

RESEARCH

Open Access



Research and application of personalized human body simplification and fusion method

Lulu Ma¹, Ke Zhang² and Dianjie Lu^{3*}

Abstract

In crowd simulation, 3D (three-dimensional) character modeling is an important topic since the appropriate character models are helpful to improve the efficiency and realism of crowd simulation. The reconstructed 3D character model based on Kinect has a strong sense of reality and low cost. However, these models are all complex and cannot be applied to large-scale crowd simulation directly. In this paper, we propose a novel personalized human body modeling method for mass crowd simulation based on Kinect. The human modeling process is divided into head modeling and torso modeling, and then they are fused into each other to build a personalized human body model. This method can be divided into two parts: In the first part, a simplified method is presented based on edge curvature and area of error. In addition, to preserve the detail characteristics of model, the way of interactive operation is introduced. In the second part, the automatic fusion for the simplified head model and body model is made by using the improved FCF (fusion control function) fusion method. Finally, a hierarchical database for the personalized human body models is built. The experiment results show that the method proposed in this paper has high efficiency and good robustness in practical applications.

Keywords: Kinect, Model simplification, Detail features, FCF

1 Introduction

Traditional 3D (three-dimensional) human body models usually are captured by the structured light or laser scanner. Although it can obtain high-precision personalized 3D human body model, the cost is high and the operation is complicated [1]. The Kinect device launched by Microsoft can make use of infrared technology to achieve the fast acquisition of three-dimensional information at low cost. This breakthrough has greatly promoted some applications using 3D technology, such as human action recognition based on Kinect, skeleton modeling, face recognition, and 3D reconstruction of the scene, which have become a research hotspot in the related fields [2–4]. Because of its low cost and simple operation, Kinect depth camera is also

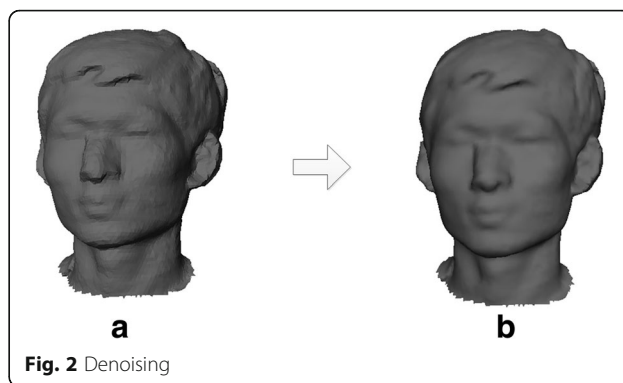
used as a scanner to rapidly build personalized 3D human body models in real time [5, 6]. However, the point cloud density of the model of Kinect scanning is too large, so the 3D human model constructed by it is difficult to be widely applied. For example, in the crowd simulation scene, the real-time rendering speed of human body is required to be fast due to the large scale of the crowd. If the 3D human body model scanned by Kinect is applied directly to the group simulation, it will undoubtedly increase the system cost and reduce the efficiency of crowd simulation. Therefore, how to preserve the necessary details of the model while simplifying the model with high similarity to original model is a very meaningful problem.

At present, the simplified method commonly used for three-dimensional models is the simplified method in CGAL (computational geometry algorithms library). CGAL is a computational geometry algorithms library written in C++. In the simplification of the 3D mesh, the simplified method of edge collapse is used [7, 8], and the

* Correspondence: ludianjie@sina.com

³School of Information Science and Engineering, Shandong Normal University, Shandong Provincial Key Laboratory for Distributed Computer Software Novel Technology, Jinan 250014, China
Full list of author information is available at the end of the article

Lindstrom-Turk method is used to calculate the collapse cost of each side. Although the efficiency is high, the accuracy is not high, and the effective detail features of the model cannot be retained. The simplified method based on the fillet surface reconstruction proposed by Peng et al. [9] realizes the simplification of the fillet surface, which is applicable to the model with fillet characteristics but not universal. Other simplified methods for 3D models, such as the 3D model simplification algorithm with texture proposed by Feng and Zhou [10], consider the geometric information of the model and the geometric error of the texture information. Zhou and Chen [11] put forward a mesh model simplification algorithm based on polygon vertex normal vector, which is a simplified method of visual feature optimization. Quan et al. [12] propose the geometric model simplification method based on the region segmentation. This method needs to keep the detail features and introduce the region segmentation principle of the image. A mesh model region segmentation method based on curvature for region growing is proposed, and then it is simplified according to the ratio of the number of triangles in the region. Zhang et al. [13] put forward a new simplification method of terrain model based on divergence function. This method combines the discrete particle swarm theory to simplify the terrain model based on the hierarchical structure of the implicit quadtree. An improved mesh simplification algorithm based on edge collapse was introduced in [14]. In this algorithm, the quadric error metric was utilized to compute the vertex significance and control the sequence of edge collapse. Sanchez et al. [15] use an estimated local density of the point cloud to simplify the point cloud. In this method, the point clouds are clustered by using the expectation maximization algorithm according to the local distribution of the points. Then, a linear programming model is applied to reduce the cloud. Han et al. [16] propose a point cloud simplification method with preserved edge points. In this work, the authors try their



best to retain the edge points since these points have more significant properties than non-edge points. First, a least square plane is constructed by using the topology relationship of the points and normal vectors and then each point in the vicinity is projected to the fitted plane. Next, the edge points can be extracted according to the homogeneity of projection. These detected edge points are retained in the decimating process. As For the non-edge points, the authors delete the least important points until the predesigned simplification rate is satisfied. In [17], the incremental segmentation and triangulation of planar segments from dense point clouds are studied to enhance the quality and efficiency. In this paper, the authors proposed a point-based triangulation algorithm to improve the planar segment decimation and triangulation in a gradually expanding point cloud map as well as a polygon-based triangulation algorithm. Both of the two algorithms can produce more accurate and simpler planar triangulations. Although the above method can simplify the model, it cannot keep the effective details of the model, and it is inefficient when dealing with the models with large amount of points.

