

A Novel Hole Filling Method based on the Hybrid PSO-BP Algorithm

Hong Meng
Student

Department of mechanical,
Inner Mongolia university of
technology, Inner Mongolia,
China

Chunxiang Wang
Professor

Department of mechanical,
Inner Mongolia university of
technology, Inner Mongolia,
China

Yong Zhang
Reader

Department of mechanical,
Inner Mongolia university of
technology, Inner Mongolia,
China

ABSTRACT

This paper presents a novel hole-filling algorithm in reverse engineering (RE) domain that can fill holes directly from the point clouds-a hybrid PSO-BP approach: Firstly, boundary of the hole is detected and feature points on the boundary are extracted. Secondly, a hole filling method based on the commercial reverse engineering software (Geomagic and Imageware) is employed to cover the hole with a rough mesh. Finally, a hybrid PSO-BP algorithm is exploited to refine the original mesh. The performance of the approach proposed has been evaluated by applying it to two different scattered point clouds from real-world scanned objects-a bucket of an excavator and a gear. The experimental results show that the suggested approach performs quite well, it is able to deal with highly accurate and extremely complicated data points. Besides, it can handle shapes with delicate details as well, the favorable fidelity and efficiency make it a promising candidate for many practical applications.

General Terms

Neural network, Particle swarm optimization, Reverse Engineering

Keywords

Reverse Engineering, Point clouds, Hole filling, BP algorithm, PSO-BP algorithm.

1. INTRODUCTION

Holes are not uncommon in the process of reconstructing a physical model through an optical scanner for the following reasons[1]:1, defects of primitive models themselves, like the worn, deforming or scratching areas of the failed component to be reworked.2, due to surface reflectance properties, occlusions and accessibility limitations. 3. Due to the limitation of the scanning instruments themselves: scanning devices with high performance are usually not available in the field. 4. Parts under scanning should always be positioned beforehand, where their mounting surfaces are couldn't be sampled. Hence, it is crucial to fill the holes before moving to the downstream stages or the next step like model rebuilding, rapid prototyping and finite element analysis.

Many surface reconstruction methods, whether global or local in fitting the scattered point cloud data, could fill holes to a certain extent. In the early years, Davis [2] used volumetric diffusion to fill the gaps. Volume based approaches, in general, can guarantee water tightness and work well for complex holes, unfortunately they usually significantly change the underlying model during the process and incorrect topology may be generated in some cases. What's more, it is time-consuming and computationally more expensive. A few years later, researchers began to concentrated on interpolation

schemes such as the RBF network presented by John Branch [3]and the Poisson method proposed by Michael Kazhdan [4] to complete point clouds. These approaches are effective over relatively dense scans of geometry that contain small holes, however, when big holes are present in the mesh, the interpolating function cannot adequately fit the surface anymore.

To recover the missed features more effectively, numerous state-of-the-art techniques have arisen: The Long [5] algorithm simulated the energy diffusion process and progressively contracted a hole boundary until the hole was closed. It is effective in recovering models with sharp features, however, the method fails to develop an automatic hole boundary recognition technique in the whole geometry completion process and it is incapable of recovering those deficient point cloud surfaces containing high frequency geometry features. The Yann[6] method relies on the priori knowledge of the CAD model as a nominal mesh. The algorithm depended on a stress-strain deformation of the nominal geometry performed with the aim of minimizing the differences between the CAD model geometry and the point cloud while preserving the initial topology. It was thus possible to fill the holes with the deformed geometry. This approach is valid for a complex shape but is invalid when points data are affected by the digitizing noise. Further more, for parts need to be reverse Engineered, especially the mechanical parts from real applications, their original CAD models are no longer available because of the upgrading, being discontinued with others, or it is just designed for ordering as an assembly and not supportive for repairing, thus CAD models are not created at all. Hanh[7]'s method was designed for meshes and he adopted the strategy which first recovered the feature curve under user interactions, then filled the divided smooth sub-holes. Morfit suggested by Kangxue Yin [8] can recover some complex surfaces by interactively manipulating the curve skeleton and profile curve of the input point cloud model. The technique offers the handling of sharp features under simple user guidance but it is not the best at handling regions with large amount of missing data. An advancing front mesh technique was used by Wei zhao [9] , Wang xiaochao[10] , and Bin Xu[11] to cover the hole with newly created triangles. Since lapacian was used to depict geometric details of mesh surface model, it can make the filled mesh patch and the original mesh surface inosculate more naturally, but the filling result may lose some geometric details when the hole is too big. A neural network based B-spline surface construction has been proposed by Mehmet Ersin Yumer and Levent Burak Kara [12] , where an unstructured point set was projected into R2 by locally linear embedding (LLE) method in order to generate the parameterization. Then, a neural network was trained for

