Proc. of the 4th International Conference on Intelligent Interactive Multimedia Systems and Services (KES IIMSS 2011), University of Piraeus, Greece, Jul. 20-22, 2011, published in Intelligent Interactive Multimedia Systems and Services, Smart Innovation, Systems and Technologies, 2011, vol. 11, pp. 153-162, DOI: 10.1007/978-3-642-22158-3_16

Frame Extraction Based on Displacement Amount for Automatic Comic Generation from Metaverse Museum Visit Log

Ruck Thawonmas and Akira Fukumoto

Abstract The paper describes a system for automatically generating comics from visit log at a metaverse museum. Metaverse is a 3D virtual world in which users can act freely, such as visiting museums or chatting with others, according to their own purposes. Compared with existing approaches for representing user experiences using snapshots or video clips, the comic approach can 1) allow users to grasp the whole story at a glance, 2) facilitate distinguishing of important frames, and 3) exploit varieties of comic writing techniques. In order to summarize user experience into comic's frames, detection of important experience, interesting exhibits in case of the museums, is an important task. In this paper, we propose a method for frame extraction based on the displacement amount of a user of interest at a metaverse museum. After describing each module of our system, we discuss a user evaluation in which the effectiveness of the proposed frame extraction method is confirmed when it is compared with a typical baseline method.

1 Introduction

Virtual 3D space called metaverse has recently gained interests from not only educators but also researchers as a promising educational and research platform. Second Life (SL) is representative metaverse equipped with a function that enables its users to build objects or architectures. Unlike online games, metaverse, in general, has no specific roles assigned to the users. As a result, user experiences are arguably limitless including, for example, visiting to museums built by other users or chatting to other users. Other usages

Intelligent Computer Entertainment Laboratory
Graduate School of Science and Engineering, Ritsumeikan University
Kusatsu, Shiga, 525-8577, Japan
e-mail: ruck@ci.ritsumei.ac.jp

of metaverse include an experiment (Prendinger 2009) that aims at realization of an eco-friendly society as well as Machinima (Lowood 2006) that uses in-game avatars (user characters), objects, and architectures for filming.

A number of metaverse systems have functions that allow users to take snapshots of their experiences or record them into video clips. Such snapshots or video clips are used by the users to recall their memories or shown to other users via, for example, blogs or SNS sites. However, manually taking a snapshot each time imposes a burden to the user. And video clips require considerable time to grasp the story and distinguish between important and unimportant events therein. In order to cope with these issues, we focus on the use of comics for representing user experiences.

Using the comic style, one can summarize a given user experience into a limited number of frames, provided that a proper summarization mechanism is used. Hence a whole story can be apprehended at one glance. In addition, a frame layout technique can be applied to emphasize important frames and underemphasize unimportant ones. In this paper, we focus on museum visiting, one of the major usages in metaverse, present a system for generating a comic based on the museum visiting experience of a user of interest, and propose a frame extraction method that uses the information on the amount of displacement at a museum of interest.

2 Comic Generation System

We have developed a comic generating system and related methods (Shuda and Thawonmas 2008; Thawonmas and Shuda 2008; Thawonmas and Oda 2010) for on-line games. Our objectives therein are to summarize players' game experiences into comics so that the players can exchange their comics with others and recall their memories about the game. However, an online game, the research target in our previous work, is essentially different from metaverse, which is the research target of this paper.

In a typical online game, players' actions, such as "attack a monster" or "open a treasure" are prepared and provided in advance by the game developer. Such actions are meaningful, can be interpreted as important episodes, and are thus worth memorizing. However, most of the actions in metaverse, such as "move" or "sit" have not much meaning. It is therefore difficult to extract important episodes from log. As far as the museum visit application is concerned, one might come up with a method for extracting comic frames that cover important exhibits specified in advance by the museum's developers. However, this method lacks the universality because it cannot be used at other museums where such information is not available. In addition, the method does not take into account the preference of a user of interest.

According to a study on a real-world museum (Sparacino 2003), the more interest a user has in a given exhibit, the higher probability is that he or

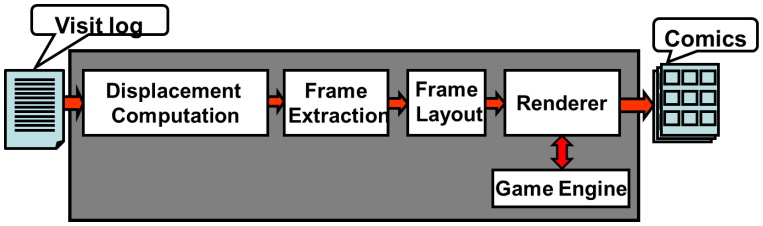

Fig. 1 Architecture of the comic generation system

she will spend longer time viewing the exhibit. Based on this finding, we propose a frame extraction method that bases on the displacement amount (cf. (1)) to predict the user's interest at a given virtual museum. Namely, the displacement amount of zero implies that the user is stopping at a particular place viewing an exhibit, and the longer stop period, the more interest the user has in the exhibit. The proposed frame extraction method puts higher priorities to such stops.

