IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 24, 2017, accepted January 17, 2018, date of publication February 12, 2018, date of current version April 23, 2018.

*Digital Object Identifier 10.1109/ACCESS.2018.2798640*

# A Novel Slicing-Based Regularization Method for Raw Point Clouds in Visible IoT

**YINGHUI WANG[1,2], LIJUAN WANG[1,3], WEN HAO[1,2], XIAOJUAN NING[1,2], ZHENGHAO SHI[1,2], AND MINGHUA ZHAO[1,2]**

[1]Xi'an University of Technology, Xi'an 710032, China
[2]Shaanxi Key Laboratory of Network Computing and Security Technology, Xi'an 710032, China
[3]Xi'an Technological University, Xi'an 710021, China

Corresponding author: Lijuan Wang (wanglijuan@xatu.edu.cn)

This work was supported in part by the Nature Science Foundation of China under Grant 61472319 and Grant 61602373, in part by the Shaanxi Science Research Plan under Grant 2015JZ015 and Grant 2017JQ6023, and in part by Shaanxi Education Department Key Laboratory Research Plan under Grant 14JS071.

**ABSTRACT** Three-dimensional scanning technology is becoming increasingly popular in the Internet of Things and cyber-physical social systems. Existing reconstruction algorithms and shape extraction algorithms usually require regularized points and clear topological relationships between them. In this paper, a novel slicing-based method is proposed to regularize raw point clouds, in which we first slice the raw point to get cross section points, and then generate regular point clouds under the topology control of cross section shapes, and finally produce regular point clouds at multiple levels of detail. The proposed method is effective specifically for huge raw point clouds and the experiment results show that the proposed method can produce regularized point clouds for surface meshing, shapes analysis, shapes extraction, and so on.

**INDEX TERMS** Cross section, point clouds, regularization, shape, slicing.

## I. INTRODUCTION

With the development of IoT and cyber-physical-social systems (CPSS), a large amount of 3D raw point clouds are collected by laser scanner in many applications, such as surface meshing, reverse engineering, shape analysis, object recognition, etc. The raw point clouds collected contain large amounts of cluttered, unorganized and irregular points. Due to the irregularity of the raw point clouds, many existing algorithms, such as Delaunay based surface reconstruction methods [1]–[3], marching cube [4], shape context [5], shape extraction [6]–[8], and the following algorithm [9]–[12], cannot generate valid results. The irregular points can result inaccurate topological relationship seeking, which is necessary in many surface reconstruction and shape extraction.

Irregularity of the raw point clouds is mainly caused by the laser scanning methods and the noise of raw points. As a result, irregularity of point clouds could be alleviated by some de-noised methods [13]–[15]. Unfortunately, it cannot be handled thoroughly by these methods.

In this paper, we propose a regularization method for raw point clouds. The main idea is that obtaining the cross section points of the raw point clouds, and generating a regular point set under the rigid topological control of cross section shapes. In the proposed method, two adjustment parameters $h$ and D are introduced to control the regular point density at different levels. Furthermore, the shapes of the input point clouds are maintained well, and the input point clouds are de-noised simultaneously. In the proposed method, the total point clouds are sliced into multiple parts. Each of them will be handled separately, and it is able to effectively process models with huge point clouds. Figure 1 shows the three main stages of proposed method, in which (a) shows the raw point clouds, (b) describes the slicing process of raw point clouds and (c) gives the output regular point clouds of different level of details.

The rest of this paper is organized as follows: Section 2 presents an overview of the proposed method; Section 3 addresses the detailed regularization algorithm; Section 4 compiles and evaluates the results; and Section conclusion concludes the findings of our method.

## II. RELATE WORK

In surface reconstruction, a number of point clouds consolidation methods have been proposed. They not only can de-noise point clouds but also can generate a relative regular representation of the input point clouds. Meanwhile, surface geometry of input point clouds is also kept. The well-known point clouds consolidation methods include moving least

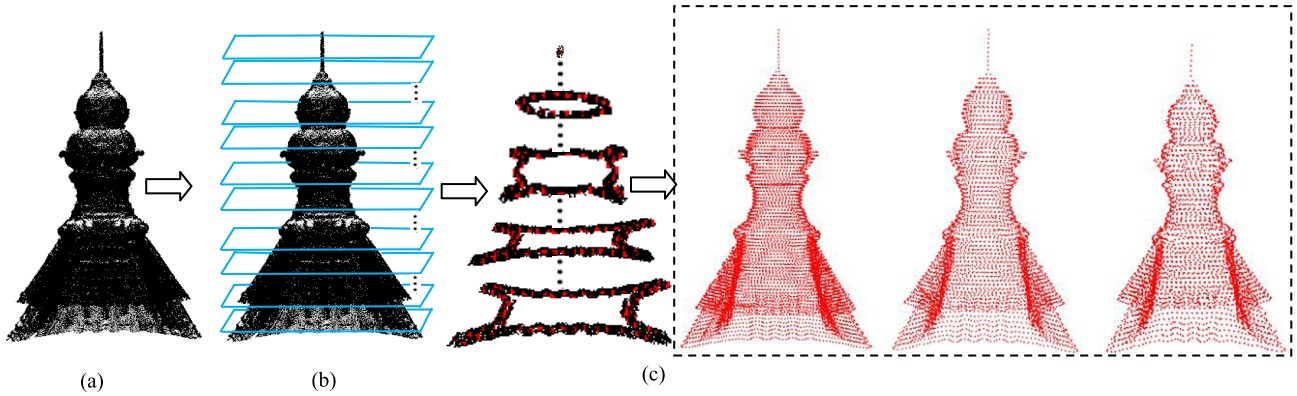



**FIGURE 1.** The profile of the proposed method. (a) Raw point clouds, (b) slicing of raw point clouds, (c) the middle points of each slice of the raw point clouds(the third from left), and the final regular representation (red) of raw point clouds in different level of details.

squares (MLS) based methods [20]–[24], locally optimal projection (LOP) methods [25]–[29] and others [17]. Essentially, de-noising, smoothing, and resample have been implemented during consolidation process.

MLS-based consolidation methods are based on differential geometry [20], [24]. Alexa et al. proposed a thin point set to adheres to the underlying shape [20]. They firstly defines a smooth manifold surface approximated by the MLS close to the original surface. Then the points are projected onto a 2D surface that approximates the points. Finally, the points that projected on themselves are proved to be the thinned points. Amenta and Kil [21] proposed a cloud of surfels (point equipped with normal) from point clouds when defining an extreme surface. Representative later works of MLS methods include an adaptive MLS method (AMLS) [22], algebraic point set surfaces [23], robust implicit MLS [24].

LOP methods use statistics to handle the consolidation problem. Lipman et al. introduce a multivariate L1-median based LOP operator [25]. Two cost functions are defined for the projection: one is to minimize the sum of weighted distance to point clouds P from output points of projection Q; another cost function is proposed to regularize the points in Q by incorporating local repulsion forces. The final points are solution of the minimization problems of the two functions. Huang et al. proposed a weighted locally optimal projection (WLOP) operator, that has solved the problem lead by the non-uniform samples in LOP in [26]. Recently, an iterative WLOP consolidation method [27] and an accelerated version of WLOP [28] are developed. The newly representative WLOP works include deep points consolidation [29].