mapping from the parameter space into 3D space. Finally, the surface tessellation was performed by generating of a grid in the parameter space and using the trained network for mapping the parameter value to the 3D space coordinates. Since the author utilized feed-forward multilayer neural networks as learners without considering the inherent drawbacks of the network itself, the optimizing results need further improved. A global particle swarm optimization (PSO) method was employed by Akemi Gálvez and Andrés Iglesias [13] in order to find knot vector required for estimation of a B-spline curve from noisy measured points, however, the author didn't apply his method to real-world problems.

Inspired by the last two methods, a novel PSO-BP algorithm to fill holes directly on point cloud models is suggested, and it turns out to be an effective method for many practical applications. PSO algorithm has a much more profound intelligent background and can be performed more easily. As a global optimization algorithms, it has a better convergence speed. However, PSO algorithm may converge prematurely. Compared with PSO, BP method is more famous for their strong adaption, self-learning, and fault tolerance abilities, but BP network is also have some drawbacks, such as slow convergence speed, easily trapping in local minima during optimization and learning signals propagating one way. Therefore, a combination of PSO and BP algorithms would yield a hybrid algorithm with better optimization efficiency.

The whole filling process includes: boundary points detecting, basic surface (the rough mesh) reconstructing and vertex position adjusting based on the hybrid PSO-BP algorithm. Several tests have been made on both man made holes (figure 3) and naturally generated holes (figure 14) after scanning. And it can be clearly seen from the test results that the PSO-BP model is feasible and can obtain much better performance in hole filling process.

2. TECHNICAL OVERVIEW

Here, a scattered point cloud extracted from an excavator dipper teeth (see figure 1) is taken as an example to show the basic procedure of PSO-BP algorithm in the hole filling process as well as its simulation results.

The casted dipper teeth has no finish requirement on the outer surface but roughness of the internal must be strictly controlled because it is the mating area with the teeth holder, thus reverse engineering the inner surface becomes more challenging, and for this reason, the inner side was identified as the candidate surface for hole filling. Following figure 1 is a complete point data of a dipper teeth whose top cover has been partially removed for clarity purpose only. The inner cavity was extracted from figure 1 and was shown in figure 2. And in figure 3, a total of 1315 data points were circle selected from top of the inner cavity, where the hole covered area was calculated to 222.25mm^2 and the average curvature was evaluated as 1928.4 mm . All of the three figures as well figures in the following pages were scaled down to have a better fit with the paper layout.

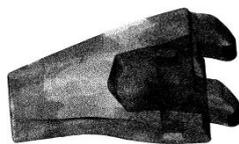


Fig 1: Excavator dipper teeth with top cover partially removed for clarity purpose only.

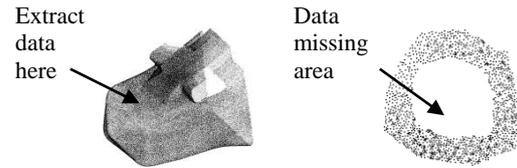


Fig 2: Inner cavity of the excavator dipper teeth

Fig 3: Data circled selected from figure 2

The problem of filling holes can be divided into two sub-problems: hole identification and construction of the missing data using the available data near the holes. Differing from the above techniques, the basic surface here was reconstructed in an automatic hole detecting and hole-filling process.

2.1 Hole Detecting and Hole Filling

The algorithm starts by locating hole boundary regions, because the property of the surface being recovered inherently is the property of the local neighborhood of the hole. In order to complete the hole more accurately, it is required to seize the local neighborhood more formally.