When the whole 3D body model is collected by Kinect, the amount of data is huge and the later processing cost is big instead of its convenience. The

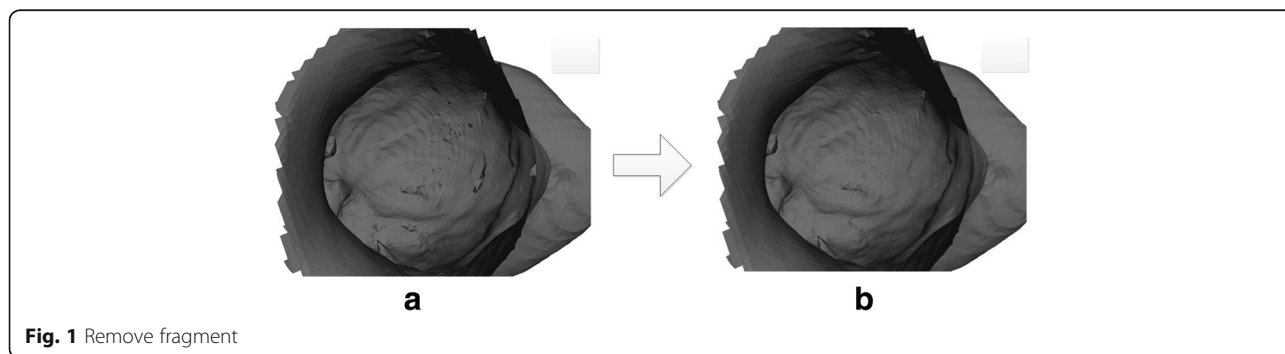


Fig. 1 Remove fragment

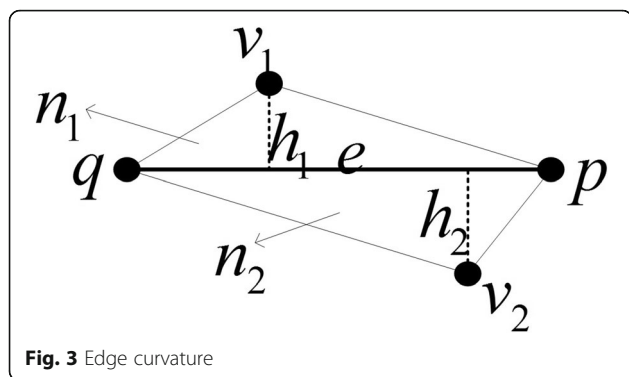


Fig. 3 Edge curvature

huge amount of 3D information and data is mainly manifested in three aspects: dense point, dense edge, and dense surface. Thus, the simplification of 3D model is mainly from the three aspects of point, edge, and surface. In this paper, we first use Kinect to get the head with the most personalized features, and then use the edge collapse simplification method based on edge curvature and area error. In addition, the interactive approach is used to preserve the detail characteristics and simplify the model. Furthermore, we also use the improved FCF (fusion control function) [18] model fusion method to realize the seamless integration between models automatically. With this model, the complete character models can be reconstructed to build a database of LOD (levels of detail) character model. The simplified 3D body models can reduce system overhead and improve system efficiency when they are applied to crowd simulation.

2 The simplification and fusion method

The related work of 3D human reconstruction algorithm proposed in this paper was three-dimensional scanning technology, model denoising, model simplification, model fusion, and simulation experiment.

Firstly, the head model was obtained by using a cheap Kinect depth camera, and then a smooth model was obtained by removing the fragments in the model, the model denoising, and other preprocessing operations. Secondly, we simplified the complex head model by using the simplified method of preserving model features based on edge curvature and area error proposed in this paper. Finally, the improved FCF model fusion method was used to fuse the head model and the body model, and the personalized hierarchical model library was built, which was finally applied to group simulation.

2.1 Preprocessing

The denoising of the initial model can be divided into two steps: One is to delete the fragments generated during scanning. In this paper, we used DFS (depth first search) algorithm to delete fragments. The other is to use the weighted Laplace smoothing algorithm [19] to remove the tiny noise on the model and make the model as smooth as possible. The Laplace smoothing algorithm has a low computational complexity and can control the details of the model very well in operation. Therefore, this paper used the Laplace smoothing algorithm to remove the small noise in the model.

The topological structure of the model was mapped to a graph, and the connectivity of the whole graph can be obtained by using the method of DFS. The largest connected graph was the structure that needed to be retained, and the rest of the disconnected graph was deleted. Figure 1a is the graph before deleting the fragment, and Fig. 1b is the graph after the fragment is deleted.

The specific operation of the Laplace algorithm is as follows: The 3D position of the vertex is moved to the center of gravity position of the surrounding vertices to minimize the difference between the vertex

