A 3D Model Skeleton Correcting Algorithm using Templates of Inspecting Voxel Disconnection

Xun Jin
Dept. of Copyright Protection,
Sangmyung University
Seoul, Korea
email: jinxun@cclabs.kr

Jongweon Kim
Dept. of Contents and Copyright,
Sangmyung University
Seoul, Korea
email: jwkim@smu.ac.kr

***Abstract*—A 3D model skeleton is an abstraction of a 3D model, which is useful for feature description, 3D model identification and many other applications. However, the skeleton may be disconnected when thinning the 3D model after voxelization. In this paper, a 3D model skeleton correcting algorithm is proposed. It uses 3 pre-defined correcting templates to inspect disconnections and correct them. The templates inspect 26-adjacent voxels of each target voxel after it is removed. After a target voxel is removed, the templates inspect the distances among the rest of 26-adjacent voxels of the target voxel whether any of the distances are greater than or equal to 2. The proposed algorithm is simple and practical. The experimental results show some comparisons between before and after correcting skeletons. The skeleton generated by the proposed method has less noises than that of conventional method and maintains connectivity.**

Keywords-3D model; voxelization; thinning; skeletonization; correcting.

I. INTRODUCTION

In recent years, the development of 3D printing technology has led to the explosive growth of 3D models. Hence, the 3D printing services are increasing rapidly [1][2]. However, copyright infringement of 3D models has become an issue for the 3D printing ecosystem of product distribution websites, design-sharing and 3D scanning [3][4]. To prevent unauthorized use of copyrighted 3D models, the identification of 3D models remains.

Many researchers have proposed various methods for 3D model feature extraction and retrieval [5][6][7][8][9]. Because of the high discriminative property of 3D skeleton-based features for 3D model representation, 3D skeleton-based approaches have attracted much attention [5][6][7]. A 3D model skeleton is not only an abstraction of the model at the center-line of the model but also a fundamental shape feature, which is used for shape description, 3D model identification, and many other applications.

However, the conventional methods of generating skeleton have some weaknesses. In [10], authors used as input a subvoxel precise distance field and employed a number of fast marching method propagations to extract the skeleton at subvoxel precision. The algorithm can't generate the skeleton of each branch of a 3D model. In [11][12], authors used a thinning algorithm to generate skeleton and rechecked eight subvoxels with non-overlapping neighborhoods in parallel. However, the generated skeleton included some noises. In [13], authors modified and improved a fully parallel 3D thinning algorithm described in [14]. The parallel thinning algorithm is based on several pre-defined removing templates (class A, B, C and D). The target voxel will be removed, if the neighborhoods of the voxel match one of the templates. The modified algorithm generates the skeleton of each branch of a 3D model and has very few noises, but the skeleton may be disconnected when thinning the 3D model.

In this paper, we present a 3D model skeleton correcting algorithm using some templates of inspecting voxel disconnection. It uses 3 pre-defined correcting templates to inspect disconnections and correct them. The templates inspect 26-adjacent voxels of each target voxel after it is removed. The templates inspect the distances between the rest of 26-adjacent voxels of the removed voxel, whether they are greater than or equal to 2. The proposed algorithm is simple and practical. Some comparisons between before and after correcting the skeletons are shown in the section of experimental results.

The remainder of this paper is organized as follows. A brief overview of the basic theory of the adjacencies in a 3D binary array is given in Section 2. The methodology of the 3D parallel thinning algorithm is reviewed in Section 3. In Section 4, the proposed 3D skeleton correcting algorithm is introduced. The experimental results of skeletons generated by the proposed method and the other methods are shown in Section 5. Finally, we conclude the proposed method in Section 6.

II. BASIC THEORY

First of all, a voxelization process is performed to a 3D mesh model as in [15]. The 3D space of voxels is transformed to a 3D binary array. Each voxel is denoted by 1 in the 3D binary array. A binary value 0 means that there is no voxel.

Suppose there are two voxels p_1 and p_2 with coordinates (x_1, y_1, z_1) and (x_2, y_2, z_2) in the 3D binary array. The Euclidean distance between p_1 and p_2 is defined as (1) [13].