By using the built-in hole detecting feature that comes with the software, the author came up with a quick hole detecting method. First, all boundaries were automatically detected in the form of boundary lines with the help of Geomagic 2015 which include not only the boundary of the hole but also all boundaries of the whole selected point cloud (as shown in figure 4). Second, by manually removing the undesired boundaries the hole boundary was left. The inner boundary line was then converted into limited boundary points (see figure 5). The basic principle of the hole detecting method is as follows: A hole consists of a closed path of edges of boundary triangles that have at least an edge, which is not shared with any other triangle. Recently, this idea is quite popular in the hole detecting process and has widely acknowledged by many scholars. Distinct from others, the author discovered and took full advantage of this inherent capabilities of the software eliminating the programming process which in turn is more time-saving.

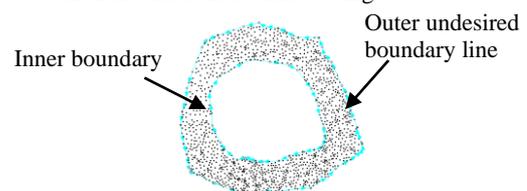


Fig 4: Inner and outer boundaries

Once a boundary edge was found, the author operated the software to trace the entire boundary. A ring of points around the boundary was generated (as seen in figure 6) and used for the hole filling procedure to eventually fill the hole and used for training the network as well. An important feature of this technique is to preserve the inherent adjacency or connectivity information between the point sets, and guarantee that the reconstructed patches blend smoothly into the original surface.

The hole was then fitted with the Imgeware13.2 to fill the inside of the hole using neighboring edge.

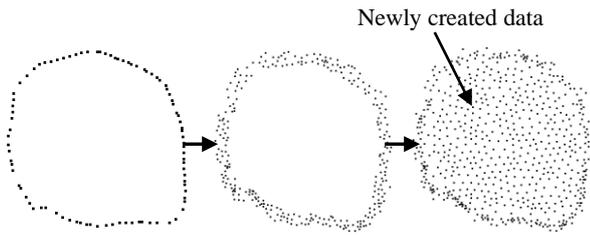


Fig 5: Hole boundary point **Fig 6: Points around the boundary** **Fig 7: Hole filled automatically**

2.2 Surface Refining by PSO-BP

Algorithm

The surface fitted by the software must be improved and optimized to make it a natural transition with the surrounding surface and to make the repaired surface perfectly reflect the real model. That's why PSO-BP algorithm is proposed. Here the effectiveness of the proposed approach was demonstrated by using it in real data sets and the following content shows how it can significantly improve the overall quality of a roughly fitted surface.

2.2.1 Prediction Model of BP

2.2.1.1 The Optimum Number of Neurons in the Hidden Layer

The structure of the BP neural network is determined by its input and output data. In the experiment shown here, there are two inputs and one output, so the number of neurons in the input layer was set to be 2 and number of neurons in the output layer was set to be 1.

To determine the optimal number of hidden neurons is a crucial one since it will directly affect the prediction results. Here, the Hecht–Nelson method was applied to determine the number of nodes in the hidden layer: when the number of nodes in the input layer is determined as n , the number of nodes in the hidden layer should be $2n + 1$ [14]. So the number of neurons in the hidden layer here was set to be 5. With two neurons in input layer, five neurons in hidden layer, and one neuron in output layer, the structure of the BP neural network here is 2-5-1.

2.2.1.2 Definition of Transfer Function and Train Function

The tan-sigmoid transfer function is most commonly used and was assigned in hidden layers for processing the inputs layer

$$\text{as } f(x) = \frac{1 - e^{-\beta x}}{1 + e^{\beta x}} \quad (1)$$

and the log-sigmoid transfer function was assigned in hidden layers for processing the output layer as

$$f(x) = \frac{1}{1 + e^{-\beta x}} \quad (2)$$

2.2.1.3 The Optimum Training Function in the Network

Here the Levenberg-Marquardt algorithm[15] was chosen as the learning algorithm of BP. The essence of LM method is a combination of gradient steepest descent method and Gauss–Newton algorithm, which makes the best of these two methods, and to a certain extent, can overcome the shortcomings of the traditional BP network, such as easily be trapped to the local minimum and slow convergence speed.

