International Conference on Advanced Technologies, Computer Engineering and Science (ICATCES2019), Apr 26-28, 2019 Alanya, Turkey

Reassembly of Synthetically Fractured Objects

Ö. ÇAKIR[1] and V. NABİYEV[1]

[1] Karadeniz Technical University, Trabzon/Turkey, cakiro@ktu.edu.tr
[1] Karadeniz Technical University, Trabzon/Turkey, vasif@ktu.edu.tr

Abstract **– In this paper we present a reassembly system for synthetically fractured 3D objects. Given as input synthetically fragmented parts of the tetrahedralized 3D vase model, for segmentation and matching stages we classify faces into two divisions whether they belong to the original model or fracture surface using indices of shared triangles as background knowledge. Then the vase model is reconstructed using an alignment algorithm based on unit sphere coordinates approach. The system is evaluated on several synthetically fracturing scenarios. The performance of the proposed system indicates that all reassembly computations are done in real-time.**

Keywords **– reassembly, fractured objects, 3d puzzle.**

I. INTRODUCTION

THE reassembly of fractured 3D objects has gained an increasing importance in criminal policy, anthropology and especially in the field of archeology. It requires the solutions to two sub-problems, namely, matching and alignment to determine the shared faces between fragments and compute the inverse transformation matrices of the fragments prior to reassembly, respectively.

In this study a vase model is fractured synthetically. Triangles are used as the rendering primitive for fracture surfaces, so a volumetric triangulation method is needed for 3D points belongs to fracture surfaces. To this purpose, we use TETGEN [1] which generates not only tetrahedrons but also inner and outer triangles. As fracturing the vase model, the shared fracture surface of fragments is constructed using inner nearest triangles along the 3D fracturing path and duplicated for each fragment.

When fracturing is synthetical, the matching is simpler than the alignment because fracture surfaces are duplicated in each fragment. To classify a face as original versus fractured, we use the background knowledge of indices of shared triangles. For alignment, we use normal vectors of the fracture surfaces to compute the matrices of inverse rotation and translation. Each normal vector is matched with unit spherical *(phi, theta)* coordinates with respect to the reference vector. Details are discussed in section III.

In this paper, we present a reassembly system that automatically matches and aligns synthetically fractured 3D objects. We select a vase model and evaluate our reassembly system on several fracturing scenarios successfully. Performance results show that all reassembly computations are done in real-time.

II. RELATED WORK

The literature of the field of the reassembly of fractured 3D objects mostly focuses on the reconstruction of the broken archeological objects. To the best of our knowledge, the first work in reconstruction of the 3D objects is conducted by Papaioannou et al. [2] where fracture surfaces are assumed to be nearly planar and match each other one-to-one. The work in [2] use region-growing algorithm for surface segmentation. Then vertices of each face are projected in direction of the average face normal and the resulting depth maps are used for matching stage and extends matching method with boundary curve matching approach in [3].

Huang et al. [4] extract edges from point cloud data and faces are extracted from cycles of these edges. Original faces versus fracture faces segmentation are performed using graph cut algorithm runs on all faces of each fragment. For pairwise matching between two fractured surfaces they use patch-based surface features.

Winkelbach and Wahl [5] introduce a hierarchical clustering algorithm that decomposes each point set into a binary tree structure. The pairwise fragment matching is performed by descending the cluster trees simultaneously using depth-first search.

Altantsetseg et al. [6] proposed pairwise matching of 3D fragments using Fast Fourier Transform (FFT). They introduce two new descriptors: (a) the cluster of feature points extracted by using the curvature values of points and (b) curves along the principal directions of the cluster that are computed using Fourier series. They compare descriptor curves in terms of Fourier coefficients computed by FFT and total energies of curves.

Various other matching and alignment descriptors have been proposed such as curvatures [7], integral invariants [8] and RANSAC algorithms [9].

Recently, Son et al. [10] proposed reassembly of fractured objects using surface signature descriptor. The descriptor is based on whether a point on the fractured surface is convex or concave. The similarity between two fracture surfaces is calculated based on the spin images of feature curve points. For matching distance and normal deviation between two feature curves are used.

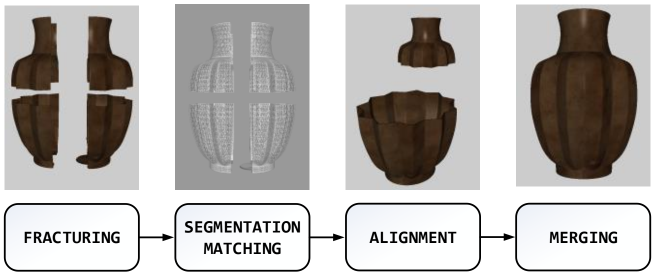

Figure 1: Overview of the reassembly system.

