An Automatic Hole Filling Method of Point Cloud for 3D Scanning

Yuta MURAKI
Osaka Institute of Technology
Osaka, Japan
yuta.muraki@oit.ac.jp

Koji NISHIO
Osaka Institute of Technology
Osaka, Japan
koji.a.nishio@oit.ac.jp

Takayuki KANAYA
Hiroshima International University
Hiroshima, Japan
t-kanaya@hw.hirokoku-u.ac.jp

Ken-ichi KOBORI
Osaka Institute of Technology
Osaka, Japan
kenichi.kobori@oit.ac.jp

ABSTRACT

In recent years, due to the development of three-dimensional scanning technology, the opportunities for real objects to be three-dimensionally measured, taken into the PC as point cloud data, and used for various contents are increasing. However, the point cloud data obtained by three-dimensional scanning has many problems such as data loss due to occlusion or the material of the object to be measured, and occurrence of noise. Therefore, it is necessary to edit the point cloud data obtained by scanning. Particularly, since the point cloud data obtained by scanning contains many data missing, it takes much time to fill holes. Therefore, we propose a method to automatically filling hole obtained by three-dimensional scanning. In our method, a surface is generated from a point in the vicinity of a hole, and a hole region is filled by generating a point sequence on the surface. This method is suitable for processing to fill a large number of holes because point sequence interpolation can be performed automatically for hole regions without requiring user input.

Keywords

Hole filling, 3D Scanning, Surface Approximation, Point Cloud

1. INTRODUCTION

In recent years, due to the development of three dimensional scanning technology, the opportunities for real objects to be three-dimensionally measured, taken into the PC as point cloud data, and used for various contents are increasing. In three-dimensional scanning, data of the entire surroundings of the scanning object is acquired by generally measuring the scanning object from a plurality of directions and aligning the obtained plural point group data. In Figure 1, the point cloud data obtained from a plurality of directions are displayed in a color-coded manner, and one model data is constructed by integrating them.

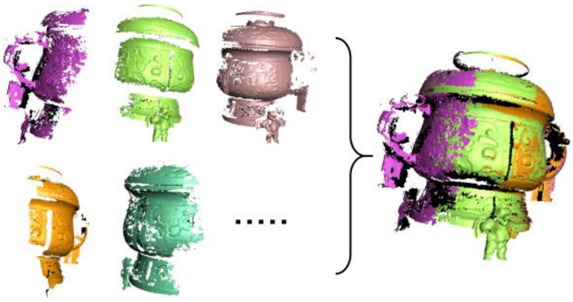
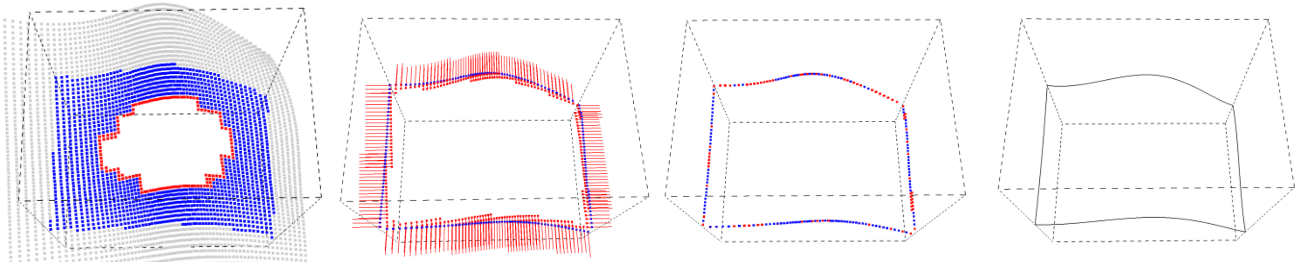


Figure 1. The point cloud data obtained from three-dimensional measurement.

However, the point cloud data obtained by three-dimensional scanning has many problems such as data loss due to occlusion and the material of the object to be measured, and occurrence of noise. Therefore, it is necessary to edit point cloud data obtained by scanning. Especially, since the point cloud data obtained by scanning contains many data missing, it takes a huge amount of time to fill holes.