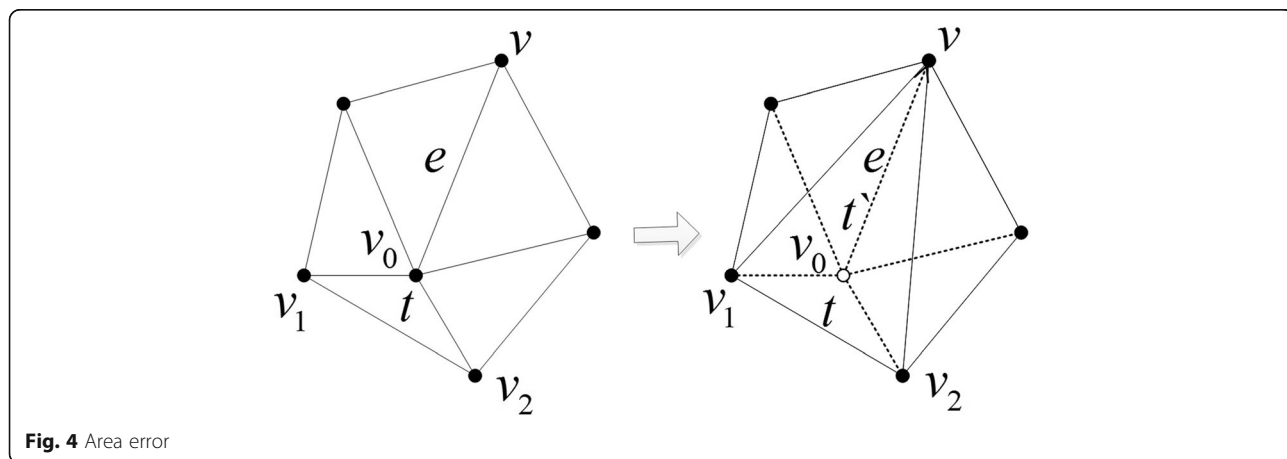


Fig. 4 Area error

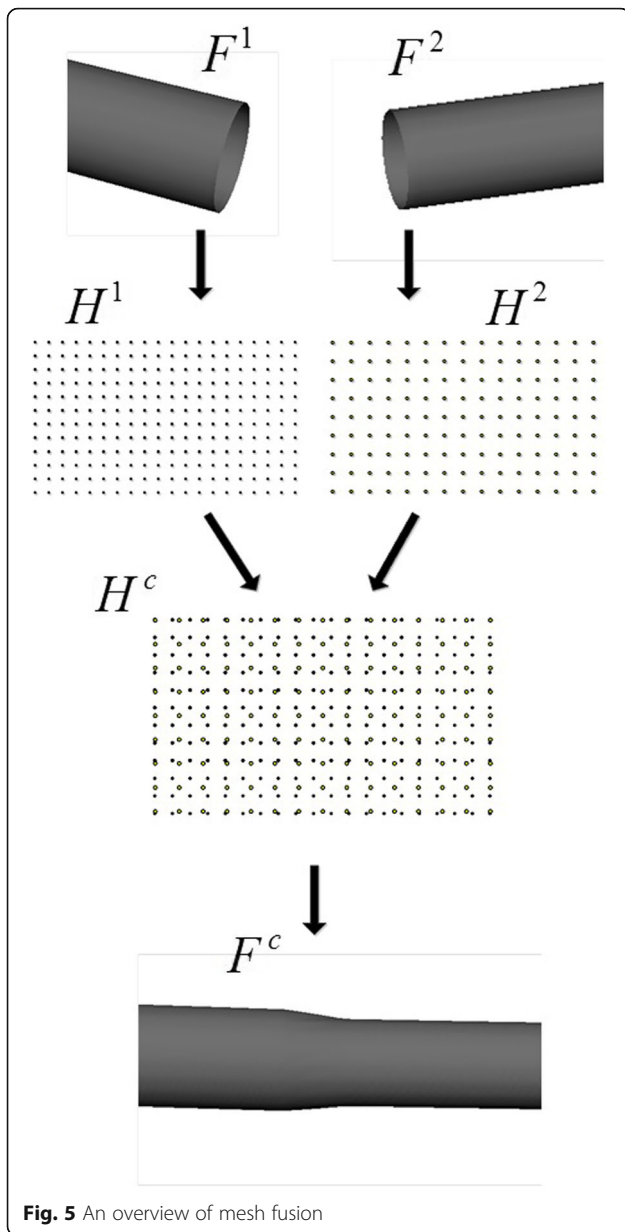


Fig. 5 An overview of mesh fusion

and the surrounding vertexes. For every point in the model, the space position of the point is recalculated according to the position information of the surrounding vertexes. The Laplace smoothing formula is as follows:

$$\bar{x}_i = \frac{1}{N} \sum_{j=1}^N x_j \tag{1}$$

N is the number of vertexes around the current point and \bar{x}_i is the new coordinate of the i -th vertex.

In practice, N cannot be too large. If it is too large, the details may be lost. If N is too small, the smooth effect will not be achieved. Therefore, an improved Laplace smoothing algorithm was used in this paper, which was weighted by Gauss weighted method. For the point far away from the center point, its influence on the center point was reduced, and for the point near the center point, its influence on the center point was increased. Figure 2a is a model before denoising, and Fig. 2b is a model after denoising. It is obvious that the preprocessing method proposed in this paper can effectively remove the small noise on the model.

2.2 Detail features preserving

According to the different geometric elements, mesh model simplification method is divided into vertex deletion, edge collapse, triangle deletion, and patch deletion. Edge collapse operation is based on half edge structure and an edge with the smallest triangular patch cost is collapsed in order to achieve the purpose of deleting the simplified model of triangular patch.

When the mesh model is simplified by the simplification method based on geometric elements, the higher the simplification rate, the more serious loss of the details of the model, and the detail features of the model cannot be preserved. The purpose of model simplification is to maintain the detail features of the model to the maximum extent on the basis of

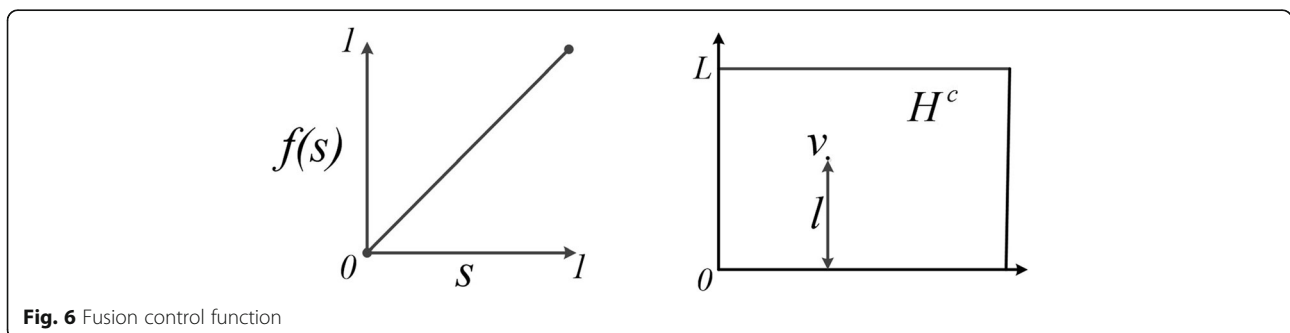


Fig. 6 Fusion control function

reducing the scale of the model. Based on the edge collapse method, an interactive method was proposed in this paper, which can effectively retain the detail features of the model and make the model have higher identifiability.