Figure 1 shows the architecture of the comic generation system, where "visit log" indicates a file that contains information on the user traces (the coordinate (x, y, z) at each loop-count, the number of times in sampling the coordinate), the user actions, and the object positions in the museum. In the following, we describe each of the four modules shown in Fig. 1.

1. At this module, the amount of displacement for a given user trace at each loop-count l is calculated as follows:

$$\text{displacement}^2 = \big(x(l)-x(l-1)\big)^2+\big(y(l)-y(l-1)\big)^2+\big(z(l)-z(l-1)\big)^2 \quad (1)$$

   Next, the given user trace are segmented into stop segments or move segments. The stop segments are those whose displacement amount is zero while the remaining segments become the move segments. For segment n, the following pieces of information are stored in Segment(n), i.e., state, length, start, end, sumDisplacement, and count, indicating the type of this segment (stop or move), the length of this segment in terms of the number of loop-counts, the starting loop-count, the ending loop-count, the total amount of displacement, and the number of frames extracted from this segment, respectively. In addition, for each stop-type segment, the information on the stop position, stopPosition, is also stored.
2. At the frame extraction step, the following two conditions are applied to all stop-type segments for extraction of them:

   Condition I: Is its stopPosition not adjacent to stopPosition of the previously extracted stop-type segment?

   Condition II: Is its length above the average length of all stop-type segments?

The former condition is employed to prevent extraction of frames showing similar user experiences, such as frames showing the user stopping to view the same exhibit. The latter condition is employed to prevent extraction of frames showing spontaneous stops by the user.

For each stop-type segment n satisfying both conditions, its count is incremented and a frame, Frame(f), is added into a frame-candidate list, by which the content of Segment(n) is copied to FrameInfo(f), containing the information about Frame(f), where f represents the frame number. If the number of extracted frames is more than the number of frames specified by the user, frames in the frame-candidate list will be removed in increasing order of FrameInfo(f).*length* until the specified number of frames, N, is met.

Frame extraction is done for move-type segments only when the number of stop-type frames extracted above is lower than N. In this case, this is done for the move-type segments in decreasing order of sumDisplacement.Namely, first, a frame, Frame(f), is extracted from the move-type segment with the highest sumDisplacement and added to the frame-candidate list, by which after incrementing count Segment(n) is copied to FrameInfo(f); then the one with the second highest sumDisplacement; and so on. For example, if N is 10, and there exist 7 stop-type segments and 5 move-type segments, then, after extraction of 7 frames from the stop-type segments, 3 frames will be extracted from the move-type segments with the top-three sumDisplacement.

If frame extraction has been done for all move-type segments, but the total number of extracted frames is still lower than N, then another round of frame extraction will be done for the move-type segments. This process is repeated until N is filled. Due to such repetitions, multiple frames might be extracted from a same move-type segment; however, such frames have different snapshot timings as discussed below.

Once N is met, the frame-candidate list will be copied to a frame list, at which the snapshot timing $t(f)$ is determined for being used later in the renderer module. For a stop-type frame, Frame(f), the snapshot timing is as follows:

$$t(f) = \text{FrameInfo}(f).start + 0.5\text{FrameInfo}(f).length \tag{2}$$

Although any timing in [FrameInfo(f).*start*, FrameInfo(f).*end*] should produce the similar (not same because the user avatar's face or hands might move) comic frame, we empirically decided to use the above formula.

For a move-type frame, Frame(f), the snapshot timing is as follows:

$$t(f) = \text{FrameInfo}(f).start + \frac{\text{FrameInfo}(f).length}{\text{FrameInfo}(f).count + 1} \tag{3}$$

3. At the frame layout module, each frame in the frame list is assigned a page and the position therein. This decision is based on the information on N, the page margins, and the number of frames per row and column.
4. At the renderer module, the system scans the visit log from the beginning and renders the snapshot image of the user experience with the snapshot timing $t(f)$ of each frame in the frame list. Each rendered image is then placed on the specific position in the predetermined comic page according to the information stored in the corresponding frame. Finally, all comic pages are outputted as an image file.

3 Evaluation

After implementing the proposed system, we conducted a user evaluation. The objective of this user evaluation is to examine if the proposed system can generate a comic that properly summarizes the user experience viewing exhibits at a metaverse museum.