(a) Selection of point group **(b) Projection to OBB plane** **(c) Extracting sample points** **(d) Generate closed region**

Figure 2. Procedure for generating a closed area.

Therefore, we propose a method to automatically fill hole of point cloud data obtained by three-dimensional scanning. In this method, the bounding box (OBB) is generated from the neighboring point of the missing region and the region is expanded. After that, a point group included in the expanded bounding box is extracted and approximated to generate a B-spline surface. Then, a point sequence is generated on the generated B-spline surface and interpolation is performed by deleting points other than the portion corresponding to the missing. This method is suitable for processing to fill a large number of holes because point group interpolation can be performed automatically for hole regions without requiring user input.

2. RELATED WORK

The hole filling method of point cloud data is roughly divided into the following three types.

1. Method of interpolating using a database[1,2].
2. Method of interpolating using the shape information of the entire model[3,4].
3. Method of interpolating using the surrounding shape[5-11].

The method [1, 2] of interpolating using a database is a method of selecting a shape similar to the target object from the database and integrating it with the target object. In these methods, there is a problem that it is costly because a shape similar to the target object needs to be modeled and stored in advance.

As a method of using the shape information of the entire model, a method of copying an area having the maximum degree of similarity of shape and texture has been proposed [3, 4]. However, in method [3], hole is filled with one copy. Therefore, when there is no similar shape of the same size as the hole in the model, it is an interpolation result which is liable to cause discomfort.

In the method [4], interpolation of the hole is performed by copying a shape with the minimum energy function by using similarity of shape and texture. In this method, there are problems that the parameters are determined manually and that the interpolation result depends greatly on the parameters. In addition, there is a problem that these methods [3, 4] commonly have high calculation cost.

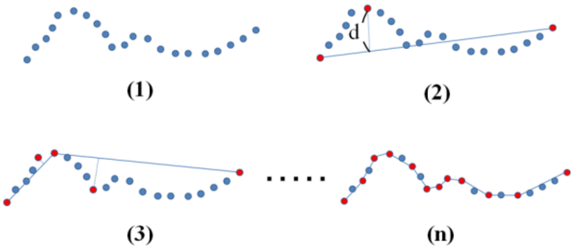
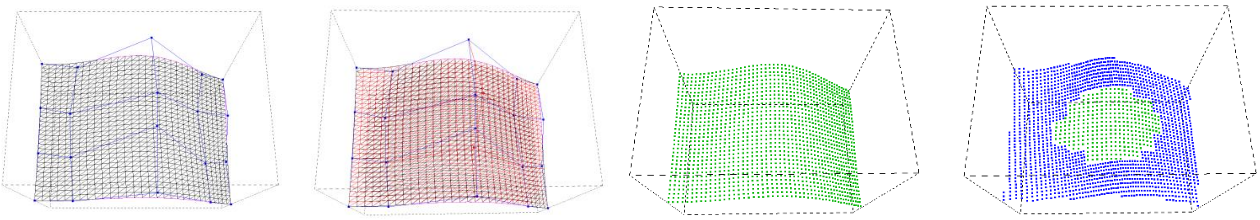


Figure 3. Piecewise linear approximation.

The method [5-11] that uses surrounding shapes for interpolation is a method of performing smooth interpolation by using differential equations and surface fitting. In these methods, good results can be obtained for small hole regions. In particular, the method [9] is also applicable to complex hole regions with "islands". However, when large hole regions are filled, there are cases where shapes greatly different in the properties of three-dimensional structures are generated. If the area of the hole is larger, these method produce plane based results.

3. PROPOSED METHOD

We propose a hole filling method based on surface approximation. In our method, by approximating neighboring points of a hole region, a surface covering a hole region is generated and an interpolation point is generated. In the proposed method, the control point is obtained so that the error between the surface to be generated and the point representing the outline of the hole is small, so it is possible to calculate an interpolation point with relatively high accuracy. In addition, since the continuity with the point group around the hole is taken into consideration, the interpolation points are small in gap with input point cloud.