The detail features of the model can be reflected by the edge curvature and the area change of the triangular patch on the model. If the curvature of the edge is large in a region, the features of the model are obvious here. The change of the area of the related triangle patch caused by the collapse edge also reflects the features of the model here. The smaller the change, the smoother. Therefore, the edge collapse cost can be defined as two errors: one is the curvature of the edge, and the other is the change of the area brought by the deletion point.

As shown in Fig. 3, the way to calculate the edge curvature is:

$$E_c = \beta(1 - \cos\alpha) = \left(\frac{l}{h_1} + \frac{l}{h_2}\right)(1 - n_1 n_2) \quad (2)$$

Among them, $l = \|q - p\|$, $h_i = d\{v_i, e\}$.

As shown in Fig. 4, t is the reproduced triangular patch while deleting the triangular patch T . This process can be achieved by rotating the triangular patch T by an angle. When the angle of triangle rotation is set as θ , the error Q_t resulting from the deletion of one surface t is as follows:

$$Q_t = l_t \times \theta \quad (3)$$

Among them, $l_t = (A_1 + A_2)/2$, A_t is the area of the triangle $t = (v_0, v_1, v_2)$, A_2 is the area of the triangle $T = (v, v_1, v_2)$. The computation angle θ involves trigonometric operation, so the computation speed is slow. In order to improve the computation speed, let $\theta = 1 - n_t n_v$, in which n_i is a normal vector of the surface i . As shown in Fig. 4, the area error E_a caused by the deletion edge is:

$$E_s = \sum_{t \in P} Q_t \quad (4)$$

P is the set of all triangle patches connected to v_0 . Therefore, the error of the edge collapse is as follows:

$$E = E_c + E_s \quad (5)$$

Although the detail features of the model can be retained by using this method, with the improvement of the rate of simplification, the detail features of the model will also be lost gradually. In order to keep the detail features of the model, an interactive method was used in this paper to retain

the detail features of the model. The simplified operation to preserve the detail features is shown in Algorithm 1:

Algorithm 1: The simplified operation

Step.1 Let the set of edges of the reserved area $P = \{e_i\}$, among which $i = 0, 1, 2, \dots, c$ denotes the number of edges of the reserved area.

Step.2 Calculate the collapse cost E_j of the rest non-reserved edges, among which, j represents the j -th non-reserved edge.

Step.3 Calculate the average value $\bar{E} = \frac{1}{n} \sum_{j=1}^n E_j$ of the edge collapse cost, n denotes the number of all edges.

Step.4 Let $E_{e_i'} = (\text{rand}() + a)\bar{E}$, among which $E_{e_i'}$ represents the collapse cost of the edges that need to be reserved, $(\text{rand}() + a)$ represents the random weight of \bar{E} and a is a constant.

Step.5 According to the order of collapse cost, we carry out the edge collapse operation from small to large until we reach the target result.

The complexity of the algorithm is the calculation of the folding cost of each edge. The folding cost of each edge includes two parts: the edge curvature and the area difference. For each edge, the value of both of the two parts should be calculated. Generally, the simplification of the whole model can be realized if the order of all edges is sorted by the size of all edges according to the folding cost of all edges and the edge folding operation. However, the specific details on the model cannot be preserved and it cannot guarantee the real sense of the model through the simple model can be obtained according the simplification method mentioned above. Therefore, we introduce an interactive method to mark the area that needs to be preserved by a manual way when choosing the folding edge, so as to retain the local features with special significance. In this paper, a random weight is set up for the folding cost of the edge of the reserved region, that is $(\text{rand}() + a)$. The average

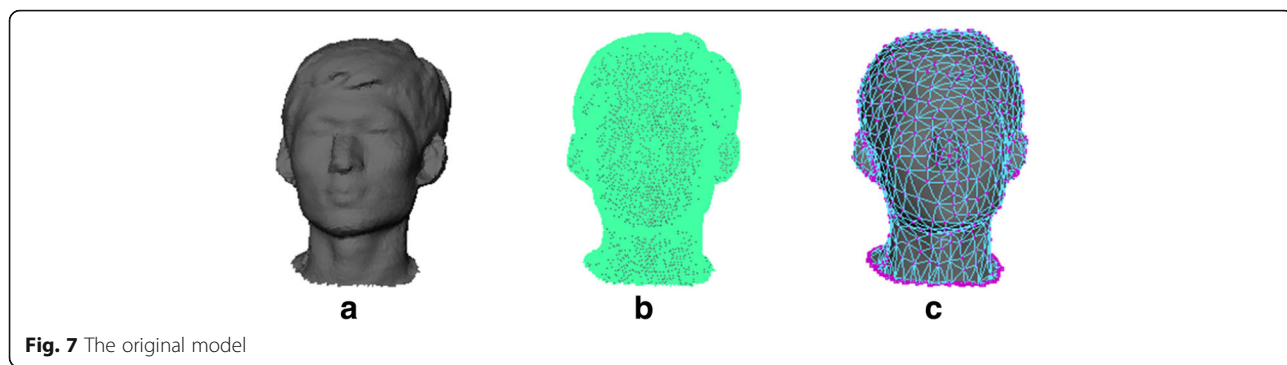


Fig. 7 The original model

folding cost of all edges is added to obtain the edges of the reserved area with a larger folding cost, so that the edges of the feature area can be retained effectively. For the head model collected by Kinect, we can adjust the value of a to control the simplification rate of the reserved area. We introduce a function $\text{rand}()$ to reduce the effect of noise on the model. When a takes a large value, it can completely preserve the details of the feature area; when a is taken for smaller values, the simplification ratio of the reserved area becomes larger, which cannot achieve the purpose of preserving the details of the model. When a is taken between the two values, the reserved area can be simplified to a suitable level.