3.1 Implementation

We implemented the system by adding the above four modules to the open-source SL client program (SL viewer program). For this work, we adopted a typical comic layout where the order to read is in the raster order, from top-left to bottom-right, and all frames have the same size.

For the evaluation, we targeted user experiences at a SL museum[1] designed and operated by members of Global COE (Center of Exellence) Progam“ Digital Humanities Center for Japanese Arts and Culture ” of Ritsumeikan Unviersity. This museum is aimed at a virtual exhibition of Kaga Okunizome Dyeing, kimono and bedding from the Ishikawa region in Japan during the latter part of the Edo period until the beginning of the Showa period. However, we would like to point out that our system is also applicable to other museums in SL as well as other metaverse.

Figure 2 shows the museum building in which 19 exhibits and two posters are located. Our reasons for extracting this museum are that (1) the exhibition therein has a high cultural value because Kaga Okunizome dyeing is famous in Japan and (2) there is no copyright problem because the authors belong also to the aforementioned Global COE. For other representative museums in SL, please refer to the work by Urban et al. (Urban et al. 2007).

[1] http://slurl.com/secondlife/rits%20gcoe%20jdh/167/189/22

Fig. 2 The metaverse museum used in this work

3.2 Evaluation Outline

We compared comics whose frames were extracted by the proposed frame-extraction method and a baseline frame-extraction method. Each method was implemented in our comic-generation system. The latter method uses a fixed-and-equal interval for extraction of frames in a given comic. We first requested 26 participants, who are undergraduate or graduate students in our department, to watch each of the three video clips, showing typical visits of each of the three visitor types (Sparacino 2003): busy[2], selective[3], and greedy[4]. After watching a video clip, each participant was asked to compare two comics of the same number of frames that were extracted by the two methods from the corresponding visit log used for the video clip, without being told which method was used for a given comic. In particular, each participant was asked to answer the following four questions for each comic in the typical five-level Likert scale:

Q1 Does the comic have the content well representing the video-clip play?
Q2 Does the comic have the minimum amount of unnecessary frames?
Q3 Does the comic well display exhibits?

[2] http://www.youtube.com/watch_popup?v=ijxwUrHejG8&vq=medium
[3] http://www.youtube.com/watch_popup?v=H8eUJ6JZGXo&vq=medium
[4] http://www.youtube.com/watch_popup?v=DtVFf5gAUZI&vq=me dium

Table 1 Average score of each method for the busy type

Question #	Proposed Method	Baseline Method
Q1	3.38	3.08
Q2	3.92	3.15
Q3	2.77	2.77
Q4	3.73	3.38

Table 2 Average score of each method for the selective type

Question #	Proposed Method	Baseline Method
Q1	4.19	3.35
Q2	3.73	2.19
Q3	3.92	3.62
Q4	3.81	2.38

Table 3 Average score of each method for the greedy type

Question #	Proposed Method	Baseline Method
Q1	4.23	3.85
Q2	3.69	2.19
Q3	3.5	4.04
Q4	3.58	2.31

Q4 Does the comic have a dynamic story?

The participants were also asked to write a short reason behind their rating for each question.

3.3 Results and Discussions

Tables 1, 2, and 3 show the average score of each method for the busy-type, the selective-type, and the greedy-type, respectively. The proposed method outperforms the baseline in all questions, except Q3. Q3 is related to the quality of the camerawork and the relating parameters. Hence, the camerawork of the comic generation system should be improved.

Figure 3 shows the first page of the selective-type comics whose frames were extracted by the proposed method and the baseline method. It can be seen that the latter method extracted a number of similar frames while the former method did not. Similar comments were obtained from participants as their reasons for rating the former method higher than the latter one.

4 Related Work

A storytelling system was implemented that generates slides from images taken at multiple locations on a given map (Fujita & Arikawa 2008). For segmentation of videos, a method (Xu et al. 2009) exists that uses histograms of human motions, but this method aims at fast movements such as sports or dances, not slow movements typically seen in SL avatars while they are visiting museums. A scripting system (Zhang et al. 2007.) was proposed for automating cinematics and cut-screens that facilitate video-game production processes such as the control of transitions between images, and the annotation of texts and sounds to image backgrounds.

There is a number of existing work in addition to the authors' previous work (Shuda & Thawonmas 2008; Thawonmas & Shuda 2008; Thawonmas and Oda 2010) on comic generation for online games. Such work includes comic generation for online games using a combination of screenshots (Chan et al. 2009), rather than using the targeted game's engine for re-rendering frames as done in our approach, and for first person shooters (Shamir et al. 2006). Comics were also used for summarization of experiences in a conference (Sumi et al. 2002), daily experiences (Cho et al. 2007), and videos or movies (Calic et al. 2007; Hwang et al. 2006; Tobita 2010).