(a) Calculation of internal control point **(b) Refine the surface control point** **(c) On-surface point generation** **(d) Interpolation point generation**

Figure 4. Curved surface and interpolation point generation

3.1. Selection of point group

In this section, a method of extracting a point group used for surface approximation will be described with using a part of the input point cloud shown in Figure 2 (a).

First, as shown by the red dots in the Figure 2 (a), points near the missing region representing the outline of the hole are extracted from the input point cloud by using based on Gerhard's method [12]. Then, a bounding box (OBB) of the extracted points is generated. Then using the vector obtained by PCA (principal component analysis), the area is expanded by expanding the generated OBB to the outside. In this method, as shown by blue dots in the Figure 2 (a), a point group included in the expanded OBB is extracted from the input point cloud as a point group to be used for fitting a surface.

3.2. Surface fitting

In this section, we describe a method to automatically generate a surface by using point group extracted in Section 3.1.

3.2.1. Generation of boundary curve

We describe the generation of a boundary curve of a fitted B-spline surface. First, as shown by the red dots in Figure 2 (b), neighbor points of each plane constituting the expanded OBB are extracted from the point group obtained in Section 3.1. Next, as shown by the red line in the Figure 2 (b), the extracted neighboring point of the OBB plane is projected onto the OBB plane to generate a new point sequence indicated in blue. After that, a sequence of points is extracted based on piecewise linear approximation from a number of blue points on the OBB plane. As shown in Figure 3, piecewise linear approximation is a method of approximating a point sequence with line segments and repeating recursive division until the distance d of the point farthest from the line segment is within the threshold ε. As a result, characteristic points can be extracted. The extraction result is indicated by red dots in Figure 2 (c). The point sequence obtained as described above is approximated by a B-spline curve. The four generated B-spline curves are independent each other, and the end points are not connected. Therefore, four neighboring B-spline curves are connected by extracting the neighboring points of the line segment representing the four corners of the OBB one by one and adding it to the control point of the B-spline curve. Then, as shown in Figure 2 (d), a closed quadrilateral region is generated.

3.2.2. Calculate internal control points of surface

As shown in Figure 4 (a), using the B-spline curve generated in Section 3.2.1 and the point group obtained in Section 3.1, the internal control points of the B-spline surface are calculated by the least squares method. The generated control points of B-spline surface is indicated by blue in Figure 4 (a). The generated polygons of B-spline surface is indicated by black in Figure 4 (a).

3.2.3. Refinement of internal control points

Since the curved surface generated by this method is an approximation of a point group, the center part of the surface region where the hole exists generally has low accuracy. Therefore, the accuracy of the interpolation point is improved by recalculating the internal control points so that the error of the hole neighborhood point becomes small. Polygons of B-spline surface and the surface control points after the refinement processing are indicated by red in Figure 4 (b).

3.3. Generate interpolation points

As shown in Figure 4 (c), a point sequence is generated on the generated B-spline surface. In order to eliminate discomfort to the input point cloud, the interval of the generated point sequence is calculated by the ratio of the length in the *u* and *v* direction of the generated quadrilateral region. After that, as shown by green dots in Figure 4 (d), distance from the generated surface points are extracted and set as interpolation points. This method is not influenced by the outline shape of the missing region.

Table 1. Accuracy evaluation

Data Set	#Original points	#Vicinity points	#Fill points	#New points	#control points	error (%)	time (sec.)
Data A	4,977	522	168	5,145	16	0.482436	0.19
Data B	4,770	600	155	4,925	16	0.344566	0.29
Data C	4,370	746	154	4,524	16	0.335054	0.24
Data D	9,419	647	190	9,609	16	0.285562	0.24
Data E	2,034	795	219	2,253	20	0.813469	0.27
Data F	4,375	1,636	366	4,741	20	0.302318	0.55
Data G	8,854	1,901	603	9,457	16	0.496557	0.69
Data H	8,349	1,620	714	9,063	16	0.532111	0.58
Data I	22,093	5,607	1,888	23,981	20	0.524364	3.27

In other words, it can also be applied to complex holes such as including "island" [9] where points exist inside the hole.