The main goal of the consolidation methods is to improve the effects of surface reconstruction. The consolidation methods put more emphasis on improving surface attributes of points to guarantee the smoothing of surface. The objects that have sharp features such as CAD models are not involved in these methods. In these methods, whether the local shapes and global shapes of the input point clouds can be maintained accurately or not also needs extensive discussion.

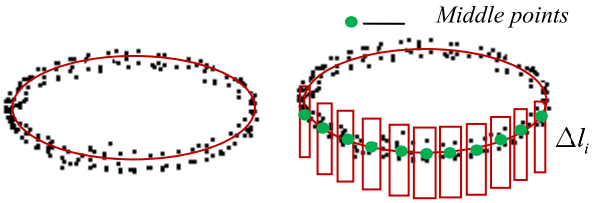



**FIGURE 2.** Strap-like point sets and their *middle points.*

Furthermore, MLS serial methods depend on oriented normal for the projection control. So, these methods are not robust if the noise is serious. In contrast, our method is more robust.

## III. REGULARIZATION OF UNORGANIZED POINT CLOUDS

In order to regularize the point clouds, cross section shapes are needed. The cross section shape is used to provide topological control for transforming the cluttered cross section points into a regular point set. Because the cross section points are strap-like, they are called strap-like point set in this paper. Given a cluttered strap-like cross section point set $C$, a regular point set M can be got by $F(T_C)$, where $T_C$ is the shape of cross section points $C$, F denotes the regularization function. According to [30] and [31], a high dimensional manifold can be represented accurately with resample points. So the regularization function F could be fulfilled by resample, as shown in figure 2. The cross section points set is divided into many small parts denoted by $\Delta l_i$ along the tangent direction (the red curve) i.e. $P = \sum_{\Delta l_i} \sum_{k \in \Delta l_i} p_k$. The centroid points of these small parts constitute the regular point set M. M is named as *middle points* in this paper. M of the all cross section points forms the regular representation of the total raw point clouds. The regularization process also can reduce the noise. Given a de-noised point set $P_1 = \{p_k\}_{k \in \Delta l_i}$ from a scanned surface slice $S_1$ and their *middle points* (figure 3(a)), if there is some noise that leads into two points $p_k$ and $p_{k+1}$ deviation from surface, as shown in figure 3(b), the centroid point $p_c'$ of noise point set will deviate the centroid point $p_c$ less.

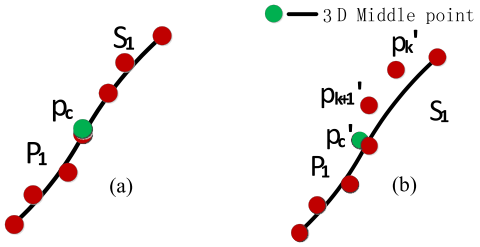



**FIGURE 3.** De-noised effects of *middle points.* (a) Clear points, (b) noise points.

The proposed regularization method has the following two steps:

1) Getting the cross section points by slicing and projection.

2) Generating the regular point set middle points with the regularization algorithm.

### A. SLICING AND PROJECTION OF POINT CLOUDS

Point clouds slicing is usually used in rapid prototype (RP) process in CAD area [32]–[35] and other applications [37], [38]. It is extended into regularization of point clouds in the proposed method. Slices of the point cloud are generated by moving two parallel planes that have the thickness $l$ along directional vector $\vec{n}$ at a step size $h$, as shown in figure 4(a). In each moving step, cross section points that lie between the two parallel planes are extracted. The thickness $l$ is defined by $L$ divided by a constant number N. $L$ is computed as the distance between extreme points along slicing directional vector $\vec{n}$. The slicing direction is user-defined. The value of N mainly depends on the precision of point clouds. If the precision is high, the thickness $l$ computed by a larger value of N can extract complex geometry with less redundant points are removed and vice versa. The step size $h$ mainly depends on the desired levels of results. If $h$ is small, the level of details will be high and vice versa. In our experiments, we set the value for N with 30, 50, 60, 70, etc.; $h$ is set with the value from 0.4 $l$ to 1.0 $l$.

After slicing, each slice of cross section points is projected into a projection plane that is parallel to slicing planes and is equidistant to the upper and lower slicing plane, as shown in figure 4(b).

### B. REGULARIZATION ALGORITHM

After slicing and projection, one or more 2D strap-like cross section point sets are generated. When there are many 2D strap-like cross section point sets in the same slice, they need to be segmented with clustering algorithms such as [36] and [43]. Here Density-Based Spatial Clustering of Applications with Noise (DBSCAN) clustering algorithm is employed. For each clustered 2D strap-like cross section points set, a grid-growing algorithm (GGA) [39] is used to generate the initial *middle points*. Figure 5 shows initial *middle point* of 2D strap-like cross section point sets with different shape.

In terms of the initial *middle points* there are two special cases:

#### 1) NON-UNIFORM WIDTH

If the slicing plane is not vertical to the sliced surface, the 2D strap-like point sets generated by projection may have non-uniform width shown in figure 6(a), which could affect the interval between middle points. Figure 6(b) shows the extreme case in getting the *middle points*. A single *middle point* (the green point) is generated from a large area which can't show real shape of strap-like point set.

#### 2) *MIDDLE POINTS* OF AMBIGUOUS STRAP-LIKE POINT SETS

Some special strap-like point sets have ambiguous shapes information. Their shapes mainly are determined by the density of points. For example, the strap-like point set in figure 7(a) will have an exact shape as shown in figure 7(b) if the point density is big enough. Otherwise it will have a coarse shape as shown in figure 7(c).

In order to make the initial *middle points* more regular, an equal interval resample procedure is proposed. Given a set of ordered initial middle points $points = \{p_i | i = 1, 2, \ldots, N\}$ and a resample interval D, the resample procedure is described in algorithm 1.

**Algorithm 1** Resample *Middle Points*.

Input: initial *middle points points*, and resample interval D
Output: resample *middle points, i.e. newpoints*
1. *newpoints.push_back*(*points*0); // *points*0 is the first point in *points*
2. for each point $p_i$, $i = 0, 1, \ldots,$ *points.size*()
3. *dis*=Distance($p_i, p_j$), $j = i + 1, \ldots,$ *points.size*() // Euclidean distance between two points
4. while ($dis < D$)
5. $j = j + 1$;
6. $dis =$ Distance($p_i, p_j$);
7. *points*.delete($p_{j-1}$);
8. end while
9. $q =$ Interpolate($p_i, p_{j-1}, p_j, D$);// make $\|p_i - q\| = D$
10. *newpoints*.push_back($q$);
11. *points*.insert($q, i$);// $q$ will be the next $p_i$
12. end for

On the base of slicing procedure and the equal interval resample algorithm, the detailed regularization is given in algorithm 2.