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2} \qquad (1)$$

If $d = 1$, p_1 and p_2 are 6-adjacent. If $d \leq \sqrt{2}$, p_1 and p_2 are 18-adjacent. If $d \leq \sqrt{3}$, p_1 and p_2 are 26-adjacent. Fig. 1 shows the adjacencies of a voxel p.

 ISBN: 978-1-61208-506-7

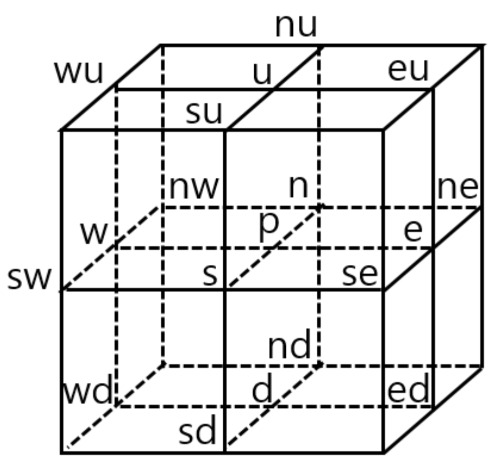


Figure 1. The adjacencies in a 3D binary array.

The n(p), e(p), s(p), w(p), u(p) and d(p) are 6 adjacent voxels of p, where the n, e, s, w, u and d denote north, east, south, west, up and down. The 18-adjacent voxels of p (not include 6-adjacent voxels) are nu(p), ne(p), nd(p), nw(p), eu(p), ed(p), su(p), se(p), sd(p), sw(p), wu(p) and wd(p), and so on.

III. 3D Parallel Thinning Algorithm

The parallel thinning algorithm is based on several pre-defined removing templates (class A, B, C and D) [13][14]. The target voxel will be removed, if the voxel is a non-tail voxel and the neighborhoods of the voxel match one of the templates.

Fig. 2, 3, 4 and 5 show the templates of four classes. There are 6 templates in class A, 12 templates in class B, 8 templates in class C and 36 templates in class D. A solid black circle denotes a binary value 1, a white circle denotes a binary value 0. The voxels in the unmarked positions are "don't care" voxels, which means the binary values can be either 1 or 0. The white squares mean that at least one of them is 1. The two white triangles in Fig. 5 mean that both of them can't be 1 simultaneously, which means at most one of them is 1 or both of them are 0.

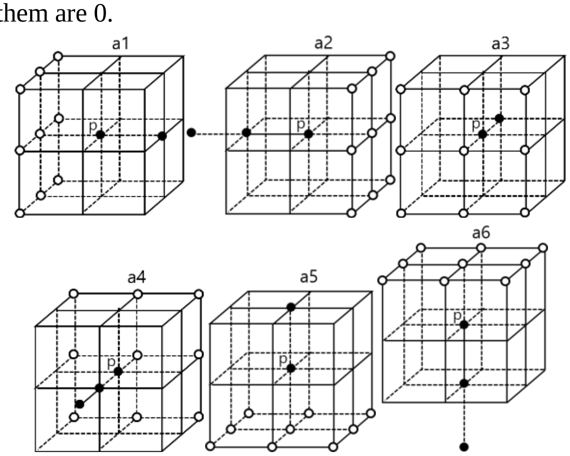


Figure 2. 6 removing templates in class A.