4. EXPERIMENTAL RESULTS

In order to verify the effectiveness of this method, accuracy and processing time were measured by applying this method by generating holes for randomly generated surface shapes. After that, we applied this method to actual scan data and verified the usefulness of the method. For the experiment, we used a desktop PC with *Core i7 (3.4 GHz)* and *8GB.*

4.1. Application to experimental data

Figure 5 shows the results of applying our method to nine point cloud data including hole. Gray points are point cloud with hole, and green points are interpolation points. Also, the generated B-spline surface is indicated by black polygons.

4.2. Accuracy evaluation

In this section, shape evaluation of the generated interpolation points will be described. In order to verify the accuracy, the distance between the generated interpolation points and the true value (surface) was measured. Table 1 shows the results of shape evaluation. Vicinity points in Table 1 are the points used for surface generation. Fill points are interpolation points generated by our method. Error indicates the accuracy of the interpolation points and is the value obtained the average of the distance between the interpolation points generated by our method and the true value (surface), and normalizing by the bounding box size. As shown in Table 1, it was confirmed that the error of the interpolation points is less than 1%, and approximation can be performed with good precision. In addition, the processing time is less than 1 second, and practicality is also proved.

4.3. Application to actual scanning data

Figure 6 shows the results of applying our method to scanning data obtained from a 3D scanner. Figures 6 (a), (c) and (e) are input data including hole, and (b), (d) and (f) are the result of applying the method. We can see that interpolation that maintains the shape features is possible.

5. CONCLUSIONS

In this paper, we proposed a hole filling method for point cloud based on surface approximation. In this method, a point group to be used for approximation is extracted by enlarging a region from a hole neighbor point, and it is possible to automatically generate an interpolation point. Moreover, by using refined outline information and performing refinement processing so that the accuracy of the interpolation point is high, high precision hole filling was made possible. We applied our method to the hole region including concavities and convexes and confirmed that high precision and high speed interpolation is possible. In addition, applying our method to actual scanning data, we confirmed that it is possible to fill hole without feeling uncomfortable.

It is difficult to apply our method to the part where the point near the hole becomes irregular like the base of the handle. In addition, if the area where the points are missing is none smooth then our method may not produce satisfactory results. Therefore, adaptation to more shapes can be cited as future works. Further improvement of the accuracy of the interpolation point can be cited as future works.

6. ACKNOWLEDGMENTS

This work was supported by JSPS KAKENHI Grant Number JP 17K13571.

7. REFERENCES

[1] V. Kraevoy and A. Sheffer, "Template-based mesh completion", Proc. Eurographics Symp. On Geometry Processing, pp.13-22, 2005.

[2] M.Pauly, "Example-based 3D scan completion", Proc. Eurographics Symp. On Geometry Processing, pp.23-32, 2005.

[3] S. Park, X. Guo, H. Shin, H. Qin, "Shape and Appearance Repair for Incomplete Point Surfaces", Proc. ICCV, Vol. 2, pp.1260-1267, 2005.

[4] N. Kawai, A. Zakhor, T. Sato, N. Yokoya, "Surface Completion of Shape and Texture Based on Energy Minimization", Proc. ICIP, pp.913-916, 2011.

[5] J. Verdera, V. Caselles, M. Bertalmio and G. Sapiro, "Inpainting surface holes", Proc. Int. Conf. on Image Processing, Vol.2, pp.903-906, 2003.

[6] P. Liepa, "Filling Holes in Meshes. Symposium on Geometry Processing", pp.200-205, 2003.

[7] J. Wang, M. M. Oliveira, "Filling holes on locally smooth surfaces reconstructed from point clouds", Image and Vision Computing, Vol.25, Issue.1, pp.103-113, 2007.