Our regularization method needs further improvement in terms of several special cases. The first is that in some cases, the cross sections are so complex that the strap-like points can't be segmented automatically by DBSCAN algorithm. It will depend on the user-interaction to segment the complex cross sections into strap-like point sets perfectively. The second case is that, when the surface to be sliced is parallel to the slicing plane or has a very little intersect angle with the slicing plane, the corresponding projected point sets no longer should be viewed as strap-like point sets, otherwise the wrong

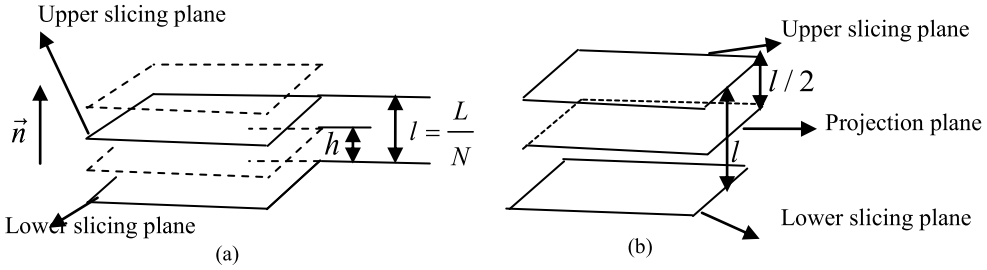



**FIGURE 4.** Slicing plane and projection plane. (a) Moving of slicing planes, (b) projection plane.

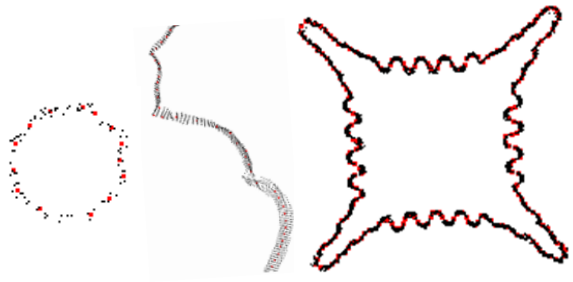


**FIGURE 5.** 2D *middle points* of different strap-like.


**FIGURE 6.** *Middle points* of 2D strap-like cross section point sets with non-uniform width.

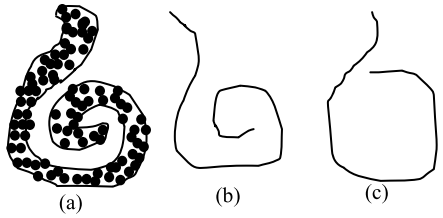



**FIGURE 7.** Ambiguous 2D strap-like cross section points set and its shape. (a) Points set, (b) exact shape, (c) coarse shape.

results will be produced. These mentioned surfaces could be distinguished from other surfaces by normal and they need a new slicing direction. The concrete research will be done in our next work.

## IV. EXPERIMENTS

The proposed method is implemented using C++, PCL (Point Clouds Library), and OpenGL. The face model tested in this paper is coming from [26]. The bunny model is downloaded from the Stanford University website (http://graphics.stanford.edu/data/3Dscanrep/).

**Algorithm 2** Total Regularization Algorithm.

Input: point clouds, $h$, $l$, $\vec{n}$, D
Output: regular point clouds RP
1. RP=$\phi$;
2.while points in point clouds are not sliced
3.$P_{cross}$= Getslice(point clouds, $h$, $l$, $\vec{n}$);// $P_{cross}$ is the cross section points of the point clouds
4.{$P_{\_lemda}$}=DBSCAN($P_{cross}$);
5.for each $P_{\_lemda}$
6. *middle points*=GGA($P_{\_lemda}$);
7. $RP_{\_lemda}$=Resample(*middle points*, D);
8. RP= RP $\cup$ $RP_{\_lemda}$;
9.end for
10.end while

The others are presented by Institute of Computer science of Xi'an University of Technology.

### A. EXPERIMENTS RESULTS

The proposed method can produce regular point clouds that preserve the same shape with the input point clouds in different level of details (LOD). It can be seen from the test results ( figure 8, figure 9 ) that the output point clouds generated by the proposed method are regular. The LOD is adjusted by slicing step size $h$ and resample interval D. If $h$ and D are small, more points are reserved, and more details are maintained and vice versa. The proposed method not only can produce the regular point clouds in which point density is lower than the input point clouds. When $h$ and D are set with values that are smaller than the scanning resolution, the method also can produce the regular point clouds whose point density is higher than input point clouds.

Figure 10(b), figure 11(a) and figure 12(a) show the meshing results of irregular raw point clouds with Delaunay based surface reconstruction method. Figure 10(c), (d), (e), figure 11(d), (e), (f) and figure 12(d), (e), (f) show the meshing results of the corresponding regular point clouds of different LOD. It can be seen that the irregular raw point cloud leads to locally wrong meshing results while the regular point clouds can produce right meshing results.

A bunny model with 10% Gaussian noise is tested in the experiment and the results are shown in figure 10. It can

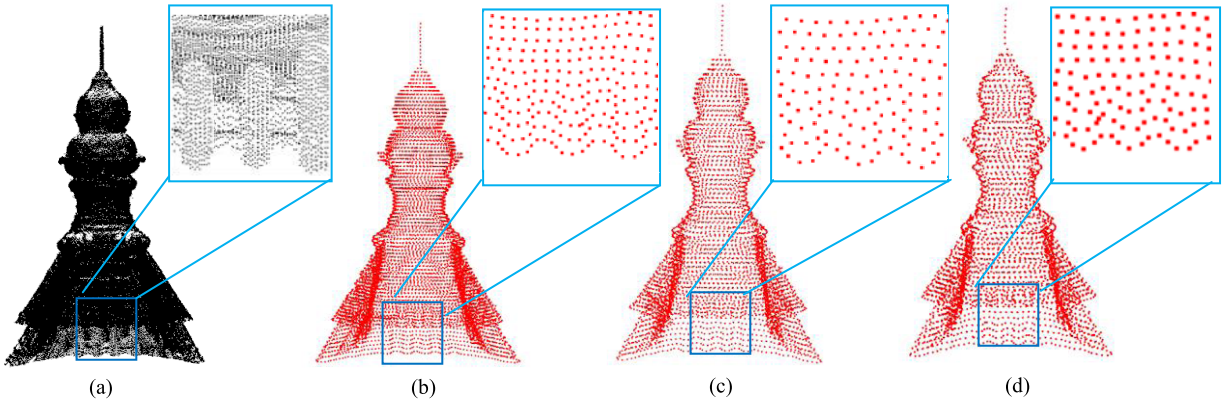



**FIGURE 8.** Results of tower model. (a) The original tower model (105227points); (b) N=70, h=0.6*l, D=h (7726points); (c)N=70, h=0.8*l, D=h (4612 points); (d) N=70, h=1.0*l, D=h (3153points).