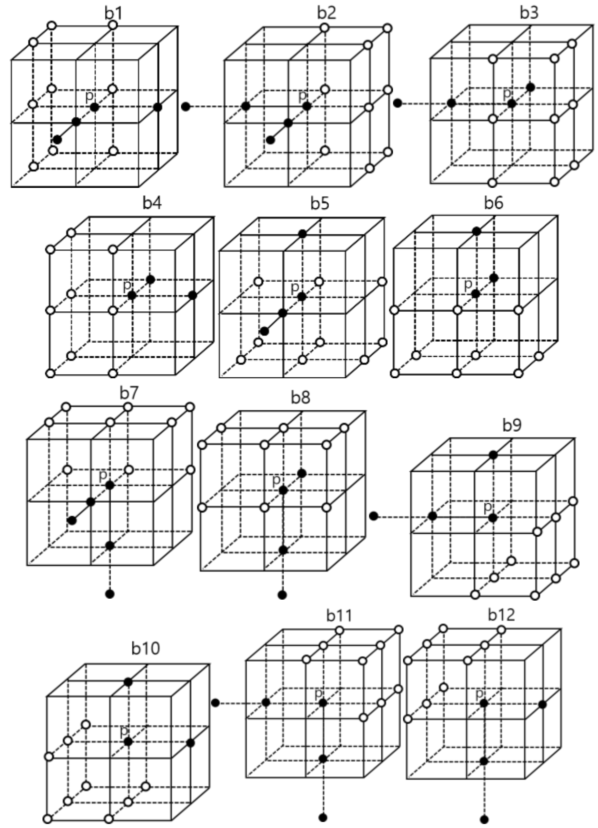


Figure 3. 12 removing templates in class B.

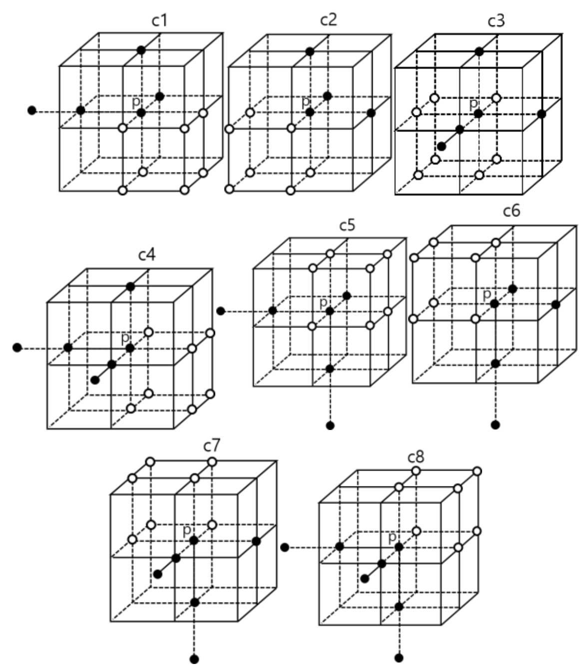
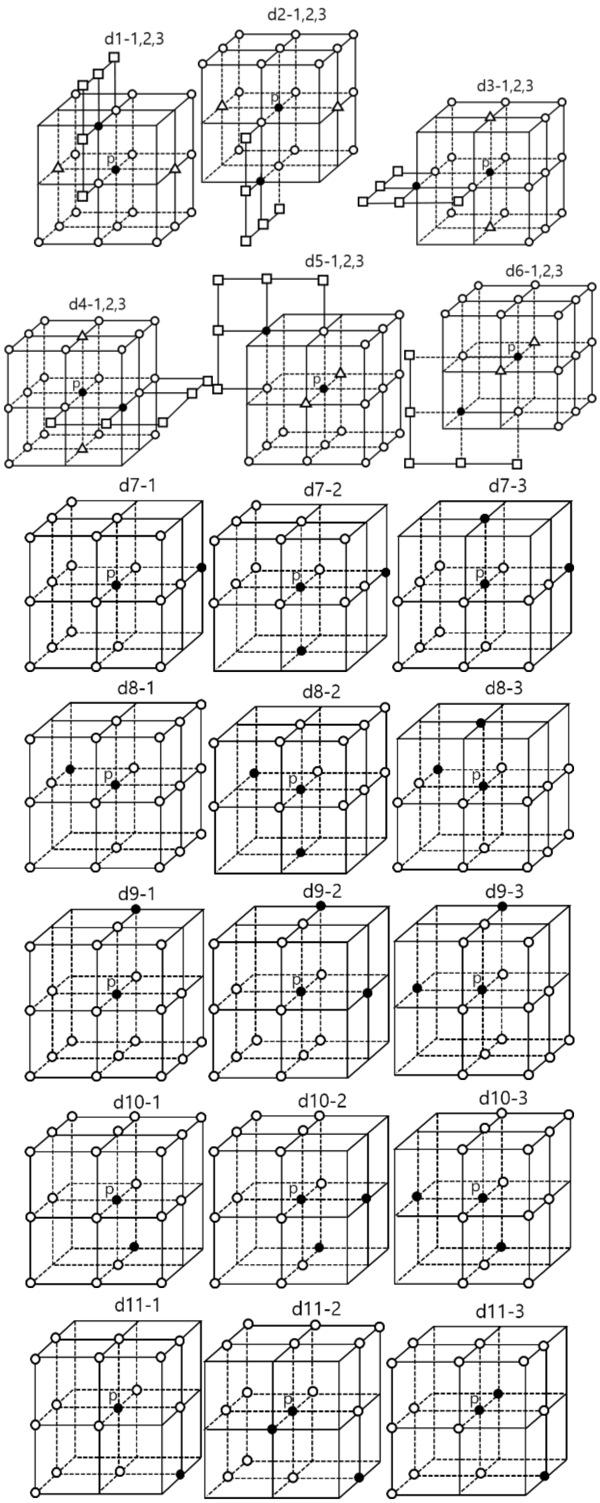