2.2.1.4 The Standard Error

The mean square error MSE was used as a quantitative index to capture the prediction accuracy for BP neural network. And in order to find the global minimum, the precision was pre-set to 0.00001.

2.2.2 Using PSO to Optimize BP

The main idea of PSO-BP model is to use PSO algorithm to optimize the initial connection weights and thresholds of the traditional BP method in order to improve the predicting performance of a normalized BP. Each particle in the particle swarm corresponding to the weight and threshold value of BP neural network. And the output error of BP neural network was taken as the fitness function value of PSO algorithm. The fitness function is defined as follows:

$$F = k(\sum_{i=1}^N \text{abs}(y_i - o_i)) \quad (3)$$

Where: N is the number of neurons in the output layer, y_i is the desired output of the i^{th} neuron and o_i is the actual output of the i^{th} neuron.

2.2.3 Data Preparation

Training and testing data: point clouds in figure 6 was used as samples to train the network. Among them, 80% were used for training while the remaining were used for testing.

Predicting data: the newly created points at the center of figure 7 were used for predicting.

2.2.4 PSO-BP Algorithm Training and Testing

Load the training and testing data, their coordinate value x and y were separated from z and chosen as the input, and z was taken as output. The network output was compared with z during each iteration, and the mean square error(MSE) between the simulated output and z was calculated. The iteration was continued until MSE was less than the above predetermined training goal 0.00001. At this time, save the network for further predicting.

2.2.5 PSO-BP Algorithm Predicting

Again, load the predicting data, separate their coordinate value x and y from z and take them as the input, simulate and record the output, i.e., PSO-BP predicting result Z . Integrate the input x and y with the predicted value Z making an optimized data sets, save it as a file. Combine these newly saved points with the points adjacent to the hole(namely data in figure 6), the hole filling process is finished.

2.2.5.1 Accuracy Evaluation

In the study, deviation between the constructed surface and the measured point cloud was utilized as a measurement to evaluate the effectiveness of the proposed PSO-BP algorithm. Deviations were illustrated in the right color vector graphics.

2.2.5.2 Results and Discussion

As you can see from following figure 8, deviation of the initial mesh can reach up to 0.132 mm which is far from enough for some applications. By contrast, the maximum deviation is only 0.037mm after the surface was refined by PSO-BP method as illustrated in figure 9. Furthermore, the whole surface optimized by PSO-BP model looks much more smoother and the newly filled areas blends smoothly to the neighborhood.

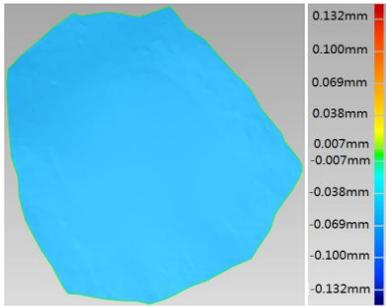


Fig 8: Hole filled by software and the deviation

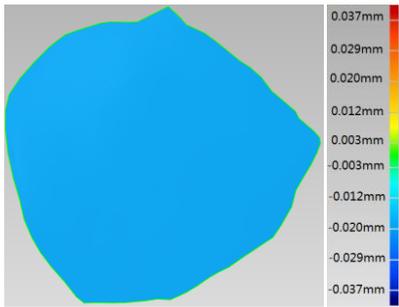


Fig 9: Hole filled by PSO-BP algorithm and the deviation

For the surface refined by the PSO-BP algorithm, namely surface in figure 9, position of the highest deviation was also analyzed and was shown in the following figure10.

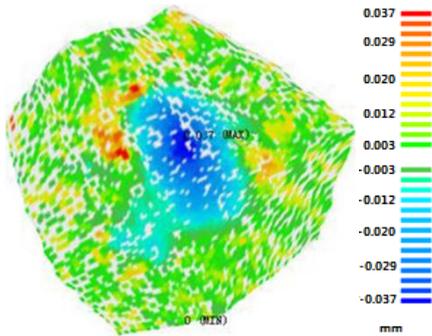


Fig 10: Position of the highest deviation on the surface of figure9

2.2.6 Algorithm Flowchart

A flowchart for the PSO-BP algorithm is shown in Figure 11.