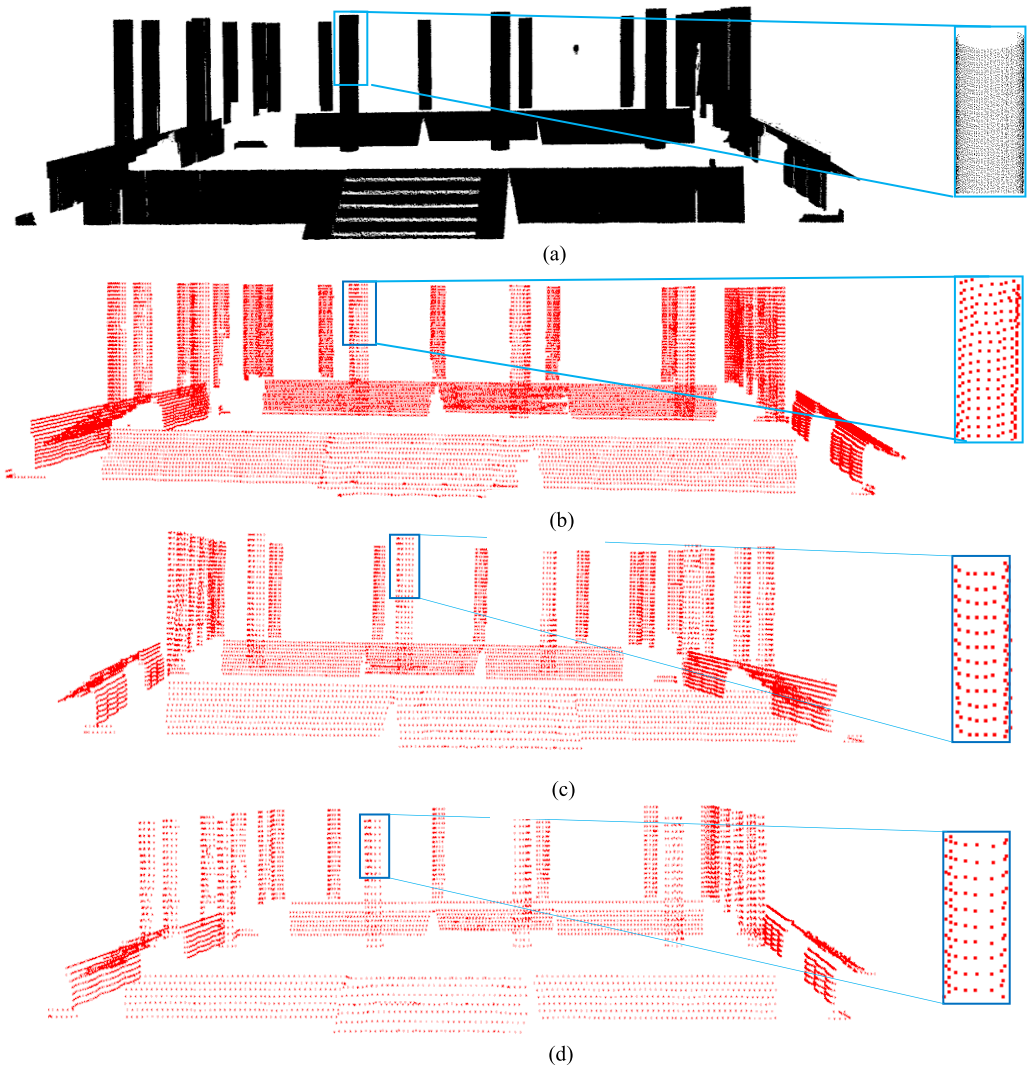



**FIGURE 9.** Results of scene model. (a) The original scene model (1301379 points); (b) N=30, h=0.6*l, D=h (19935 points); (c) N=30, h=0.8*l, D=h (11564points); (d) N=30, h=1.0*l, D=h (6782points).

be seen that the proposed method can restore the regular de-noised point clouds under the influence of noise. Meanwhile, the meshing results in figure 10 (c), (d) and (e) show geometric features of the de-noising point clouds are reserved. It can be seen that the sharp features in the ears of bunny are not preserved well. That's because the cross section points set generated by projection of sliced layer of the sharp parts in ears is the ambiguous

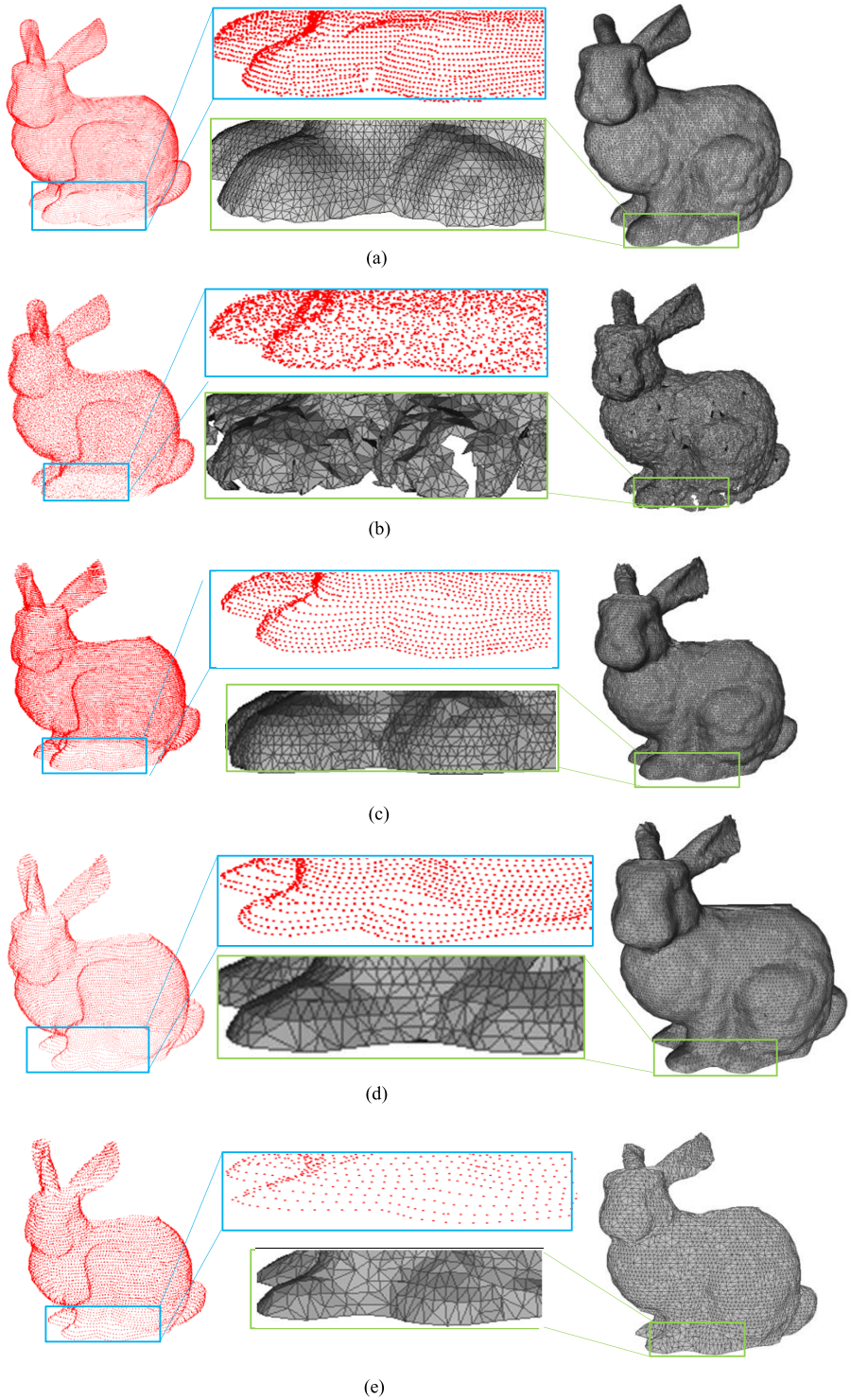



**FIGURE 10.** Results of bunny model. (a) The original bunny model (31418points), its details and its mesh model; (b) The bunny model with noise, its details and its mesh model; (c) Using our method with parameter (N=50, h=0.4*l, D =h), finally produce points(20016),and its mesh model, $\Delta_{avg} = 0.07929$, $\Delta_{max} = 0.8406$, $\sigma = 0.1782$; (d) Using our method with parameter (N=50, h=0.6*l, D =h), finally produce points(12282), and its mesh model, $\Delta_{avg} = 0.09599$, $\Delta_{max} = 0.8547$, $\sigma = 0.2392$; (e) Using our method with parameter (N=50, h=0.8*l, D =h), finally produce points(6992), and its mesh model, $\Delta_{avg} = 0.1138$, $\Delta_{max} = 0.8055$, $\sigma = 0.4406$.