5 Conclusions and Future Work

We proposed and implemented a system for generating comics from user experiences during their museum visits. The system enables extraction of frames that contain interesting exhibits, from the view point of a user of interest. This was achieved by taking into account the amount of displacement in the museum space and giving higher priorities to stop-type segments having long stop time. The user evaluation confirmed the effectiveness, in properly representing the aforementioned user experiences, of the proposed system for all visitor types. As our future work, we plan to increase the expressivity of comics. One possible research theme is to strengthen the comic layout mechanism. This would increase the variety of comic frames, such as large frames, small frames, slanted frames, etc. As already stated in 3.3, another theme is to improve the camerawork so that a more proper set of camera parameters, i.e., the camera angle, camera position, and zoom position, is applied to a given frame. The mechanisms proposed in our previous work for online games (Thawonmas and Shuda 2008; Thawonmas and Ko 2010) will be extended, respectively, for these two research themes.

Acknowledgment

This work is supported in part by the MEXT Global COE Program - Digital Humanities Center for Japanese Arts and Cultures, Ritsumeikan University.

References

1. Calic, J. et al., 2007. Efficient Layout of Comic-like Video Summaries, IEEE Transactions on Circuits and Systems for Video Technology, Vol. 17, No. 7, pp. 931-936.
2. Chan, C.-J. et al., 2009. Automatic Storytelling in Comics: A Case Study on World of Warcraft, CHI Extended Abstracts 2009, pp. 3589-3594.
3. Cho, S.B. et al., 2007. Generating Cartoon-Style Summary of Daily Life with Multimedia Mobile Devices, Proc. of IEA/AIE 2007, pp. 135-144.
4. Fujita, H. and Arikawa, M., 2008. Animation of Mapped Photo Collections for Storytelling, IEICE TRANS. INF. & SYST., Special Section/Issue on Human Communication III, Vol. E91-D, No. 6, pp. 1681-1692.
5. Hwang, W.-I., et al. 2006. Cinema comics: Cartoon generation from video stream. Proc. of GRAPP 2006, pp. 299-304.
6. Lowood, H., 2006. High-performance play: The making of machinima, Journal of Media Practice, Vol. 7, No. 1, pp. 25-42.
7. Prendinger, H., 2009. The Global Lab: Towards a Virtual Mobility Platform for an Eco -Friendly Society, Transactions of the Virtual Reality Society of Japan, Vol. 14, No. 2, pp. 163-170.
8. Shamir, A. et al., 2006. Generating Comics from 3D Interactive Computer Graphics, IEEE Computer Graphics and Applications, Vol. 26, No. 3, pp. 53-61.
9. Shuda, T. and Thawonmas, R., 2008. Frame Selection for Automatic Comic Generation from Game Log, Lecture Notes in Computer Science, Vol. 5309 (ICEC 2008: Entertainment Computing), pp. 179-184
10. Sparacino, F., 2003. Sto(ry)chastics: A Bayesian Network Architecture for User Modeling and Computational Storytelling for Interactive Spaces, Lecture Notes in Computer Science, Vol. 2864 (UbiComp 2003: Ubiquitous Computing), pp.54-72.
11. Sumi, Y. et al., 2002. ComicDiary: Representing individual experiences in a comic style, Ubicomp 2002, pp. 16-32.
12. Thawonmas, R. and Oda, K. 2010. Rule-Based Camerawork Controller for Automatic Comic Generation from Game Log, Lecture Notes in Computer Science, Vol. 6243 (ICEC 2010: Entertainment Computing), pp. 326-333.
13. Thawonmas, R. and Shuda, T., 2008. Comic Layout for Automatic Comic Generation from Game Log, Proc. of 1st IFIP Entertainment Computing Symposium, Vol. 279, pp. 105-115.
14. Tobita, H., 2010. DigestManga: interactive movie summarizing through comic visualization, CHI. Extended Abstracts 2010, pp. 3751-3756.
15. Urban, R. et al., 2007. A Second Life for Your Museum: 3D Multi-User Virtual Environments and Museums, Proc. of Museums and the Web 2007, San Francisco, April 2007.
16. Xu, J. et al., 2009. Temporal segmentation of 3-D video by histogram-based feature vectors, IEEE Transactions on Circuits and Systems for Video Technology archive, Vol. 19, No. 6, pp. 870-881.
17. Zhang, W. et al., 2007. Story scripting for automating cinematics and cut-scenes in video games, Proc. of the 2007 conference on Future Play, pp. 152-159.

(a) Proposed Method

(b) Baseline Method

Fig. 3 Fist page of the generated comics for the selective type