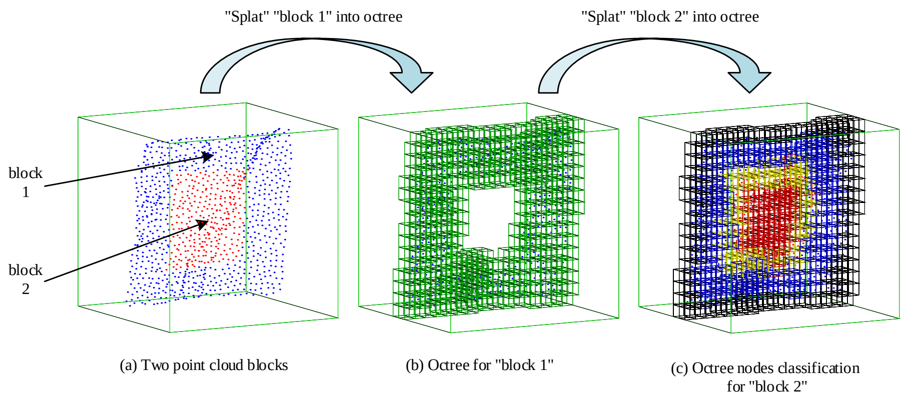

Fig. 3: The proposed octree nodes classification method. Left: the input two point cloud blocks, where different colors represents different blocks. Middle: the octree after the reconstruction of the blue block. Right: the new octree when reconstructing the red block. The nodes of categories 1, 2, 3 and 4 are colored in red, yellow, blue and black. The nodes of category 5 are ignored for clarity.

3 The proposed method

In this section, we describe the proposed IPSR method for surface reconstruction from oriented points. Our method is specially designed such that it can perform the reconstruction process in an incremental manner. It is very flexible and resource saving with comparable reconstruction accuracy to the original PSR method.

3.1 Motivation

In the original PSR method [9], the coefficients vector of the indicator function $\mathbf{x} \in \mathbb{R}^n$ is obtained by solving Eq. (6), where n is the total number of octree nodes. When the scale of the scene grows, the number of octree nodes become larger accordingly, which makes solving the linear system more time-consuming. Besides, when the scene of points are obtained multiple times, the linear system have to be constructed and solved repeatedly, which is resource wasteful.

Our method aims at reconstructing large scale scene incrementally, such as landscape or urban regions point cloud from the LiDAR on aircraft *etc.*. There are two situations which we pay attention to. First, the points are provided online, that is, one point cloud is provided each time the aircraft flies a line. Thus, we should incrementally reconstruct a series of point clouds from different flight lines. Second, all points are provided offline. In this situation, in order to make the use of our IPSR method, the point cloud should be first partitioned into a series of neighboring blocks. Fig. 2 shows the pipeline of our method.

Comparing with the PSR method, the proposed IPSR method in incremental reconstruction mode requires less time and space.

3.2 Point Cloud Partition

When applying the proposed IPSR method to one single large scale point cloud, the whole point cloud needs to be partitioned into a number of blocks of arbitrary size. To make the full use of boundary constraints, we create two conditions to guide the point cloud partition progress.

A good partition should satisfy two conditions:

1. The point density in boundary regions should be as high as possible;
2. The underlying surface in boundary regions should be as smooth as possible.

3.3 Octree Nodes Classification

After dividing the point cloud into different blocks, each block is "splatted" into the octree and a corresponding Poisson equation is constructed on inner nodes while the boundary constraints are imposed on intra nodes. In this subsection, we classify the octree nodes to determine whether they should be taken into the reconstruction. According to the influence from the new coming points, the corresponding octree nodes can be classified into five categories as follows:

1. Nodes that are newly created.
2. Nodes where corresponding normal vector contributions are updated.
3. Nodes that are not included in categories 1 and 2 but the corresponding node functions interact with that of the nodes in categories 1 and 2.
4. Nodes that are not included in categories 1, 2 and 3 but the corresponding node functions interact with that of the nodes in categories 1, 2 and 3.
5. The rest nodes.