strap-like point set. The density of points in the strap-like point set is not big enough, and the *middle points* are wrong.

We match the output point clouds and input point clouds with ICP [40] algorithm to prove the regular point clouds can preserve the shape of the input point clouds. The root

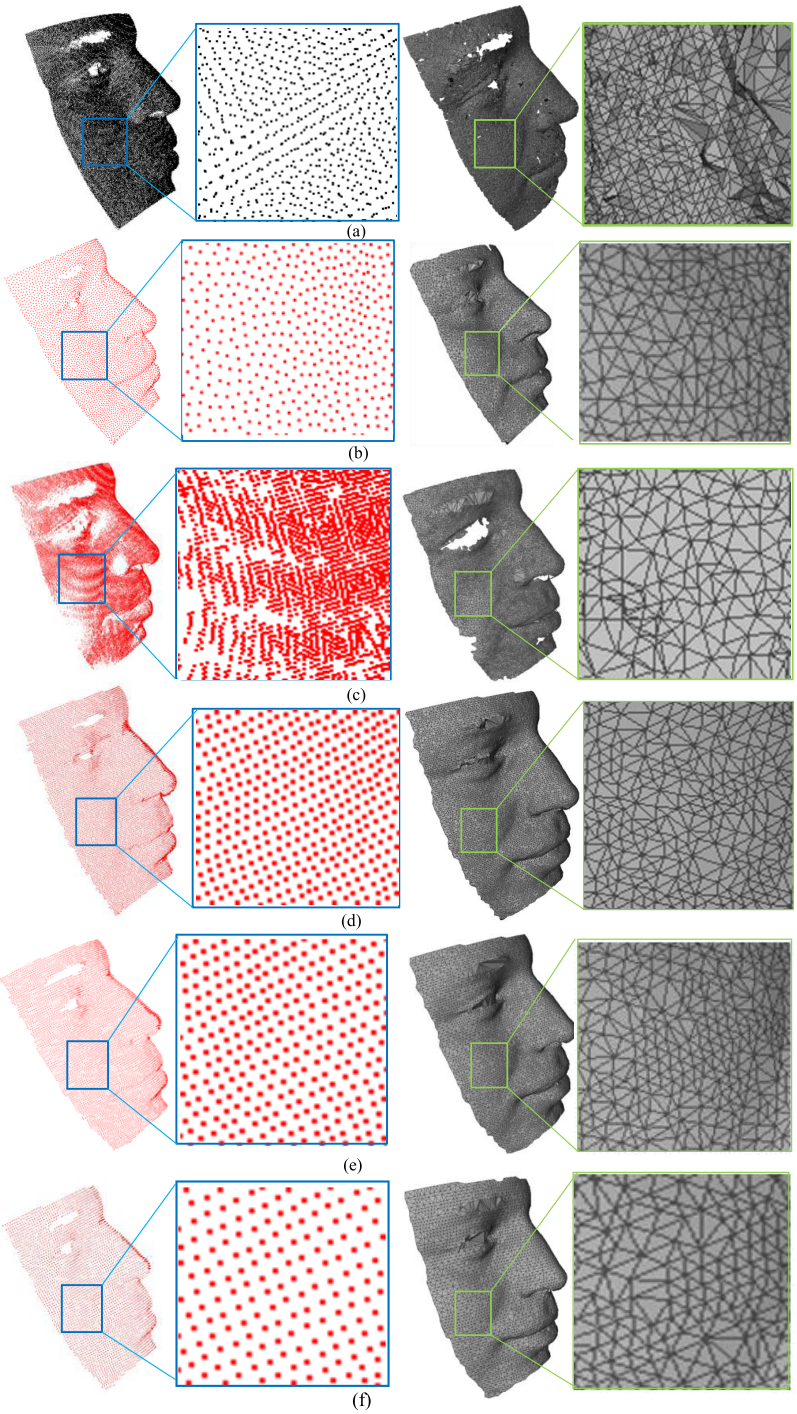



**FIGURE 11.** Results of face model. (a) The original face model with points (84398) and its mesh model; (b) Using WLOP method, partial details and its mesh model with points(4220), $\Delta_{avg} = 0.05225$, $\Delta_{max} = 0.4544$, $\sigma = 0.03788$; (c) Using AMLS method, partial details and its mesh model with points (84398), $\Delta_{avg} = 0.06919$, $\Delta_{max} = 0.8312$, $\sigma = 142.087$; (d) Using our method with parameter (N=60, h=0.6*l, D =h), finally produce points(8637), partial details and its mesh model, $\Delta_{avg} = 0.05224$, $\Delta_{max} = 0.4580$, $\sigma = 0.04841$; (e) Using our method with parameter (N=60, h=0.8*l, D =h), finally produce points(5390), and its mesh model, $\Delta_{avg} = 0.07847$, $\Delta_{max} = 0.4297$, $\sigma = 0.05736$; (f) Using our method with parameter (N=60, h=1.0*l, D =h), finally produce points (3464), and its mesh model, $\Delta_{avg} = 0.1052$, $\Delta_{max} = 0.5867$, $\sigma = 0.07727$.

mean square (RMS) registration errors of ICP algorithm are taken as match errors that are shown in table1. The normal of input point clouds and output point clouds produced by our method is extracted respectively. The method [41] is used to compute the normal. The normal is exploited to constitute geometric features sets for input point clouds and output point