The proposed algorithm can effectively retain the detail features of the model. Experiments show that this way of processing can effectively retain the important features of the model, and it can also retain the features of the model well when the simplification ratio is high.

2.3 Fusion of models

The simplified head model also needs to be fused with the body model to get a personalized human model. Kanai et al. propose a FCF-based mesh model fusion method. This method needs to map the model to

two-dimensional space and then map it to 3D space according to FCF function, so as to achieve the integration between models. However, the high computational complexity does not apply to the fusion of large models, and it needs to choose the fusion area manually, which cannot realize the automatic integration between models. Therefore, in this paper, an improved FCF fusion method based on boundary edge was applied to achieve the automatic fusion between two large-scale models. The first step was to determine the fusion area according to the boundary edge of the two models. As shown in Fig. 5, the fusion areas F^1, F^2 were determined and then mapped to the two-dimensional space H^1, H^2 to achieve the mapping from three dimensions to two dimensions. By using the method in literature [20], we can approximately calculate the locations of all points on H^1 and H^2 . The second step was to merge H^1 and H^2 and get H^c . H^c included two kinds of vertexes: one was the original vertex that is directly inherited from F^1, F^2 , and the other was the new point generated by cross computing. The third step was to reconstruct the blending surface F^c according to every point in H^c . Each point in H^c contained two parts of three-dimensional information, one was the three-dimensional information on F^1 , and the other was the three-dimensional information on F^2 . The transition curved surface F^c of F^1 and F^2 can be obtained by FCF method. As shown in Fig. 6, we define a

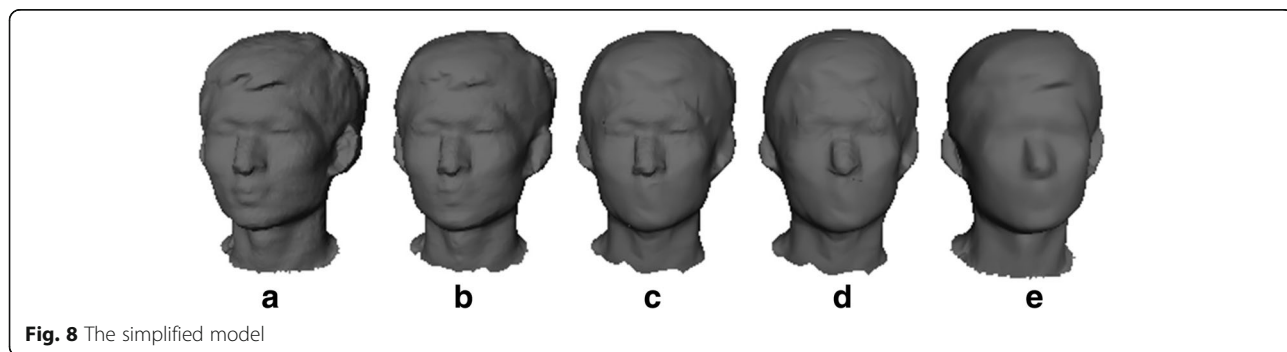
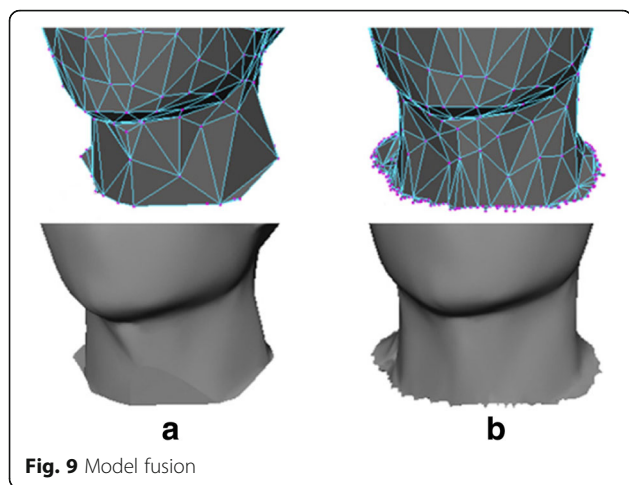


Fig. 8 The simplified model

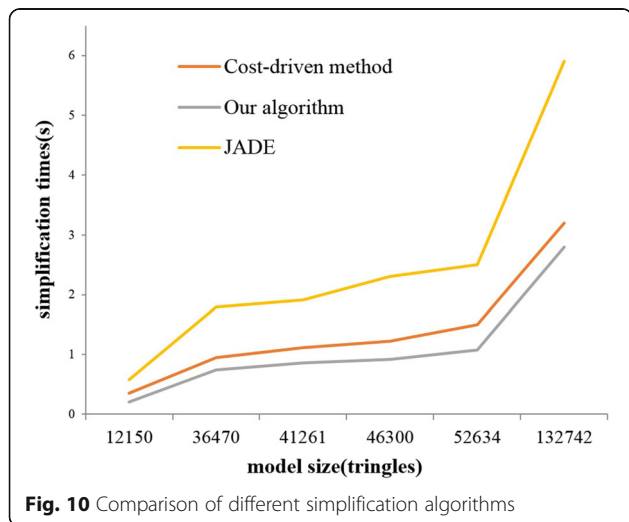


non-uniform cubic B-spline curve interpolation algorithm, called $f(s)$ ($0 \leq f(s) \leq 1$), to control the fusion models.

