

Real-Time Polygon Reconstruction for Digital archives of Cultural Properties

Megumi Okumoto¹, Yuri Iwakata², Asuka Komeda², Tomoko Yonezawa³, Masami Takata⁴, and Kazuki Joe⁴

¹Graduate School of Humanities and Sciences, Nara Women’s University, Nara, Nara, Japan

²Department of Science, Nara women’s University, Nara, Nara, Japan

³Faculty of Informatics, Kansai University, Takatsuki, Osaka, Japan

⁴Academic Group of Information and Computer Sciences, Nara women’s University, Nara, Nara, Japan

1 Introduction

To display special purpose 3D objects on Virtual Reality (VR) systems, when they can be analyzed intuitively and easily, it is possible for those 3D objects to be applied to cultural property research. It is sometimes difficult for users who are not familiar with VR to control objects displayed in VR according to their intention. Therefore, to control VR objects for such users, operations of VR systems should be intuitive and very easy. Motions are one of intuitive human’s operations. Kinect[1] is a device for capturing human’s motions. In this paper, we develop a VR system using Kinect as a VR interface device.

To faithfully describe the detail of a cultural property, its digital archive measured by a 3D scanner becomes extremely complex and huge, and the number of meshes of the polygon model is too many to be efficiently displayed in VR systems. As the number of the meshes increases, the operability of VR systems deteriorates. To use digital archives of cultural properties for research, which are displayed in VR, it is required that the number of meshes is changeable on demand from users. Consequently, in this paper, we propose a real-time polygon reconstruction method. To apply this method to a VR system with Kinect, the VR system is intuitive and easy for any users.

2 A Virtual Reality System using Kinect

In this paper, we develop a VR system using Kinect as a VR interface device. Figure 1 shows a conceptual image of our developing VR system with Kinect. The flow of the VR system operations using Kinect is shown as follows:

Step A-1 Display objects on VR.

Step A-2 Capture user’s motion by Kinect.

Step A-3 Update the objects as control requests from the captured motions.

Step A-4 Return to Step A-1.

Figure 2 (a), (b) and (c) show the three patterns. Table 1 shows the corresponding resolution for captured each motion. In this way, these motions are applied to the target object so that three kinds of resolution for the object are selectable.

3 Real-Time Polygon Reconstruction

In this paper, we propose a real-time polygon reconstruction method for digital archives of cultural properties.

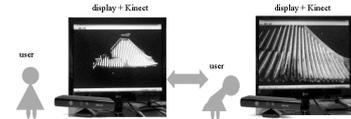


Fig. 1. Conceptual image of a VR system using Kinect.



(a) Upstanding (b) Bowing (c) Deep bowing

Fig. 2. Motions to be captured.

Table 1. Captured motions, object resolutions and polygon resolutions.

captured motions	resolutions	
	object	polygon
upstanding	normal	normal
bowing	scale-up	detail
deep bowing	more scale-up	more detail

Polygon reconstruction needs to be executed in real-time so that user feels comfortable to operate the VR objects. In [3], the meaning of real-time for interactive systems is defined as less execution time than 1.0 second that is the time user can wait without so much stress.

The flow of real-time polygon reconstruction is shown as following steps B.

Step B-1 By using QSlim, create a history of polygon reductions which are to be used for the polygon reconstruction.

Step B-2 Display a 3D object with the current polygon model on VR.

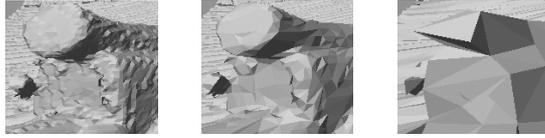
Step B-3 Capture user’s motion by Kinect.

Step B-4 According to the captured user’s motion, change the polygon model to another model from the polygon reduction history.

Step B-5 Return to Step B-2.

Table 2. Execution time for polygon reduction [sec.].