Naturally, the coefficients of nodes in the category 1 need to be calculated. In addition, the coefficients of nodes in categories 2 and 3 should be recalculated since the variation of the nodes' normal vector contributions and the corresponding items of the Laplace matrix. The nodes in category 4 are used for boundary constraints, because both their normal vector contributions and their corresponding items of the Laplace matrix keep unchanged during the reconstruction progress for the current block. In the following, we call the nodes in categories 1, 2 and 3 the Poisson nodes, call the nodes in category 4 the boundary nodes and call the nodes in category 5 the unused nodes. An example is given in Fig. 3 to demonstrate the classification results.

3.4 Incremental Reconstruction with Boundary Constraints

The underlying model of the proposed IPSR method is the Poisson equation with boundary constraints.

Let $\mathbf{x}_p \in \mathbb{R}^{n_p \times 1}$ be the coefficients vector of the Poisson nodes, $\mathbf{x}_b \in \mathbb{R}^{n_b \times 1}$ be the coefficients vector of the boundary nodes and $\mathbf{x}_u \in \mathbb{R}^{n_u \times 1}$ be the coefficients vector of the unused nodes, where the n_p, n_b and n_u are number of nodes in corresponding categories. We stack $\mathbf{x}_p$, $\mathbf{x}_b$ and $\mathbf{x}_u$ into a uniform vector $\mathbf{x}$, which is given by:

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}_p \\ \mathbf{x}_b \\ \mathbf{x}_u \end{bmatrix}. \tag{8}$$

Let $\mathbf{x}_b^*$ and $\mathbf{x}_u^*$ denote the true values of $\mathbf{x}_b$ and $\mathbf{x}_u$, which are used as the boundary constraints. Accordingly, the linear system of our IPSR method can be rewritten as:

$$\begin{cases} \begin{bmatrix} \mathbf{L}_p \ \mathbf{L}_b \ \mathbf{L}_u \end{bmatrix} \begin{bmatrix} \mathbf{x}_p \\ \mathbf{x}_b \\ \mathbf{x}_u \end{bmatrix} = \mathbf{v}_p \\ \mathbf{x}_b = \mathbf{x}_b^* \\ \mathbf{x}_u = \mathbf{x}_u^* \end{cases}, \tag{9}$$

where $\mathbf{L}_p \in \mathbb{R}^{n_p \times n_p}$ is the Laplace matrix of the Poisson node functions, $\mathbf{L}_b \in \mathbb{R}^{n_p \times n_b}$ is the pseudo Laplace matrix of the Poisson node functions against the boundary node functions, $\mathbf{L}_u \in \mathbb{R}^{n_p \times n_u}$ is the pseudo Laplace matrix of the Poisson node functions against the unused node functions and thus $\mathbf{L}' = \begin{bmatrix} \mathbf{L}_p \ \mathbf{L}_b \ \mathbf{L}_u \end{bmatrix} \in \mathbb{R}^{n_p \times n}$ is the top n_p rows of the matrix $\mathbf{L} \in \mathbb{R}^{n \times n}$ in Eq. (6) and Eq. (7).

Since the node functions of nodes in category 5 do not interact with that of any nodes in categories 1, 2 and 3, we have:

$$\mathbf{L}_u \equiv \mathbf{0}. \tag{10}$$

By substituting Eq. (10) into Eq. (9), we get:

$$\mathbf{L}_p \mathbf{x}_p = \mathbf{v}_p - \mathbf{L}_b \mathbf{x}_b^*, \tag{11}$$

where $\mathbf{L}_p$ and $\mathbf{v}_p - \mathbf{L}_b \mathbf{x}_b^*$ can be calculated from the normal vector field. It can be seen that $\mathbf{x}_p$ can be easily solved by the methods such as the conjugate gradient algorithm [7].