The coordinate of the vertex v^c in F^c is as follows:

$$v^c = f(s)v^1 + (1-f(s))v^2 \tag{6}$$

Among them, v^1 represents the coordinate of v^c on F^1 , v^2 represents the coordinate of v^c on F^2 , $s = 1 - l/L$. The definition of L is shown in Fig. 6. l is the distance from point to bottom boundary of H^c , and L is the distance between upper and lower boundaries. It can be concluded that the value of s is closer to 1 when the distance between a point and the corresponding lower boundary is smaller. On the other hand, when the distance between a



point and the lower boundary is larger, s is closer to 0.

3 Discussions and result analysis

The algorithm has been implemented on Microsoft Visual Studio platform by using C++ programming. The test computer's CPU (central processing unit) is Intel Core i5-2520M with the basic frequency 2.5 GHz. In addition, the memory is 8 GB and the GPU (graphics processing unit) is Intel HD Graphics 3000.

Figure 7 shows the original model. Figure 7b, c shows the points and triangles of model in Fig. 7a which has 123,801 points and 41,267 triangles. Figure 8 shows the comparison results between the original model and the simplified models. Figure 8a is the original model, with 20,921 points and 41,261 triangle patches. Figure 8b–d use the simplified method preserving the detail features of the model proposed in this paper, retaining the detail features around the eyes and nose. Figure 8b has 3640 points and 7986 triangle patches, and the rate of simplification is 80%. Figure 8c has 1916 points and 3799 triangle patches, and the rate of simplification is 90%. Figure 8d has 1014 points and 1995 triangle patches, and the rate of simplification is 95%. Figure 8e is the figure that does not adopt the simplified method of preserving detail features proposed in this paper, and it has 1243 points and 1905 triangles, and the rate of simplification is 95%. Through Fig. 8b–d, we can see that when the simplification ratio is very low, the detail features of the eyes and nose of the model can be also effectively retained. As can be seen from Fig. 8d, e, although the simplification ratio of the two models is the same, the detail features of the eyes and nose of Fig. 8d are more obvious than Fig. 8e and have higher identifiability.

As shown in Fig. 9, we use our proposed algorithm and the cost-driven algorithm to simplify the original model shown in Fig. 7. Figure 9a is a model after simplification of our algorithm, with 818 points and 1248 triangular patches. Figure 9b is a model after the use of cost-driven simplification, with 1036 points and 1491 triangular patches. Obviously, the boundary of the model obtained by using our algorithm is smoother than that obtained by cost-driven algorithm. Moreover, we can see from Fig. 9 that the simplified algorithm in this paper can retain the detail features of the model more effectively.

Figure 10 is the efficiency comparison when different simplification algorithms are used to simplify the head models of different sizes. Cost-driven method

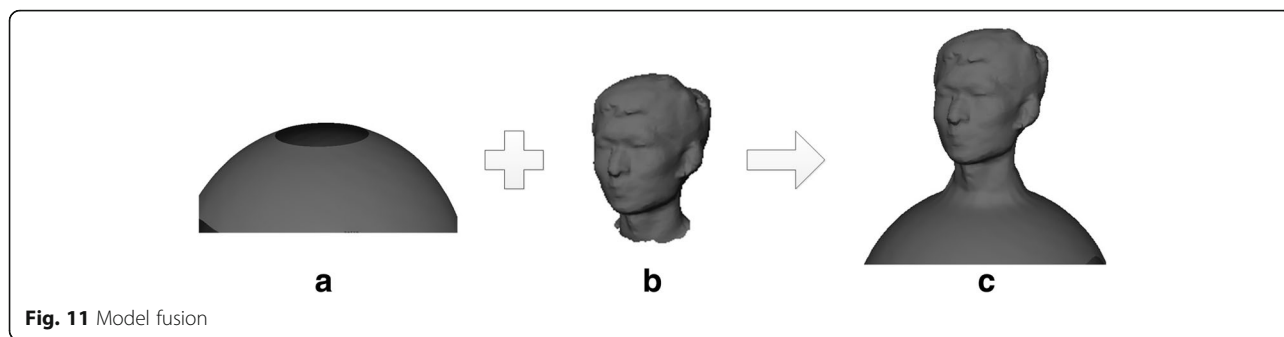


Fig. 11 Model fusion

algorithm refers to the literature [7, 8], and JADE (just another decimator) algorithm refers to the literature [21]. From Fig. 10, we can see that the algorithm efficiency in this paper is better than that of JADE algorithm, and slightly better than that of cost-driven method algorithm. For the head models of different sizes, a simplified algorithm for constructing hierarchical models proposed in this paper is very efficient.

As shown in Fig. 11, this paper uses a semicircle simulated human body with a hole at the top to fuse with a simplified head model, which is shown in Fig. 11c. After fusion, the detail features of the model will not be lost, and the original boundary edge area will be very smooth after fusion. For the fusion method based on the boundary edge, the size of the fusion region of the model is controllable, because the new vertex produced by the fusion has little effect on the complexity of the model and can be ignored. As shown in Fig 11c, there are 3740 points before the fusion of the model, and after fusion, there are 3934 points, which only less than 200 points are generated. The fusion method in this paper will not change the complexity of the model too much and still enables the model to maintain its scale before fusion.