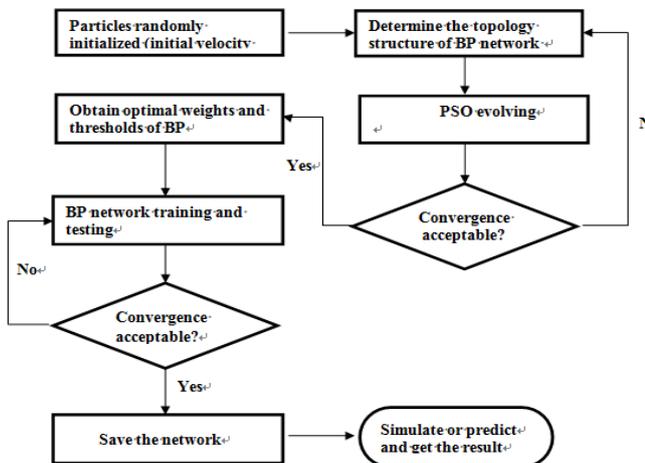


Fig 11: Flowchart for the proposed PSO-BP algorithm

2.3 Analysis of Results and Discussion

To further assess the ability of the PSO-BP network in reverse engineering domain, BP and PSO method was programmed and executed respectively for missing data prediction. Computations were performed on a 2.93GHz Intel (R), Core (TM) i3 CPU with 4GB RAM computer, and R2012b Matlab was used as the platform for the programming. The two programs were randomly ran for 5 times individually, the numerical experiments were summarized in table 1 where time was listed in table 2 as seconds and the MSE refers to mean square error between the predicted value and the desired one.

Table 1. Statistical Result

| Execution Times | MSE | |
|-----------------|------|--------|
| | BP | PSO-BP |
| 1 | 0.34 | 0.045 |
| 2 | 0.29 | 0.041 |
| 3 | 0.27 | 0.038 |
| 4 | 0.26 | 0.035 |
| 5 | 0.22 | 0.032 |
| Max. | 0.34 | 0.045 |
| Avg. | 0.28 | 0.038 |

Table 2. Computing time

| Execution Times | MSE | |
|-----------------|------|--------|
| | BP | PSO-BP |
| 1 | 3.51 | 7.16 |
| 2 | 3.18 | 7.03 |
| 3 | 3.06 | 6.92 |
| 4 | 4.41 | 7.3 |
| 5 | 4.45 | 7.06 |
| Max. | 4.45 | 7.16 |
| Avg. | 3.72 | 7.09 |

The following two figures are the results of the simulation: figure 12 demonstrates the simulation errors of BP and PSO-BP algorithm respectively for the fifth operation. As clearly shown in the picture, prediction error of BP method is much bigger than that of the PSO-BP algorithm. The prediction accuracy is well improved after BP method was optimized by the PSO algorithm.

And figure 13 illustrates the deviation between the estimated output and the desired one after the PSO-BP algorithm was utilized for predicting. The result is satisfactory indeed.