4 Experiments

In this section, we evaluate our method for incremental surface reconstruction on two datasets. Although there are many new methods such as KinectFusion [12] and field-aligned method [14], the comparison is not shown in this paper because of the limitation of paper space. In addition, our method is mainly designed on top of the PSR method, so we conduct comparative experiments with the PSR method to further demonstrate the capacity of our method.

Two different datasets are employed to evaluate our method. One is that collected from [2], which consists of 4 small point clouds . These point clouds are uniformly sampled from the corresponding MPU models [13] and hence the ground truths can be provided. With the ground truths, this dataset can be

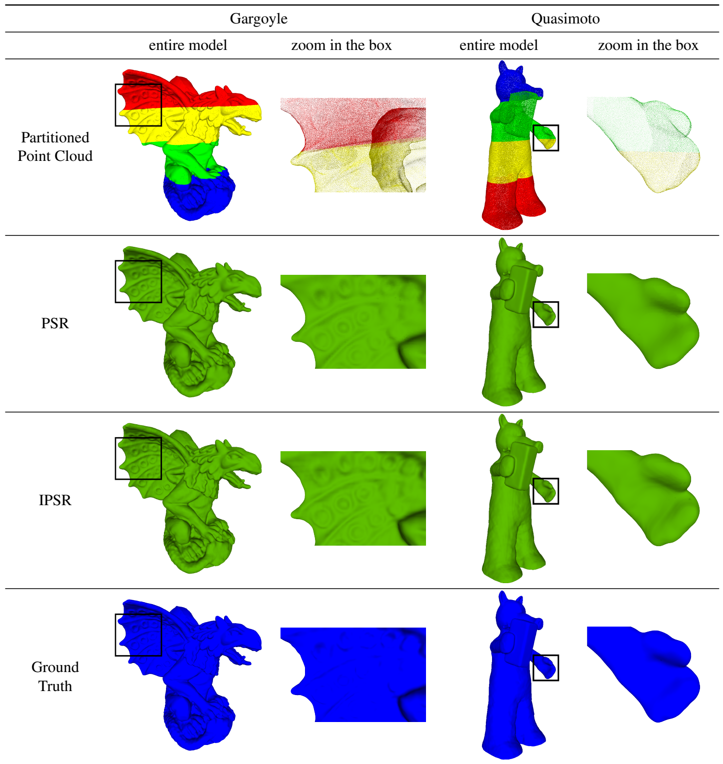

Fig. 4: The surface reconstruction results of point clouds in the benchmark dataset. Different colors indicate points in different blocks. We can see that the results of our method are nearly the same as the ground truths, and the boundary regions are natural and seamless.

used to conduct experiments for quantitative comparison and analysis. For convenience, we name it as the benchmark dataset.

Another is a dataset in which the point clouds are obtained from large scale outdoor scenes such as landscape, hill, valley and buildings . It consists of 6 large scale point clouds without ground truth. For convenience, we name it as the landscape dataset.

4.1 Reconstruction on Benchmark Dataset

In this subsection, we conduct qualitative and quantitative comparison experiments between our method and the original PSR method on the benchmark dataset. We simply partition the point clouds into several blocks uniformly and set the max depth of octree to 10 for all point clouds, which is deep enough to recover the details.