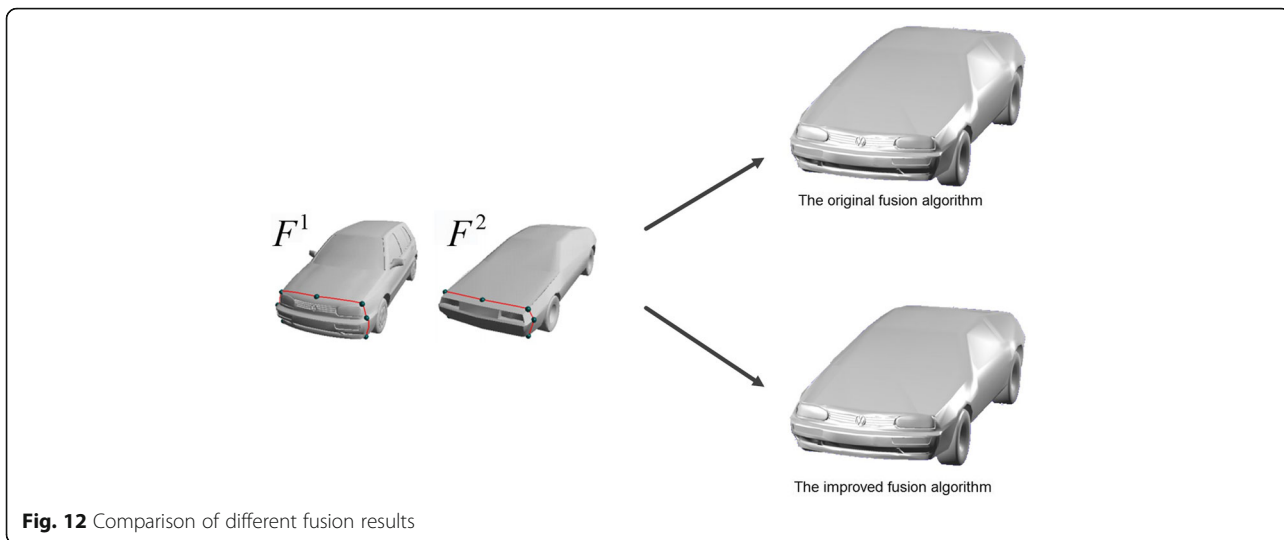
As shown in Table 1, the mesh model fusion method based on FCF is compared with the improved FCF method. When there are 420 and 398 points in F^1 , F^2 , respectively, 2058 points can be generated through fusion by using the method

before the improvement, and 1023 points can be generated through fusion by using the improved method, which is far less than the points produced by using the method before improvement. According to the comparison of data in Table 1, we can see that compared with the original fusion algorithm, the improved fusion algorithm can get a simpler transition model. The effect diagram of fusion experiment in Fig. 12 shows that, although the improved fusion algorithm making the transition model simpler, there is little difference in the effect of fusion.

The hierarchical human body model library can be constructed by this method and applied to crowd simulation evacuation [22–26]. As shown in Fig. 13, the scene of an emergency evacuation of the crowd in a teaching building is simulated. In Fig. 13a), the LOD method is used. When the viewpoint distance is less than 40, a fine human body model (with 43,261 triangular patches) is used, and a medium precision model (with 3216 triangular patches) is used when the viewpoint distance is greater than 40 less than 100, and the rough model (with 1046 triangular patches) is used when the viewpoint distance is greater than 100. The most complex model is used in the Fig. 13b) scene. The average frame rate using the LOD method is 22 FPS (frames per second), while the frame rate using the most complex model is only 7.5 FPS. It can be seen that building hierarchical model library in crowd simulation can effectively improve the performance of the system.

Table 1 Comparison of different fusion algorithms

$F^1(H^1)$		$F^2(H^2)$		$F^c(\text{before improvement})$		$F^c(\text{after improvement})$	
v	f	v	f	v	f	v	f
880	1661	70	85	1836	3567	1184	2008
420	745	398	702	2058	4031	1023	1632
131	246	472	885	1762	3849	748	1316
1561	3645	1758	2244	8327	16,544	3420	6132



4 Conclusions

The 3D human body model is single, and the production process is complex, which is time-consuming and laborious. The Kinect depth camera launched by Microsoft can quickly scan the human body model with human body construction features, so as to reconstruct the 3D model of human body. However, the amount of data of 3D human body model reconstructed with Kinect is large, and the computational complexity is high, which cannot be directly applied to the group simulation and needs to be simplified. At the same time, we need to keep the effective detail features of the model while simplifying it. Therefore, the

main contributions of this paper are as follows: An interactive method for preserving detail features of models based on the simplified methods of edge curvature and area error is proposed. The fusion method based on FCF function is improved, which realizes the automatic fusion between models. The simplified method and improved fusion method proposed in this paper have good robustness for different models. Through the above methods, the personalized character model suitable for group simulation can be reconstructed, and a hierarchical character model database can be constructed, which improves the sense of reality and efficiency of the group simulation.



Abbreviation

3D: Three-dimensional; CGAL: Computational geometry algorithms library; CPU: Central processing unit; DFS: Depth first search; FCF: Fusion control function; FPS: Frames per second; GPU: Graphics processing unit; JADE: Just another decimator; LOD: Levels of detail

Acknowledgements

We gratefully thank the anonymous reviewers and Associate Editor for the constructive and detailed comments that helped improve the paper.

Funding

This work is supported by Shandong Provincial Natural Science Foundation of China under Grant No. ZR2014FQ009.

Availability of data and materials

Please request authors.

Authors' contributions

The original idea of the research was proposed by LM but was largely inspired by the discussions with DL. KZ contributed to the experiment analysis. All three authors worked closely during the preparation and writing of the manuscript. All authors read and approved the final manuscript.

Authors' information

Author 1: Lulu Ma received her Master's degree in accounting from the Shandong Economic University, Jinan, China, in 2006. After that, she joined in the Shandong Normal University, Jinan, China. Currently, she is a lecture of the department of finance at Shandong Normal University. Her research interests include CSCW and CAD.

Author 2: Ke Zhang received his B. S. and M. S. degrees in Shandong Normal University, Jinan, China, in 2013 and 2016, respectively. He is now working at Shandong Pingyin Limited by Share Ltd. of the rural commercial bank. His research interests include CAD and swarm intelligence.