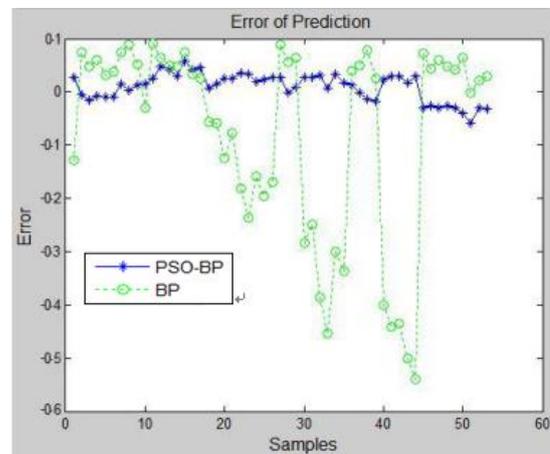


Fig12: Error of BP versus PSO-BP

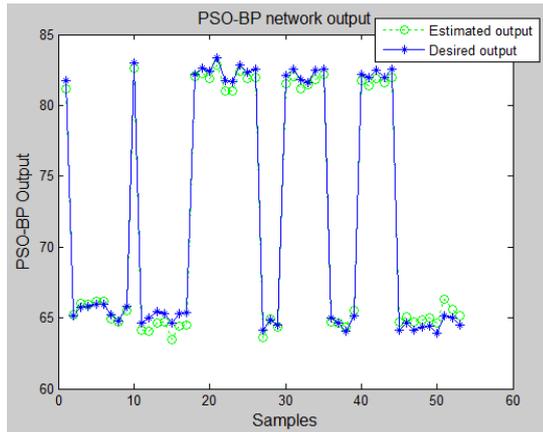


Fig 13: Simulation output of PSO-BP

3. APPLICATION

The method was applied to a gear scanned by a local factory. Hole is inevitable in the digitized point clouds as obviously shown on one of the 25 gear tooth (figure 14) after the scanning process was done. To examine the proposed method, the hole was first filled using the software and then optimized by the PSO-BP algorithm. And the surfaces constructed by the mouse-based computer interfaces and by the proposed PSO-BP algorithm were compared as well.

3.1 Application in Gear Teeth

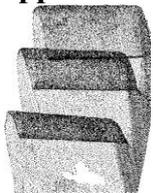


Fig 14: Hole at a gear teeth



Fig 15: Hole boundary



Fig 16: Hole Neighborhood



Fig 17: The roughly filled hole

The above figure 14 shows part of a gear point cloud, and the illustrations 15 to 17 show the whole filling process: boundary points detecting, neighborhood generating, and basic surface (the rough mesh) reconstructing.

And following are the compare results between the basic surface created by the software and after the surface was refined by the hybrid PSO-BP algorithm.

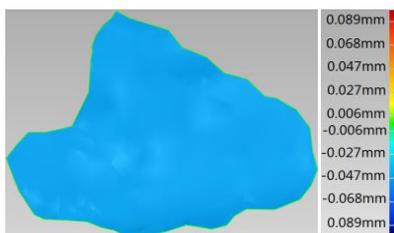


Fig 18: Hole filled by software and the deviation

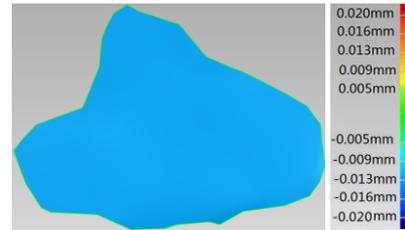


Fig 19: Hole filled by PSO-BP and the deviation

By visually inspected, the rough mesh displayed in figure 18 is far less smoother than that in figure 19. And this can also be justified by the two deviations: the former is up to 0.089mm, while the max deviation of the latter is 0.02mm. And for the surface optimized by the suggested PSO-BP method, position of the highest deviation was also studied and illustrated in the following figure 20.

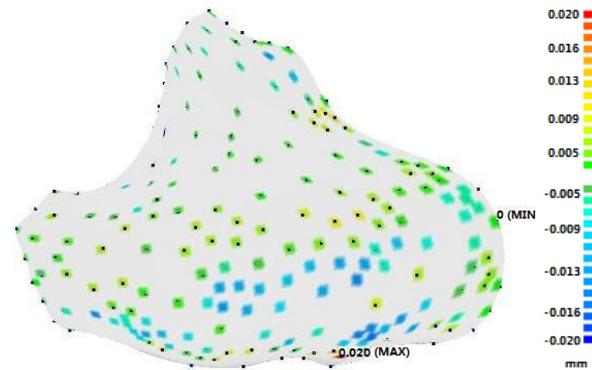


Fig 20: Position of the highest deviation on the surface of figure 19

4. CONCLUSION

In this paper, a new approach for hole filling is presented. A hybrid PSO-BP algorithm is utilized to refine the basic surface newly created in the hole area. The scheme is based on the idea of detecting and filling holes with software included techniques. After the basic surface is created and digitalized, the proposed PSO-BP neural network optimization models is then applied to compute an optimal vertex position automatically.