III. Proposed Reassembly System

Our reassembly system is composed of Fracturing, Segmentation, Matching, Alignment and Merging stages as shown in Figure 1.

A. System Inputs

Main input vase is a MAYA model that consists of 20494 vertices and 148692 triangles. Reassembly system requires various other inputs to be generated from this model:

1) *Shading Components*: Vase model is exported as *.OBJ file. Shading components like vertex, normal and texture coordinates are obtained from this *.OBJ file.

2) *Fragments*: Each fragment consists of original and fracture surfaces and each surface is defined as an array of indices of triangles.

3) *Inner Triangles*: Inner triangles are needed for the fracture surfaces. The model is exported as *.STL file and inner triangles are generated using TETGEN. Details are discussed in the following section.

B. Reassembly System

The fracturing is the first stage of the reassembly system where synthetical fracturing of the 3D vase model takes place. The vase is fractured using different planar fracturing scenarios including horizontal, vertical and both as shown in Figure 2. The system supports any planar fracturing. Vertices are split between fragments under regarding planar fracturing constraints.

We prefer triangles as a rendering primitive for fracture surfaces. To this purpose we use TETGEN which is a program to generate tetrahedral meshes. Besides tetrahedrons TETGEN also generates inner and outer triangles. We use inner triangles for fracture surfaces.

While fracturing, some vertices of the fracture surface are assigned to one fragment and some to other depending on regarding planar constraint. Please note that, inner triangles that have at least one vertex of two neighboring fragments are selected and duplicated in each fragment to generate fracture faces.

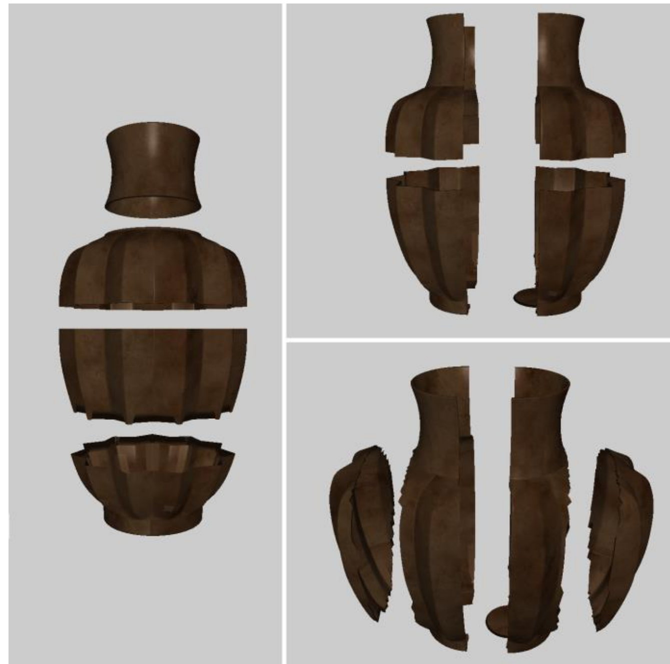

Figure 2: Various fracturing scenarios

In the stage of segmentation, every face is classified into two divisions whether it belongs to the original model or fracture surface using the background knowledge of indices of shared triangles because these indices are duplicated in fracturing stage. Background knowledge is also used for matching.

In the alignment stage, we use normal vectors of the fracture surfaces to compute the matrices of inverse rotation and translation. Each normal vector is matched with unit spherical *(phi, theta)* coordinates with respect to the *reference vector* (1,0,0). To this purpose we generate a unit sphere that consists of 64440 points (*phi* varies in [0,359], *theta* varies in [1,179]) as can be seen in Figure 3.

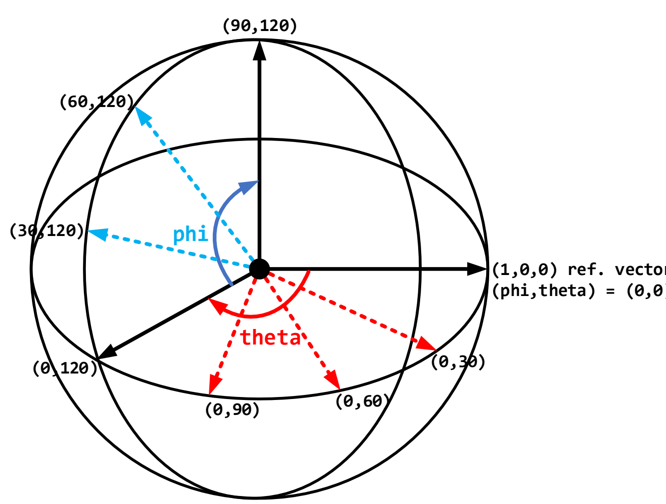

Figure 3: Unit sphere approach. Click sphere to watch animation.