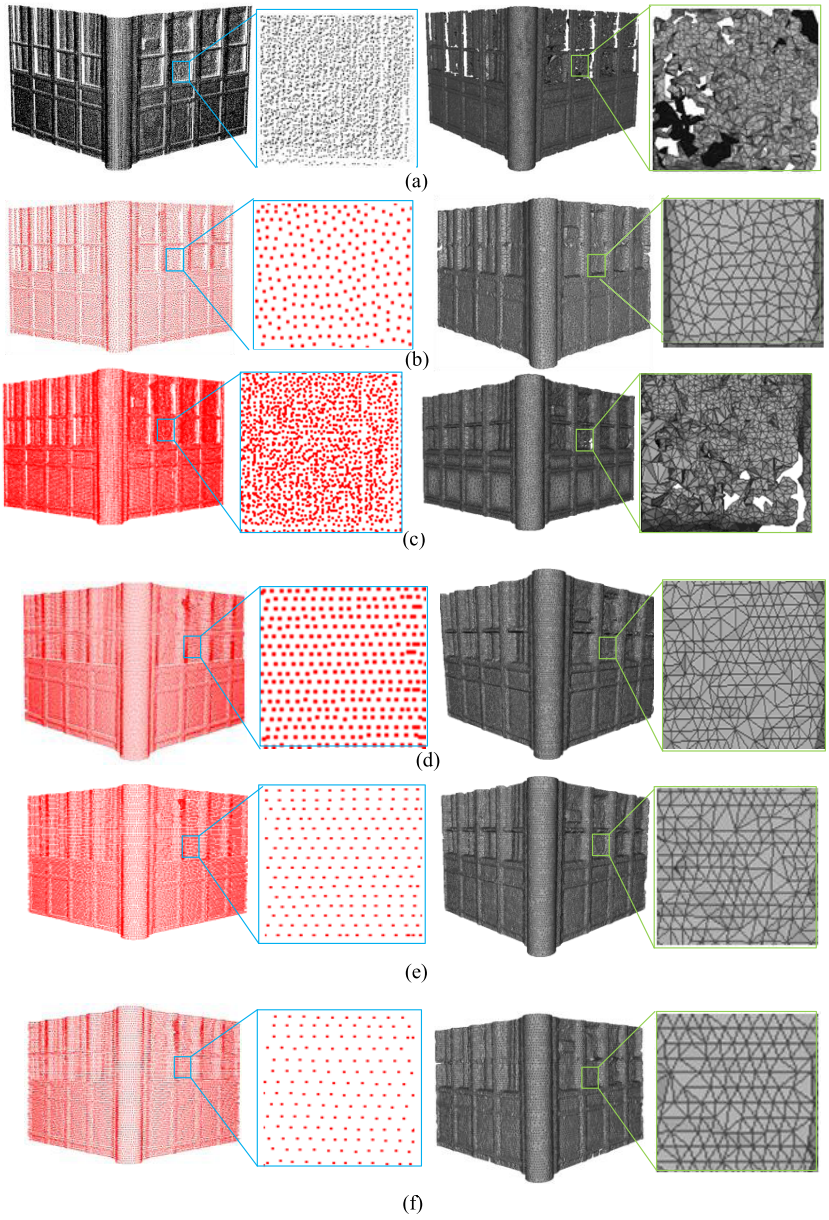



**FIGURE 12.** Results of door model. (a)The original door model (207228); (b) Using WLOP method, partial details and its mesh model with points (10362), $\Delta_{avg} = 1.0064$, $\Delta_{max} = 7.8263$, $\sigma = 1.4256$; (c)Using AMLS method, partial details and its mesh model with points (207228), $\Delta_{avg} = 1.1052$, $\Delta_{max} = 8.0023$, $\sigma = 12.3892$; (d) Using our method with parameter (N=80, h=0.6*l, D =h), finally produce points (27305), partial details and its mesh model, $\Delta_{avg} = 0.3270$, $\Delta_{max} = 7.7817$, $\sigma = 1.4647$; (e) Using our method with parameter (N=80, h=0.8*l, D =h), finally produce points (17839), and its mesh model, $\Delta_{avg} = 1.2077$, $\Delta_{max} = 8.9895$, $\sigma = 1.6181$; (f) Using our method with parameter (N=60, h=1.0*l, D =h), finally produce points (12895), and its mesh model, $\Delta_{avg} = 1.6872$, $\Delta_{max} = 9.0023$, $\sigma = 1.8346$.

clouds respectively. The distance between input point clouds features set and output point clouds features set (feature distance for short) is also shown in table 1. Given a feature set $F_1$, distance $d_f(F_1, F_2)$ between another feature set $F_2$ and feature set $F_1$ is computed by $d_f(F_1, F_2) = (\text{sqrt}(\sum_{i=1}^{N} d_p(p_i, F_2)^2)/N)$, where $F_1 = \{p_i\}\{i = 1, 2, \ldots, N\}$, $d_p(p_i, F_2) = \min_{q_j \in F_2}(d_e(p_i, q_j))$, $d_e(p_i, q_j)$ is the Euclidean distance between point $p_i$ and $q_j$.

From table 1, it can be seen that the match errors and feature distances are small, and the output point clouds by the proposed method preserve the shape of input point clouds.

In order to evaluate the accuracy of the output point clouds, the geometric error between the original and output point clouds has also been measured. In this paper, the maximum error and the average error between the original point set $S$ and the smaller point set $S'$ are measured by

**TABLE 1. The match errors and feature distance between input point clouds and output point clouds.**

| Model produced by different parameters | Match error | Feature distance |
|---|---|---|
| Bunny (N=50, h=0.4*l, D =h) | 0.08238 | 0.0159 |
| Bunny (N=50, h=0.6*l, D =h) | 0.12 | 0.0135 |
| Bunny (N=50, h=0.8*l, D =h) | 0.06548 | 0.0213 |
| Face (N=60, h=0.6*l, D =h) | 0.2782 | 0.0050 |
| Face (N=60, h=0.8*l, D =h) | 0.2784 | 0.0039 |
| Face (N=60, h=1.0*l, D =h) | 0.2632 | 0.0039 |
| Tower (N=70, h=0.6*l, D=h) | 0.0128 | 0.0042 |
| Tower (N=70, h=0.8*l, D=h) | 0.0101 | 0.0042 |
| Tower (N=70, h=1.0*l, D=h) | 0.0100 | 0.0039 |
| Door (N=80, h=0.6*l, D =h) | 0.0056 | 0.0046 |
| Door (N=80, h=0.8*l, D =h) | 0.0062 | 0.0043 |
| Door (N=80, h=1.0*l, D =h) | 0.0062 | 0.0036 |
| Scene (N=30, h=0.6*l, D=h) | 0.0056 | 0.0086 |
| Scene (N=30, h=0.8*l, D=h) | 0.0055 | 0.0089 |
| Scene (N=30, h=1.0*l, D=h) | 0.0052 | 0.0088 |

$\Delta_{\max}(S, S') = \max_{q \in S} \left| d_s(q, S') \right|$ and $\Delta_{avg}(S, S') = 1/\|S\| * \sum_{q \in S} \left| d_s(q, S') \right|$.

For each point $q \in S$, the geometric error $d_s(q, S')$ can then be defined as the Euclidean distance between the point $q$and its projection point $\bar{q}$ on the surface $S'$ that is consisted of the smaller and de-noised points. The projection point $\bar{q}$ is calculated according to the procedure presented in [42]. The results show that the geometric error between the input point clouds and output regular point clouds is very small.