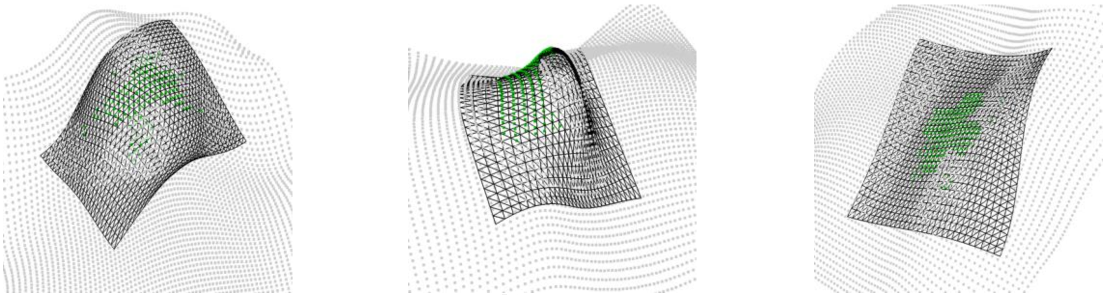
[8] E. Altantsetseg, Y. Muraki, F. Chiba and K. Konno, "3D Surface Reconstruction of Stone Tools by Using Four-Directional Measurement Machine", The International Journal of Virtual Reality (IJVR), Vol.10, No.1, pp.37-43, 2011.

[9] E. Altantsetseg, K. Matsuyama and K. Konno, "Minimum Surface Area Based Complex Hole Filling Algorithm of 3D Mesh", The Journal of the Society for Art and Science Vol.14, No.2, pp.26-35, 2015.

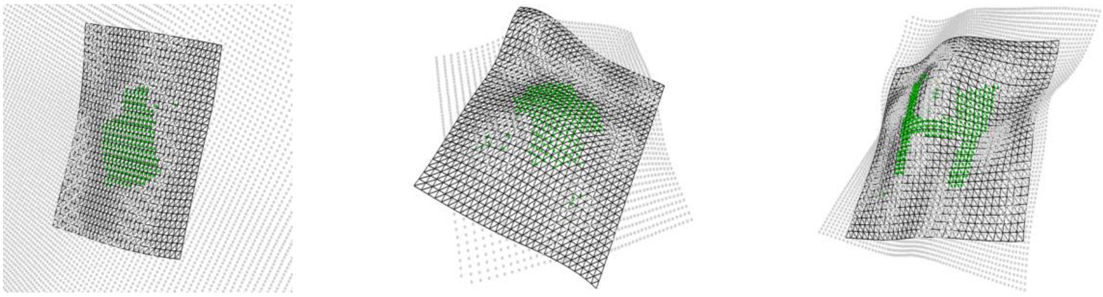
[10] Emelyanov,A., Skala,V., "Surface Reconstruction from Problem Point Clouds", Graphicon 2002, Int.Conf.on Computer Graphics, pp.31-37, 2002.

[11] Emelyanov,A., Skala,V., "A Method for Repaing Triangulations", Graphicon 2004 Int.Conf., pp.180-184, 2004.

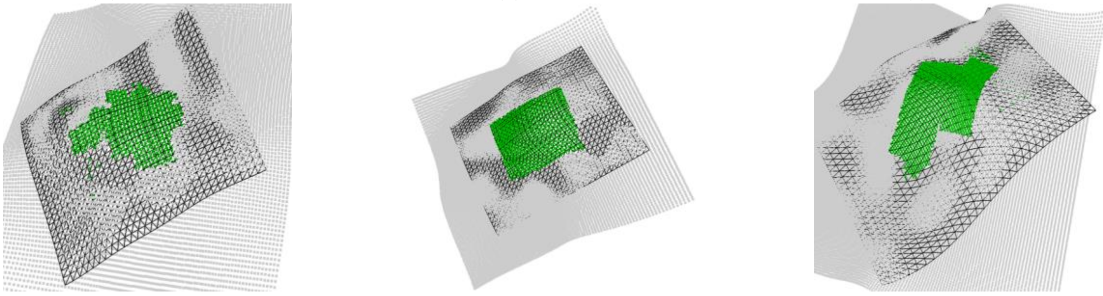
[12] Gerhard H. Bendels, Ruwen Schnabe and Reinhard Klein, "Detecting Holes in Point Set Surfaces", The Journal of WSCG, Vol.14(1-3), pp.89-96, 2006.