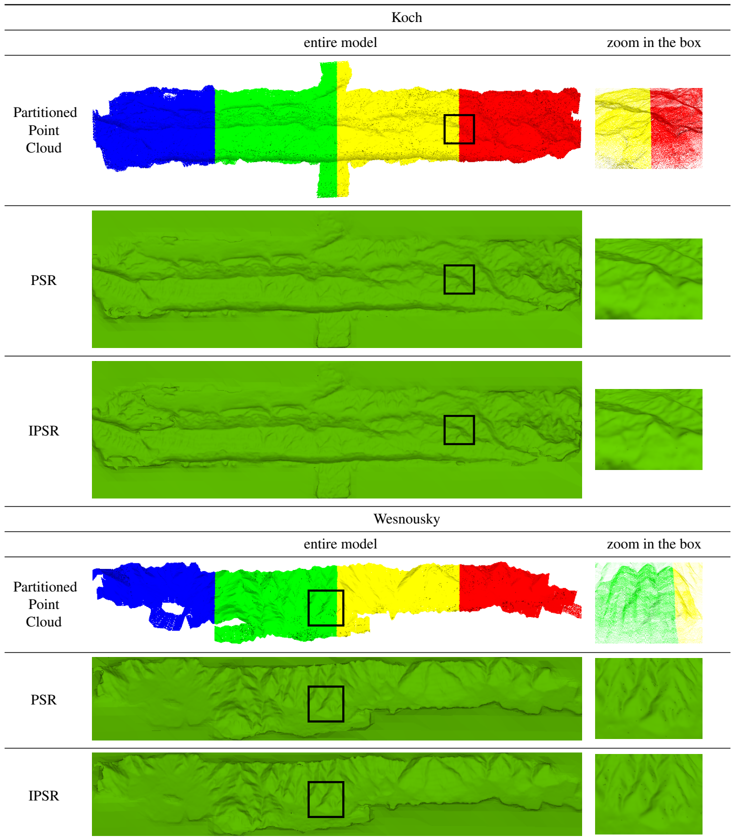

Fig. 5: The incremental surface reconstruction result of point clouds in the landscape dataset. Different colors indicate points in different blocks.

Qualitative Comparison The surface reconstruction results of the benchmark dataset are illustrated in Fig. 4 (only two models are shown due to space limitation) . We can see that the results of the proposed method are nearly the same as the ground truths. The boundary constraints constructed in Eq. (9) guarantee the seamless transition from one block to another.

Quantitative Comparison To further demonstrate the advantage of our method, we conduct a quantitative comparison to evaluate the reconstruction accuracy of our method and the PSR method. Two metrics in [2] are used to measure the reconstruction accuracy: the distance error and the angle error. This experiment is conducted on the benchmark dataset since it contains ground truth.

As expected, the error values of our method are very close to those of the PSR method. When the octree depth is set to 10, the distance error and the angle error are bounded in 0.1 *mm* and 5 *degrees* respectively.

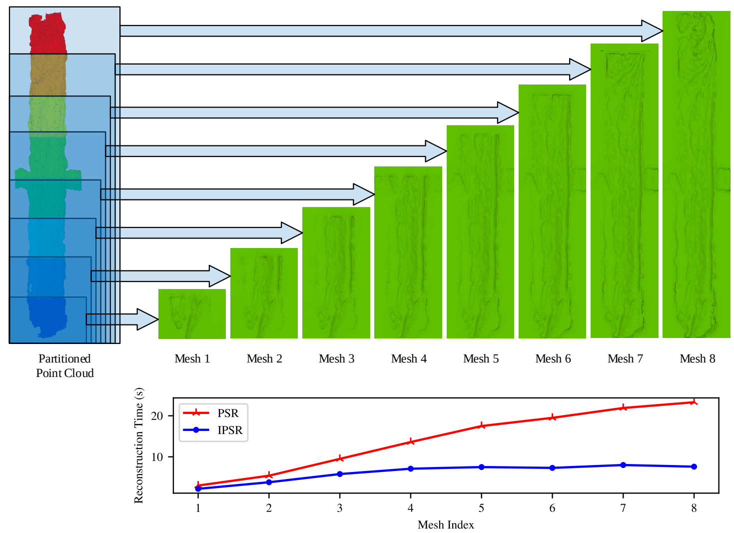

Fig. 6: The result of the incremental reconstruction for the landscape dataset. Our method saves a large amount of time than the PSR method.

4.2 Reconstruction on Landscape Dataset

In this subsection, we conduct qualitative comparison experiments between our method and the original PSR method on the landscape dataset, and show the great advantages in our incremental reconstruction method. It should be mentioned that this experiment is conducted not to prove that our method can deal with large scale data which the PSR method can not deal with, but to show that our method can perform faster and need less memory than the PSR method.