### *B. COMPARISON WITH OTHER CONSOLIDATION METHODS*

The proposed method is compared with WLOP [26] and AMLS (an adaptive MLS method) [22]. The executable package of the WLOP [26] and AMLS [22] are downloaded from the author's web page respectively. Two models are choose to compare them. One is the face model that has few points and many features. The other is a door model that has less features and huge points. It can be seen from the results in figure 11 and figure 12 that the proposed method is more robust in comparison with AMLS. Details of the output point clouds produced by the proposed method show good regularity. The output point clouds produced by the AMLS consolidation method are less regular, so there exist some wrong surface meshing results, as shown in figure 11(c) and figure 12(c). Finally, the proposed method produces regular point clouds in LOD results with different precision. Usually, when parameters h is less than or equals 0.6*l, high LOD output point clouds generated by the proposed method can obtain higher precision than other consolidation methods. And the higher levels of detail, the higher precision achieved.

It should be noted that the proposed method can work in a parallel way easily while the WLOP and AMLS cannot. When point clouds are huge, the method can reduce executing times by parallel running. We adopt the scene model and the door model that have huge number of points to compare the executing times of the proposed method and the two consolidation methods. Executing time of our algorithm depends on the slicing numbers N and step size h. The proposed method with bigger N and smaller h is compared with other consolidation methods. Experiments are carried out on a PC with 8 core 16 threads Intel(R) Core, CPU 3.30GHz, 4G memory. The results are shown in table 2. It can be seen that our algorithm executed in a parallel manner can save the executing time of huge point clouds and consumes a shorter time than WLOP and AMLS.

**TABLE 2. The executing times.**

| Models | Points | Methods | Points | Time(s) |
|---|---|---|---|---|
| Door | 207228 | WLOP | 10362 | 56s |
| Door | 207228 | AMLS | 207228 | 1210s |
| Door | 207228 | Our method(N=80, h=0.6*l) | 27305 | 434s |
| Door | 207228 | Our method(N=80, h=0.6*l) in a parallel way | 27305 | 31s |
| Scene | 1301379 | WLOP | 65069 | 331s |
| Scene | 1301379 | AMLS | 1301079 | 5840s |
| Scene | 1301379 | Our method(N=30, h=0.6*l) | 19935 | 615s |
| Scene | 1301379 | Our method(N=30, h=0.6*l) in a parallel way | 19935 | 41s |

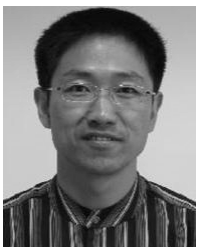
## V. CONCLUSION

In this paper, a novel regularization method of raw point clouds is proposed. The output point clouds of the method preserve shape of input point clouds well. The method is mainly impacted by the two factors: user-defined slicing step size h and uniformly resample size D. By adjusting the two parameters, different level of details and amount can be obtained. Especially, our algorithm can be running in parallel way to improve the efficiency when huge point clouds are handled. Comprehensive examples and a comparative study illustrate the good performance of the method. Our method can be applied to surface reconstruction, shape analysis, shape extraction, etc.

## REFERENCES

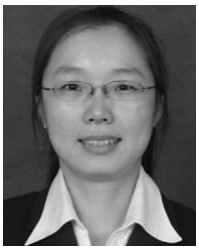
[1] N. Amenta, M. Bern, and M. Kamvysselis, "A new Voronoi-based surface reconstruction algorithm," in *Proc. Conf. Comput. Graph. Interact. Techn.*, New York, NY, USA, 1998, pp. 415–421.