(a)Data A **(b) Data B** **(c) Data C**

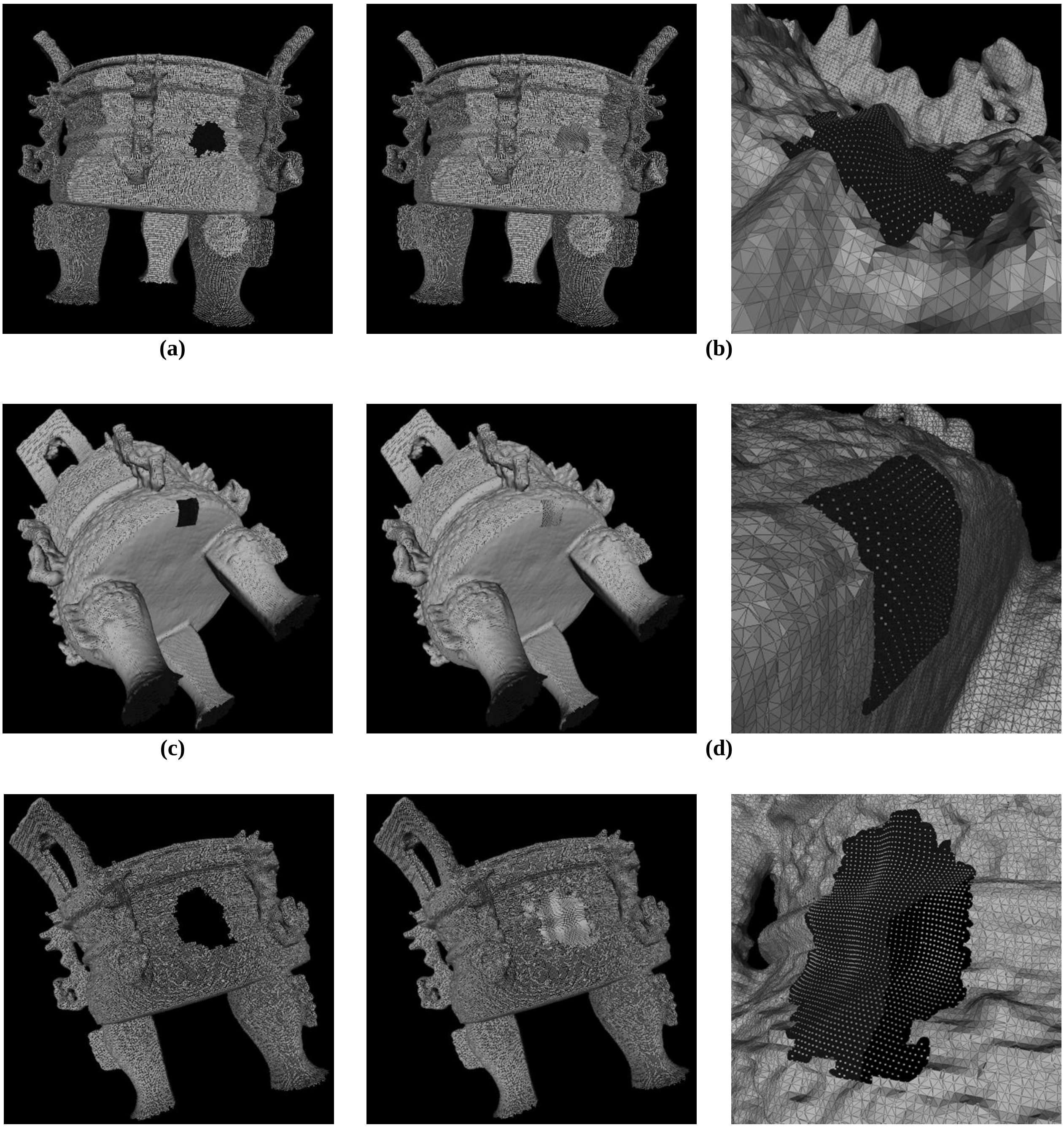


(d) Data D **(e) Data E** **(f) Data F**



(g) Data G **(h) Data H** **(i) Data I**

Figure 5. Result of applying to the experimental data



(a) (b)

(c) (d)

(e) (f)

Figure 6. Result of applying to the actual scanned data