Reconstruction Result The surface reconstruction results of the landscape dataset are illustrated in Fig. 5 (only two models are shown due to space limitation) . From the reconstruction results we can see that the details are well maintained by our incremental reconstruction method.

Incremental Reconstruction Under the circumstances that the point cloud of entire scene can not be provided at one time, our method is more profitable than almost all the IFF based surface reconstruction methods for its flexibility and resource saving ability.

The incremental surface reconstruction process is illustrated in Fig. 6.

5 Conclusion

We have proposed an incremental surface reconstruction framework, specially for large scale scenes where the point clouds are provided sequentially. The underlying mathematical model of our method is the Poisson equation with boundary constraints. Comparative experiments on different datasets verify the advantages of our method.

References

1. Amenta, N., Bern, M., Kamvysselis, M.: A new voronoi-based surface reconstruction algorithm. In: Proceedings of the 25th annual conference on Computer graphics and interactive techniques. pp. 415–421. ACM (1998)
2. Berger, M., Levine, J.A., Nonato, L.G., Taubin, G., Silva, C.T.: A benchmark for surface reconstruction. ACM Transactions on Graphics (TOG) **32**(2), 20 (2013)
3. Boissonnat, J.D.: Geometric structures for three-dimensional shape representation. ACM Transactions on Graphics (TOG) **3**(4), 266–286 (1984)
4. Carr, J.C., Beatson, R.K., Cherrie, J.B., Mitchell, T.J., Fright, W.R., McCallum, B.C., Evans, T.R.: Reconstruction and representation of 3d objects with radial basis functions. In: Proceedings of the 28th annual conference on Computer graphics and interactive techniques. pp. 67–76. ACM (2001)
5. Dessein, A., Smith, W.A.P., Wilson, R.C., Hancock, E.R.: Seamless texture stitching on a 3d mesh by poisson blending in patches. In: IEEE International Conference on Image Processing. pp. 2031–2035 (2015)
6. Edelsbrunner, H., Mücke, E.P.: Three-dimensional alpha shapes. ACM Transactions on Graphics (TOG) **13**(1), 43–72 (1994)
7. Hestenes, M.R., Stiefel, E.: Methods of conjugate gradients for solving linear systems. Journal of Research of the National Bureau of Standards **49**(6), 409–436 (1952)
8. Kazhdan, M.: Reconstruction of solid models from oriented point sets. In: Eurographics Symposium on Geometry Processing. p. 73 (2005)
9. Kazhdan, M., Bolitho, M., Hoppe, H.: Poisson surface reconstruction. In: Eurographics Symposium on Geometry Processing. pp. 61–70 (2006)
10. Lorensen, W.E., Cline, H.E.: Marching Cubes: A high resolution 3d surface construction algorithm. Acm Siggraph Computer Graphics **21**(4), 163–169 (1987)
11. Min, J., Wei, X.: Robust merge of 3d textured meshes (2017)
12. Newcombe, R.A., Izadi, S., Hilliges, O., Molyneaux, D., Kim, D., Davison, A.J., Kohli, P., Shotton, J., Hodges, S., Fitzgibbon, A.W.: Kinectfusion: Real-time dense surface mapping and tracking. In: 10th IEEE International Symposium on Mixed and Augmented Reality, ISMAR 2011, Basel, Switzerland, October 26-29, 2011. pp. 127–136 (2011)
13. Ohtake, Y., Belyaev, A., Alexa, M., Turk, G., Seidel, H.P.: Multi-level partition of unity implicits. Acm Transactions on Graphics **22**(3), 463–470 (2003)
14. Schertler, N., Tarini, M., Jakob, W., Kazhdan, M., Gumhold, S., Panozzo, D.: Field-aligned online surface reconstruction. ACM Trans. Graph. **36**(4), 77:1–77:13 (2017)
15. Wuttke, S., Perpeet, D., Middelmann, W.: Quality preserving fusion of 3d triangle meshes. In: International Conference on Information Fusion. pp. 1476–1481 (2012)