[2] Y. Ohtake, A. Belyaev, and H.-P. Seidel, "An integrating approach to meshing scattered point data," in *Proc. ACM Symp. Solid Phys. Modeling*, Cambridge, MA, USA, 2005, pp. 61–69.
[3] M. Kazhdan, M. Bolitho, and H. Hoppe, "Poisson surface reconstruction," in *Proc. Eurograph. Symp. Geometry Process.*, Graz, Austria, 2015, pp. 61–70.
[4] M. Bartsch, T. Weiland, and M. Witting, "Generation of 3D isosurfaces by means of the marching cube algorithm," *IEEE Trans. Magn.*, vol. 32, no. 3, pp. 1469–1472, May 1996.
[5] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 4, pp. 509–522, Apr. 2002.
[6] T. K. Dey, K. Mehlhorn, and E. A. Ramos, "Curve reconstruction: Connecting dots with good reason," *Comput. Geometry*, vol. 15, no. 4, pp. 229–244, 1970.
[7] S. Ohrhallinger and S. Mudur, "An efficient algorithm for determining an aesthetic shape connecting unorganized 2D points," *Comput. Graph. Forum*, vol. 32, no. 8, pp. 72–88, 2013.
[8] S. Ohrhallinger, S. Mudur, and M. Wimmer, "Minimizing edge length to connect sparsely sampled unstructured point sets," *Comput. Graph.*, vol. 37, no. 6, pp. 645–658, 2013.
[9] J. D. Boissonnat and A. Ghosh, "Manifold reconstruction using tangential delaunay complexes," in *Proc. 26th Symp. Comput. Geometry*, Snowbird, UT, USA, 2010, pp. 324–333.
[10] C. C. L. Wang and D. Manocha, "Efficient boundary extraction of BSP solids based on clipping operations," *IEEE Trans. Vis. Comput. Graphics*, vol. 19, no. 1, pp. 16–29, Jan. 2013.
[11] M. Wang, J.-Q. Feng, and W. Chen, "Efficient boundary surface reconstruction from heterogeneous volumetric data via tri-prism decomposition," *Comput. Graph.*, vol. 38, no. 1, pp. 212–221, 2014.
[12] Q. Jia, X. Fan, Y. Liu, H. Li, Z. Luo, and H. Guo, "Hierarchical projective invariant contexts for shape recognition," *Pattern Recognit.*, vol. 52, pp. 358–374, Apr. 2015.
[13] P. Jenke, M. Wand, M. Bokeloh, A. Schilling, and W. Straßer, "Bayesian point cloud reconstruction," *Comput. Graph. Forum*, vol. 25, no. 3, pp. 379–388, 2006.
[14] R.-F. Wang, W.-Z. Chen, S.-Y. Zhang, Y. Zhang, and X.-Z. Ye, "Similarity-based denoising of point-sampled surfaces," *J. Zhejiang Univ.-SCI. A*, vol. 9, no. 6, pp. 807–815, 2008.
[15] Y. Sun, S. Schaefer, and W. Wang, "Denoising point sets via $L_0$ minimization," *Comput. Aided Geometric Des.*, vols. 35–36, pp. 2–15, May 2015.
[16] C. Wu, S. Agarwal, and B. Curless, "Schematic surface reconstruction," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Providence, RI, USA, Jun. 2012, pp. 1498–1505.
[17] K. Yin, H. Huang, H. Zhang, M. Gong, D. Cohen-Or, and B. Chen, "Morfit: Interactive surface reconstruction from incomplete point clouds with curve-driven topology and geometry control," *ACM Trans. Graph.*, vol. 33, no. 6, pp. 1–12, 2014.
[18] M. Zou, M. Holloway, N. Carr, and T. Ju, "Topology-constrained surface reconstruction from cross-sections," *ACM Trans. Graph.*, vol. 34, no. 4, pp. 1–10, 2015.
[19] I. Kyriazis, I. Fudos, and L. Palios, "Extracting CAD features from point cloud cross-sections," Václav Skala—UNION Agency, Plzeň Region, Czech Republic, Tech. Rep. 1, 2009.
[20] M. Alexa, J. Behr, D. Cohen-Or, S. Fleishman, D. Levin, and C. T. Silva, "Computing and rendering point set surfaces," *IEEE Trans. Vis. Comput. Graphics*, vol. 9, no. 1, pp. 3–15, Jan./Mar. 2003.
[21] N. Amenta and Y. J. Kil, "Defining point-set surfaces," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 264–270, 2004.
[22] T. K. Dey and J. Sun, "An adaptive MLS surface for reconstruction with guarantees," in *Proc. Eurograph. Symp. Geometry Process.*, Vienna, Austria, 2005, pp. 43–52.
[23] G. Guennebaud and M. Gross, "Algebraic point set surfaces," *ACM Trans. Graph.*, vol. 26, no. 3, 2007, Art. no. 23.
[24] A. C. Öztireli, G. Guennebaud, and M. Gross, "Feature preserving point set surfaces based on non-linear kernel regression," *Comput. Graph. Forum*, vol. 28, no. 2, pp. 493–501, 2009.
[25] Y. Lipman, D. Cohen-Or, D. Levin, and H. Tal-Ezer, "Parameterization-free projection for geometry reconstruction," *ACM Trans. Graph.*, vol. 26, no. 3, pp. 1–6, 2007.
[26] H. Huang, D. Li, H. Zhang, U. Ascher, and D. Cohen-Or, "Consolidation of unorganized point clouds for surface reconstruction," *ACM Trans. Graph.*, vol. 28, no. 5, pp. 89–97, 2009.
[27] S. Liu, K.-C. Chan, and C. C. L. Wang, "Iterative consolidation of unorganized point clouds," *IEEE Comput. Graph. Appl.*, vol. 32, no. 3, pp. 70–83, May 2012.
[28] R. Preiner, O. Mattausch, M. Arikan, R. Pajarola, and M. Wimmer, "Continuous projection for fast $L_1$, reconstruction," *ACM Trans. Graph.*, vol. 33, no. 4, pp. 70–79, 2014.
[29] S. Wu, H. Huang, and M. Gong, "Deep points consolidation," *ACM Trans. Graph.*, vol. 34, no. 6, pp. 1–13, 2015.
[30] J. Giesen and U. Wagner, "Shape dimension and intrinsic metric from samples of manifolds," *Discrete Comput. Geometry*, vol. 32, no. 2, pp. 245–267, 2004.
[31] F. Mémoli and G. Sapiro, "Comparing point clouds," in *Proc. Eurograph./ACM SIGGRAPH Symp. Geometry Process.*, New York, NY, USA, 2004, pp. 32–40.
[32] Y. F. Wu, Y. S. Wong, and H. T. Loh, "Modelling cloud data using an adaptive slicing approach," *Comput.-Aided Des.*, vol. 36, no. 3, pp. 231–240, 2004.
[33] F. Javidradn and A. R. Pourmoayed, "Contour curve reconstruction from cloud data for rapid prototyping," *Robot. Comput.-Integr. Manuf.*, vol. 27, no. 2, pp. 397–404, 2011.
[34] L. Zeng, L. M.-L. Lai, D. Qi, Y.-H. Lai, and M. M.-F. Yuen, "Efficient slicing procedure based on adaptive layer depth normal image," *Comput.-Aided Des.*, vol. 43, no. 12, pp. 1577–1586, 2011.
[35] G. Q. Jin, W. D. Li, and L. Gao, "An adaptive process planning approach of rapid prototyping and manufacturing," *Robot. Comput.-Integr. Manuf.*, vol. 29, no. 1, pp. 23–38, 2013.
[36] Q. Zhang, L. T. Yang, Z. Chen, and P. Li, "PPHOPCM: Privacy-preserving high-order possibilistic c-means algorithm for big data clustering with cloud computing," *IEEE Trans. Big Data*, to be published.
[37] M. Li, Z. Lu, and L. Huang, "An approach to 3D shape blending using point cloud slicing," in *Proc. IEEE 10th Int. Conf. Comput.-Aided Ind. Design Conceptual Design*, Wenzhou, China, Nov. 2009, pp. 919–922.
[38] K. K. Sareen, G. K. Knopf, and R. Canas, "Surface reconstruction from sliced point cloud data for designing facial prosthesis," in *Proc. IEEE Toronto Int. Conf. Sci. Technol. Hum. (TIC-STH)*, Toronto, ON, Canada, 2009, pp. 6–11.
[39] H. W. Lin, W. Chen, and G. J. Wang, "Curve reconstruction based on an interval B-spline curve," *Vis. Comput.*, vol. 21, no. 6, pp. 418–427, 2005.
[40] P. J. Besl and N. D. Mckay, "Method for registration of 3-D shapes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 2, pp. 239–256, Feb. 1992.
[41] M. Pauly, M. Gross, and L. P. Kobbelt, "Efficient simplification of point-sampled surfaces," in *Proc. IEEE Vis.*, vol. 1. Oct. 2002, pp. 163–170.
[42] B. Q. Shi, J. Liang, Q. Liu, and Z. Z. Xiao, "Precision inspection of point cloud & CAD model based on constraint search sphere," *Comput. Integr. Manuf. Syst.*, vol. 16, no. 5, pp. 929–934, 2010.
[43] Q. Zhang, L. T. Yang, and Z. Chen, "A survey on deep learning for big data," *Inf. Fusion*, vol. 42, pp. 146–157, Jul. 2018.



**YINGHUI WANG** received the Ph.D. degree from North-West University, Xi'an, China, in 2002. From 2003 to 2005, he was a Post-Doctoral Fellow with Peking University, Beijing, China. He is currently a Professor with the Institute of Computer Science and Engineering, Xi'an University of Technology, China. He is also with the Shaanxi Provincial Key Laboratory of Network Computing and Security Technology, Xi'an. His research interests include image analysis, point clouds processing, and pattern recognition.

**LIJUAN WANG** received the B.S. and M.S. degrees from the Department of Electronic Information Engineering, Xi'an Technological University, Xi'an, China, in 2001 and 2004, respectively, where she is currently pursuing the Ph.D. degree with the Institute of Computer Science and Engineering, Xi'an University of Technology, Xi'an. She is currently with the Institute of Electronic Information Engineering, Xi'an Technological University. Her research interests include pattern recognition and point clouds processing.



**WEN HAO** received the M.S. degree from the Department of Computer Science, Shann'xi Normal University, Xi'an, China, in 2011, and the Ph.D. degree from the Xi'an University of Technology, Xi'an, in 2015. She is currently with the Institute of Computer Science and Engineering, Xi'an University of Technology. Her research interests include pattern recognition and point cloud processing.

**XIAOJUAN NING** received the Ph.D. degree in 2011. She is currently an Associate Professor with the Xi'an University of Technology. She has cooperated with the Lab of LIAMA and the National Laboratory of Pattern Recognition, Institute of Automation, CAS. Her current research interests include scene modeling and shape analysis.

**ZHENGHAO SHI** received the Ph.D. degree in 2005. He is an Associate Professor with the Xi'an University of Technology. His research interests include image processing and computer vision.

**MINGHUA ZHAO** received the Ph.D. degree in 2006. She is currently an Associate Professor with the Xi'an University of Technology. Her research interests include pattern recognition and image processing.