The presented method is conceptually simple to understand, easy to implement and does not require further pre- or post processing. Since the suggested hole filling technique is directly from unorganized point clouds thus eliminating the triangulating process and is more time-saving which is one of the most remarkable characteristics. As it turns out, it is especially practicable for large sets of unevenly scattered data which in future may be applied to help fitting any possible surfaces in mechanical components with complex shape and high accuracy requirements. Moreover, unlike other previous schemes in dealing with holes, the suggested method can obtain truly identical surfaces automatically when needed without assuming any underlying function of surface.

Indeed, a case worth mentioned is that the computational time required by the new method turns out to be slightly slower than that of the normalized BP, wavelet or genetic algorithm, and that's one of the areas for further improvement. In future, the researcher would continue investigating on how to accelerate the convergence speed. And since the presented PSO-BP algorithm can work with any surface fitting algorithm, in the next stage, the author would try to use other creative methods instead of the software based interpolation

algorithm to construct the basic surface before the PSO-BP algorithm is applied, aiming at further enhancing the capability of the proposed algorithms in handling shapes with delicate details.

5. ACKNOWLEDGMENTS

This work is supported by the Natural Science Foundation of Inner Mongolia, China (No. NJZY16167). The authors would like to thank the anonymous reviewers for their valuable comments and insightful suggestions.

6. REFERENCES

- [1] MF Lu, HB Wu, C Liu .etc., Classification and cause analysis of terrestrial 3D laser scanning missing data[J], Remote sensing information, Vol.28, No.6 Dec., 2013.
- [2] J Davis, SR Marschner, M Garr, M Levoy, Filling holes in complex surfaces using volumetric diffusion[C], International Symposium on 3D Data Processing Visualization and Transmission, 2002(28):428 - 441.
- [3] J Branch, F Prieto, P Boulanger, A Hole-Filling Algorithm for Triangular Meshes Using Local Radial Basis Function[C], Proceedings of the 15th International Meshing Roundtable, 2006.
- [4] M Kazhdan, M Bolitho and H Hoppe, Poisson Surface Reconstruction[C], Euro-graphics Symposium on Geometry Processing 2006, 61-70.
- [5] Y Long, Q Yan, CX Xiao, Shape-controllable geometry completion for point cloud models[J], The Visual Computer, 2016: 1-14.
- [6] Y Quinsat, C lartigue, Filling holes in digitized point cloud using a morphing-based approach to preserve volume characteristics[J], International Journal of Advanced Manufacturing Technology, 2015, 81:411–421.
- [7] HTM Ngo, WS Lee, Feature-First Hole Filling Strategy for 3D Meshes[J], Computer Vision, Imaging and Computer Graphics. Theory and Applications(2013).
- [8] K Yin, H Huang, H Zhang .etc., Morfit: Interactive surface reconstruction from incomplete point clouds with curve-driven topology and geometry control[J], ACM Transactions on Graphics, 33(6), 202:1–202:12 (2014).
- [9] W Zhao, S Gao, H Lin, A robust hole-filling algorithm for triangular Mesh[J], Visual Compute (2007) 23: 987–997.
- [10] X Wang, J Cao, X Liu, B Li, Advancing front method in triangular meshes hole-filling application[J], Journal of computer-aided design& computer graphics, Vol.23 No.6 June 2011.
- [11] B Xu, Z Li, Y Tan, Feature Based Hole Filling Algorithm on Triangular Mesh[J], Computer and Computing Technologies in Agriculture VII(2014).
- [12] ME Yumer, LB Kara, Surface creation on unstructured point sets using neural networks[J], Computer-Aided Design 44 (2012) 644–656.
- [13] A Gálvez, A Iglesias, Efficient particle swarm optimization approach for data fitting with free knot B-splines[J], Computer-Aided Design, 43 (2011) 1683–1692.
- [14] C Ren, N An, J Wang .etc., Optimal parameters selection for BP neural network based on particle swarm optimization: A case study of wind speed forecasting[J], Knowledge-Based Systems 56 (2014) 226–239.
- [15] N Liu, H Yang, H Li .etc. BP artificial neural network modeling for accurate radius prediction and application in incremental in-plane bending[J], The International Journal of Advanced Manufacturing Technology, (2015) 80:971–984.