	Setup	Init	Run	Output	Total
Reducing-1	135.2	29.5	31.3	11.0	206.9
Reducing-2	67.3	14.9	23.7	9.6	115.5



(a) 1,653,199 (b) 330,637 (c) 33,064

Fig. 3. Polygon models for the experiment.

Changing polygon models can be quickly done with the polygon reduction history, where QSlim is applied to each polygon reduction. QSlim is an application for polygon reduction based on QEM[2]. The change of polygon models in the proposed system works at the same time with scaling, which is described in section 2.

Table 1 shows the relationship between the inputted user’s motion and the number of polygons.

4 Experiments

In this section, we explain the results of two experiments for validating the proposed real-time polygon reconstruction method.

4.1 Polygon data for the experiment

For the experiments, we use a part of 3D coordinate measuring data of the five-story small pagoda (national treasure), which is enshrined in Kairyuouji temple, Nara, Japan. The data was measured by Nakagawa’s laboratory of Waseda University, Ikeuchi’s laboratory of the University of Tokyo and our research group in 2011.

4.2 Computation time for polygon reduction

As the first experiment, the calculation time of polygon reduction is measured. In polygon reduction, the number of meshes for a polygon model is reduced in two steps. In the first step, the number of meshes is reduced from 1,655,199 meshes to 330,637 meshes. In the second step, the number of meshes is reduced from 330,637 meshes to 33,064 meshes. We call these steps Reduce-1 and Reduce-2, respectively.

Figure 3 shows the three polygon models. When the number of meshes for a polygon model is small, the polygon model is coarse. Table 2 shows calculation time for Reduce-1 and Reduce-2, where Setup, Init, Run and Output represent file loading, command line parsing, pre-processing, polygon reducing and history file writing times, respectively.

As the result of the first experiment, since polygon reduction takes long time, it is unsuitable to execute polygon reduction during changing polygon models. Therefore, in the proposed system, polygons should be restructured with a history of polygon reduction.



(a) Upstanding (b) Bowing (c) Deep bowing

Fig. 4. The scenery of the second experiment.

Table 3. Required Time for Polygon Reduction [sec.].

Setup	change			
	1 → 2	2 → 3	3 → 2	2 → 1
181.703	0.813	0.781	0.782	0.765

4.3 Time for the proposed polygon reconstruction method

In the second experiment, the elapsed times for file read and polygon model change in the developed polygon reconstruction method is measured. Figure 4 shows the scenery of the second experiment. It is observed that the resolution of each polygon model is changed according to the motions. Additionally, examinee can intuitively operate the developed system by using Kinect. Therefore, we conclude that it is easy for user to operate this system without any pre-training for the operation.

Table 3 shows various times in the second experiment, where Setup and Change indicate file loading and polygon model change for given examinee’s motions, respectively. As Tab.3 shows, any changes of meshes require less than 1.0 second. Since the definition of real-time in this paper is 1.0 second, the response time for the developed system is sufficient.

5 Conclusions

In this paper, we proposed a polygon reconstruction method for digital archives of cultural properties. To use digital archives of cultural properties in VR for research purpose, it is required that mesh resolution is changeable on the demand of users. Additionally, polygon reconstruction needs to be run in real-time so that users feel comfortable for their demands. In this paper, we proposed a real-time polygon reconstruction method using history of preliminary polygon reductions. To validate the proposed system, the polygon reconstruction method is evaluated. As the result of the experiment, we confirmed that the polygon reconstruction is performed within 1.0 second, which is considered as "real-time" by the definition of typical user-interface.

Our future work includes the implementation of view-dependent refinement. By implementing view-dependent refinement, polygon models can be displayed more rapidly.

References

- [1] Xbox.com, <http://www.xbox.com/en-US/Kinect>
- [2] Michael Garland, Paul S. Heckbert, Surface simplification using quadric error metrics, SIGGRAPH '97 Proceedings, New York, NY, USA, pp.209-216, 1997
- [3] Jakob Nielsen, Kara Pernice, Eyetracking Web Usability, New Riders Press, 2009