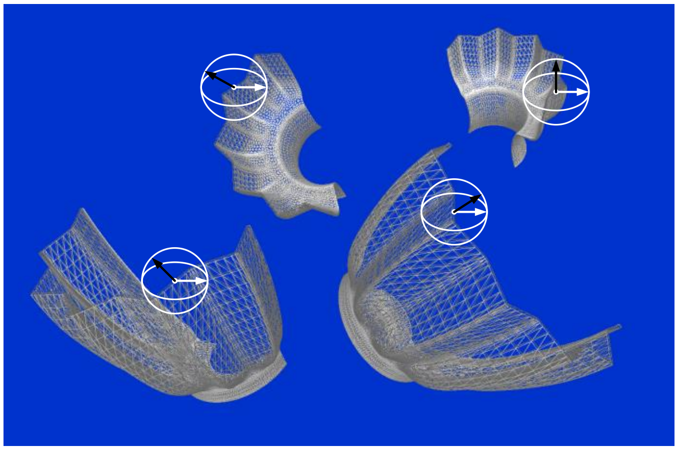

Figure 4: Spherical coordinates of the shared normal vectors.

Our reassembly system is implemented in DirectX 12. Interactive demo application runs on GeForce GTX 1050Ti GPU in real-time.

In demo application the vase model is firstly fragmented depending on regarding planar constraints. Then each fragment is rotated and moved to the arbitrary point in 3D space using keyboard. Using indices of shared triangles, the segmentation and matching tasks is performed. Then reassembly procedure executes and computes unit spherical coordinates of the shared normal vectors for each fragment as illustrated in Figure 4. Using spherical coordinates, the matrices of inverse rotation are calculated. Fragments are aligned using inverse rotation matrices and merged by translation.

Sample frames from the demo application are shown in Figure 5. In addition, a screen captured movie of the demo can be seen at: http://ceng2.ktu.edu.tr/~cakir/icatces2019.html The FPS values displayed in the movie, regarding the performance of the proposed system indicates that all computations are done in real-time.

Figure 5: Sample frames from demo application.

IV. Conclusion

In this paper, a reassembly system that automatically matches and aligns synthetically fractured 3D objects is presented and its performance is evaluated on several fracturing scenarios successfully in real-time.

As a future work, a mechanism for reconstruction of real objects like 3D scanned archaeological remains will be developed. Since there are some erosions on fracture surfaces of archaeological remains, the reassembly processes including segmentation, matching and alignment becomes more difficult than presented system in this paper. We aim to propose a novel matching and alignment descriptor for lossy fragmented objects in the future.

References

[1] http://wias-berlin.de/software/index.jsp?id=TetGen&lang=1

[2] G. Papaioannou, E. Karabassi, and T. Theoharis, “Virtual archaeologist: Assembling the past”, IEEE Computer Graphics and Applications 21, 2, 53–59, 2001.

[3] G. Papaioannou, E. Karabassi, “On the automatic assemblage of arbitrary broken solid artefacts”, Image and Vision Computing 21, 401–412, 2003.

[4] Q. X. Huang, S. Flry, N. Gelfand, M. Hofer, H. Pottmann, “Reassembling fractured objects by geometric matching”. ACM Trans. Graph. (TOG) 25(3), 569–578, 2006.

[5] S. Winkelbach, F. M. Wahl, “Pairwise matching of 3D fragments using cluster trees”. Int. J. Comput. Vis. 78(1), 1–13, 2008.

[6] E. Altantsetseg, K. Matsuyama, K. Konno, “Pairwise matching of 3D fragments using fast Fourier transform”. Vis. Comput. 30(6), 929–938, 2014.

[7] X Li, and I. Guskov, “Multiscale features for approximate alignment of point-based surfaces”, In SGP’05, 217–226, 2005.

[8] N. Gelfand, N. J. Mitra, L. J. Guibas, and H. Pottmann, “Robust global registration”, In SGP’05, 197–206, 2005.

[9] M. A. Fischler, R. C. Bolles, “Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography”. Commun. ACM 24(6), 381–395, 1981.

[10] T. Son, J. Lee, J. Lim, K. Lee, “Reassembly of fractured objects using surface signature”, Vis Comput. 34:1371–1381, 2018.