Figure 4. 8 removing templates in class C.



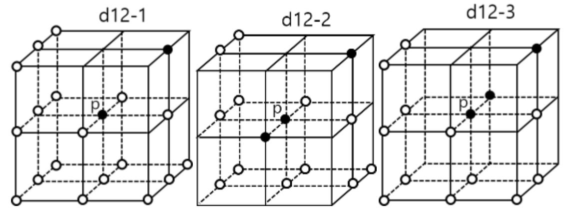


Figure 5. 36 removing templates in class D.

If a voxel p is 26-adjacent to only one voxel, it is called a line-end voxel. If the p is 26-adjacent to only two voxels, it is called a near-line-end voxel. If the p is either a line-end or a near-line-end voxel, it is called a tail voxel, otherwise it is called a non-tail voxel. Analyze each voxel of a 3D model whether it is a non-tail voxel and satisfies at least one of the removing templates of any classes. After analyzing and marking the whole voxels, the voxels which satisfy both conditions will be removed simultaneously. Repeat the process until no voxel can be removed. The authors in [13] found that the 12 conventional templates of class D in [14] may disconnect the skeleton. Therefore, they modified and expanded each template of class D to 3 templates. Each of the templates from d1 to d6 in Fig. 5 is expanded to 3 templates according to the binary values of the two triangles. Thus, there are 36 templates in class D.

IV. THE PROPOSED 3D SKELETON CORRECTING ALGORITHM

A 3D model thinning algorithm should keep the generated skeleton connected. However, we found that the modified algorithm failed to preserve connectivity. Fig. 6 shows an example about how the algorithm disconnects the skeleton. The voxel p1 is a non-tail voxel, because it is 26-adjacent to 6 voxels: p2, p3, q1, q2, q3 and q4. It also satisfies the template d7-2 in class D. Thus, it will be removed. The voxel p2 is a non-tail voxel, because it is 26-adjacent to 5 voxels: p1, p3, q2, q3 and q4. It also satisfies the template a5 in class A. Therefore, it will be also removed. The voxel p3 is a non-tail voxel, because it is 26-adjacent to 4 voxels: p1, p2, q2 and q3. It also satisfies the template d9-3 in class D. Therefore, it will be also removed. When the voxels p1, p2 and p3 are removed, the rest of the voxels are disconnected.

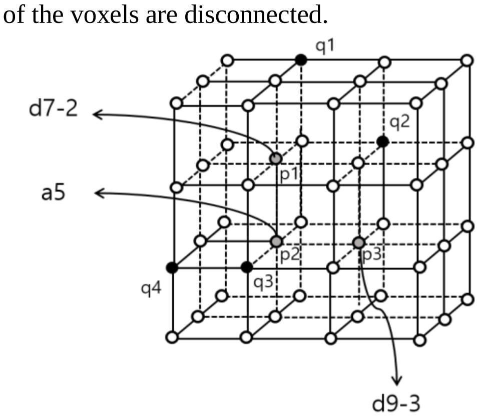


Figure 6. An example of disconnection.

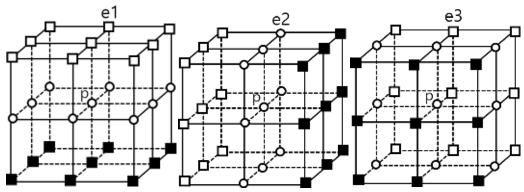


Figure 7. Correcting templates of class E.