Author 3: Dianjie Lu received his Ph.D. degree in computer science from the Institute of Computing Technology, Chinese Academy of Science, in 2012. Currently, he is an associate professor at Shandong Normal University. His research interests include CSCW, swarm intelligence, and cloud computing. His contact information is ludianjie@sina.com.

Competing interests

The authors declare that they have no competing interests.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Author details

¹Department of Finance, Shandong Normal University, Jinan 250014, China. ²Shandong Pingyin Limited by Share Ltd of the rural commercial bank, Jinan 250400, China. ³School of Information Science and Engineering, Shandong Normal University, Shandong Provincial Key Laboratory for Distributed Computer Software Novel Technology, Jinan 250014, China.

Received: 19 April 2018 Accepted: 25 June 2018

Published online: 10 July 2018

Reference

1. V Konstantinov, V Rozaliev, Y Orlova, A Zabolieva-Zotova, in *Proc. of the First International Scientific Conference on Intelligent Information Technologies for Industry (ITI'16)*. Development of 3D human body model (2016), pp. 143–152
2. I Oikonomidis, N Kyriazis, A Argyros, in *Proc. of British Machine Vision Conference*. Efficient model-based 3D tracking of hand articulations using Kinect (2011), pp. 1–11
3. X Chang, Z Ma, M Lin, Y Yang, AG Hauptmann, Feature interaction augmented sparse learning for fast Kinect motion detection. *IEEE Trans. Image Process.* **26**(8), 3911–3920 (2017)
4. M Kourakli, I Altanis, S Retalis, M Boloudakis, D Zbainos, K Antonopoulou, Towards the improvement of the cognitive, motoric and academic skills of students with special educational needs using Kinect learning games. *Int J Child-Computer Interaction* **11**(1), 28–39, 2017 (2017)

5. J Tong, J Zhou, L Liu, Z Pan, H Yan, Scanning 3d full human bodies using Kinects. *IEEE Trans. Vis. Comput. Graph.* **18**(4), 643–650 (2012)
6. Q Sun, Y Tang, P Hu, J Peng, in *Proc. of International Conference on Image Analysis and Signal Processing*. Kinect-based automatic 3D high-resolution face modeling (2012), pp. 1–4
7. P Lindstrom, G Turk, in *Proc. of IEEE Visualization*. Fast and memory efficient polygonal simplification (1998), pp. 279–286
8. P Lindstrom, G Turk, Evaluation of memoryless simplification. *IEEE Trans. Vis. Comput. Graph.* **5**(2), 98–115 (1999)
9. J Peng, G Zhang, J Li, B Fan, Model simplification based on blend surface reconstruction. *J Chinese Computer Syst* **5**(35), 1169–1173 (2014)
10. X Feng, Q Zhou, An algorithm for simplification of 3D models with texture. *J Computer-Aided Des Computer Graph* **2009**(6), 842–846 (2009)
11. S Zhou, M Chen, An algorithm of vertex normal-based mesh model simplification. *J Image Graphics* **7**(6), 601–605 (2002)
12. H Quan, T Zhang, Y Dong, A geometric model simplification method based on regions partition. *Chinese J Computers* **2006**(10), 1834–1842 (2006)
13. H Zhang, J Sun, Y Lv, N Lv, Y Wang, A new simplification method for terrain model based on divergence function. *Chinese J Computers* **2009**(5), 962–973 (2009)
14. T Ma, G Gong, J Yan, in *Proc. of IEEE 10th International Conference on Industrial Informatics*. A 3D model simplification algorithm based on edge-collapse (Beijing, 2012), pp. 776–779
15. G Sanchez, E Leal, N Leal, A Linear Programming, Approach for 3D point cloud simplification. *IAENG Int. J. Comput. Sci.* **44**(1), 60–67 (2017)
16. H Han, X Han, F Sun, et al., Point cloud simplification with preserved edge based on normal vector. *Optik* **126**(19), 2157–2162 (2015)
17. T Whelan, L Ma, E Bondarev, et al., Incremental and batch planar simplification of dense point cloud maps. *Robotics Autonomous Syst* **69**(C), 3–14 (2015)
18. T Kanai, H Suzuki, J Mitani, F Kimura, in *Proc. of Graphics Interface*. Interactive mesh fusion based on local 3D metamorphosis (1999), pp. 148–156
19. G Li, *Research and Implementation of Kinect Based 3D Reconstruction* (Jiaotong University, Beijing, 2012)
20. M Eck, T Deroose, T Duchamp, H Hoppe, M Lounsbery, W Stuetzle, in *Proc. of ACM Siggraph Computer Graphics*. Multiresolution analysis of arbitrary meshes (1995), pp. 173–182
21. A Ciampalini, P Cignoni, C Montani, R Scopigno, Multiresolution decimation based on global error. *Visual Computer, Springer International* **13**(5), 228–246 (1997)
22. M Cao, G Zhang, M Wang, D Lu, H Liu, A method of emotion contagion for crowd evacuation. *Physica A: Statistical Mechanics and its Applications* **483**(2017), 250–258 (2017)
23. H Liu, B Liu, H Zhang, L Li, X Qin, G Zhang, Crowd evacuation simulation approach based on navigation knowledge and two-layer control mechanism. *Information Sciences*, s **436–437**(2018), 247–267 (2018)
24. H Liu, P Zhang, B Hu, P Moore, A novel approach to task assignment in a cooperative multi-agent design system. *Appl. Intell.* **43**(6), 162–175 (2015)
25. G Zhang, D Lu, H Liu, Strategies to utilize the positive emotional contagion optimally in crowd evacuation. *IEEE Trans. Affect. Comput.* (2018). <https://doi.org/10.1109/TAF.FC.2018.2836462>
26. H Liu, B Xu, D Lu, G Zhang, A path planning approach for crowd evacuation in buildings based on improved artificial bee colony algorithm. *Appl. Soft Comput.* **68**(2018), 360–376 (2018)

Submit your manuscript to a SpringerOpen® journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com