To solve this problem, we present a 3D skeleton correcting algorithm with 3 templates of inspecting voxel disconnection. These templates are assigned to class E as shown in Fig. 7. The black squares mean that at least one of them is 1. The templates inspect the connectivity of the voxels by checking the distance between the voxels, whether it is greater than or equal to 2. Thus, after the target voxel p is removed, to satisfy the templates, there are at least two voxels left, one of them is black square and the other one is white square. If the 26-adjacent voxels of the removed voxel satisfy the templates of class E, the removed voxel p can be recovered. With the templates in class E, we can inspect that the voxel q2 and q3 satisfy the template e3 after the last voxel of p1, p2 and p3 is removed. Thus, we can recover the last removed voxel to correct the disconnection of the skeleton and preserve connectivity.

V. EXPERIMENTAL RESULTS

In this section, we show some experimental results of comparisons between before and after correcting the skeletons with Matlab. The proposed algorithm is compared with 3 other skeletonization algorithms. First one is described in [10], which used fast marching methods (FMM). Second one is proposed in [11][12], which built skeleton via 3D medial surface and axis (MSA). Last one is proposed in [13][14], which used a fully parallel thinning algorithm (FPT). The performances of the algorithms are evaluated with two 3D models in SHREC 2015 benchmark. Fig. 8 shows the two 3D models: bird and armadillo.

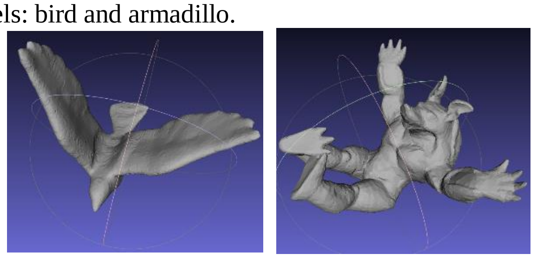
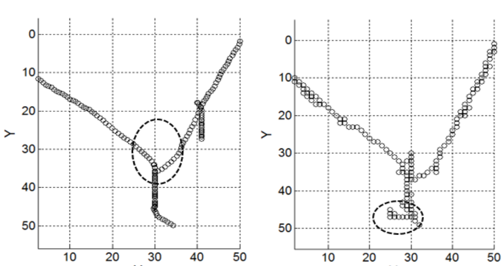
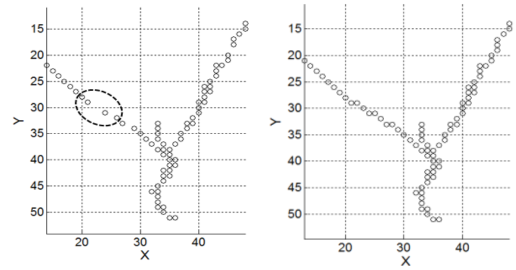


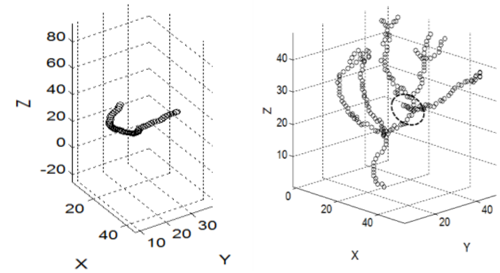
Figure 8. Bird and armadillo models.



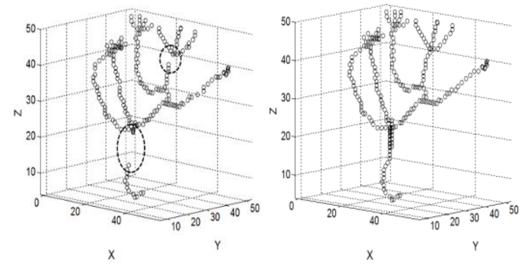
(a) (b)



(c) (d)



(e) (f)



(g) (h)

Figure 9. Skeletons generated by the four algorithms.

The skeletons generated by the four algorithms are shown in Fig. 9. Fig.9 (a) and (e) show the skeletons generated by FMM. The skeletons of the bird's head and the armadillo's two arms and one leg are not generated. Fig. 9 (b) and (f) show the skeletons generated by MSA. The skeletons of the bird's tail and armadillo's chest contain many noises. Fig. 9 (c) and (g) show the skeletons generated by FPT. The skeletons of bird's left wing and armadillo's tail and head are disconnected. Fig. 9 (d) and (h) show the skeletons generated by the proposed correcting algorithm. The skeleton of the bird's head is generated and that of bird's tail doesn't have many noises. The connectivity of the left wing is also preserved. The skeletons of armadillo's arms and legs are generated. They don't have many noises and maintain connectivity.

VI. CONCLUSION AND FUTURE WORK

In this paper, we present a 3D model skeleton correcting algorithm using templates of inspecting voxel disconnection. Three pre-defined correcting templates are used to inspect disconnections and recover them. The templates inspect the connectivity of 26-adjacent voxels of each removed voxel. The proposed algorithm is simple and practical. The experimental results show that the proposed algorithm can repair the disconnection and provide corrected skeletons. In the future work, we will apply the algorithm to the identification of 3D models.

 ISBN: 978-1-61208-506-7

ACKNOWLEDGMENT

This research project was supported by the Ministry of Science, ICT and Future Planning in 2015.

REFERENCES

[1] F. R. Ishengoma and A. B. Mtaho, “3D Printing Developing Countries Perspectives,” International Journal of Computer Applications, vol. 104, pp. 30–34, 2014.

[2] A. Harris, “The Effects of In-home 3D Printing on Product Liability Law,” Journal of Science Policy and Governance, vol. 6, pp. 1-11, 2015.

[3] D. Gupta and M. Tarlock, “3D Printing, Copyright Challenges, and the DMCA,” New Matter, vol. 38, pp. 1-16, 2013.

[4] S. H. Lee et al., “Watermarking scheme for copyright protection of 3d animated model,” IEEE Consumer Communications and Networking Conference (CCNC), Las Vegas, NV, USA, pp. 1-4, 2012.

[5] R. Shen, T. Wang, L. Shi, X. D. Yang, and I. Cheng, “3D Model Retrieval Using Semantically Rich Skeleton: A Review,” IEEE MMTC E-letter, vol. 6, pp. 22-26, 2011.

[6] N. D. Cornea et al , “3D Object Retrieval using Many-to-many Matching of Curve Skeletons,” International Conference on Shape Modeling and Applications, Washington, DC, USA, pp. 368-373, 2005.

[7] H. Lei et al , “A novel sketch-based 3 model retrieval method by integrating skeleton graph and contour feature,” Journal of Advanced Mechanical Design, Systems, and Manufacturing, vol. 9, pp. 1-16, 2015.

[8] S. Jain and S. Mishra, “Survey Paper on Various 3D View Based Retrieval Methods,” International Journal of Engineering Research and Technology, vol. 3, pp. 470-473, 2014.

[9] L. Y. Jie, B. Feng, L. Z. Min, and L. Hua, “3D Model Retrieval Based on 3D Fractional Fourier Transform,” The International Arab Journal of Information Technology, vol. 10, pp. 421-427, 2013.

[10] R. V. Uitert and I. Bitter, “Subvoxel precise skeletons of volumetric data based on fast marching methods,” Medical Physics, vol. 34, pp. 627-638, 2007,.

[11] T. C. Lee, R. L. Kashyap, and C. N. Chu, “Building Skeleton Models via 3-D Medial Surface/Axis Thinning Algorithms,” Graphical Models and Image Processing, vol. 56, pp. 462-478, 1994.

[12] M. Kerschnitzki et al , “Architecture of the osteocyte network correlates with bone material quality,” Journal of bone and mineral research, vol. 28, pp. 1837-1845, 2013.

[13] T. Wang and A. Basu, “A note on ‘A fully parallel 3D thinning algorithm and its applications’,” Pattern Recognition Letters, vol. 28, pp. 501-506, 2007.

[14] C. M. Ma and M. Sonka, “A Fully Parallel 3D Thinning Algorithm and Its Applications,” Computer Vision and Image Understanding, vol. 64, pp. 420-433, 1996.

[15] S. Patil and B. Ravi, “Voxel-based representation, display and thickness analysis of intricate shapes,” International Conference on CAD and CG, Hong Kong, pp. 415-422, 2005.