Eurographics Workshop on Intelligent Cinematography and Editing (2017)
W. Bares, V. Gandhi, Q. Galvane, and R. Ronfard (Editors)

Design of an Intelligent Navigation System for Participative Computer Animation

Iou-Shiuan Liu[1], Tsai-Yen Li[1], and Marc Christie[2]

[1] National Chengchi University, Taipei, Taiwan
[2]IRISA/INRIA Rennes Bretagne Atlantique, Rennes, France

Abstract

In this paper, we propose a novel way of interactive entertainment, called Participative Computer Animation, allowing a user to participate in a computer animated story as an observer. We consider this form of entertainment as a kind of interactive storytelling, in which the presentation and perception of the story is under the control of a user through a first-person camera. As the animation of the story unfolds, the user needs to follow and view the relevant events, a complex task which requires him to navigate in the 3D environment, and hence reduces his immersion. We therefore propose to design an intelligent navigation mechanism, in which the system can voluntarily assist the user in reaching some designated best view configurations under time constraint. We have implemented such a system and invited a few users in a pilot study to evaluate the system and provide feedback. The experimental results show that our participative computer animation system can enhance the sense of presence while the intelligent navigation mechanism can improve the quality of perceiving the animated story.

Categories and Subject Descriptors (according to ACM CCS): I.3.6 [Computer Graphics]: ComputerGraphics—Methodology and Techniques – Interaction Techniques

1. Introduction

Interactive storytelling (IS) has been an active research topic for some years. With the advances in Artificial Intelligence (AI) and Virtual Reality (VR) technologies, 3D virtual environments become a popular platform to design interactive stories for users to experience. However, it also presents a great challenge to design interactive systems that can keep a good balance between narrative and controllability. In other words, an ideal interactive storytelling system should allow the author(s) of a story to share some of the controls with the audience through interaction design. On the other hand, the level of controls should be kept simple and intuitive such that the immersion of a user into the story is not broken.

In this paper, we adopt a different view. We present an approach to what we refer to as *Participative Computer Animation*. Computer animated sequences are either fully automatic like a CG movie being played in which there is no interaction, or fully interactive in that the user interacts by changing the scene contents, typically like 3D games. In this regard, participative computer animation consists in intentionally exploring a computer animated world without interacting with it. In that, it departs from interactive storytelling in which interactions alter the story, and can be more precisely viewed as *experiential passive storytelling*, like a theater play in which there would be some specific locations from which to view the play and users could choose which views to use and at which time.

In the literature, many contributions have been exploring how a user could be guided in static 3D environments, through guided tour museums [ZSA03, AVF04], interactive navigation systems [HW97], or proximal inspection of objects [Bou14]. Yet their transposition to dynamically evolving environments is not straightforward.

The exploration of dynamic environments has been addressed for some specific applications. Typically automatically computing an overview of an animated sequence has been addressed by viewpoint computation techniques [YLCL11] and coupled with viewpoint transition techniques [ACoYL08, YLL*12]. Despite being able to automatically detect relevant events to portray, these systems are not interactive and do not offer the possibility of navigating through the environment.

The challenge to address is therefore proposing a system which assists the user in navigating in animated environments in a way to be immersed in the 3D world, without interacting. In turn, this requires to provide techniques to support the user's navigation towards relevant positions and angles at which the events can be effectively viewed. The key is in maintaining the balance between user interaction (which tends to break immersion) and passive visualisation (which improves immersion). In this paper, we propose an

DOI: 10.2312/wiced.20171075

intelligent navigation system which guides the user, as he interacts, towards temporally indexed viewpoints conveying the animation at the appropriate moments. The approach we propose in this paper is to author relevant viewpoints in the timeline of a 3D animation, and then propose means to interactively navigate between these viewpoints though minimal commands, using attraction and repulsion techniques to move the viewer closer to a relevant viewpoint, or let him/her quit the viewpoint and move to other viewpoints.

Such a process finds applications in what could be a novel experiential exploration of movies where the user plays the camera, for games and for virtual reality. We expect that in such applications, one can also find an increased replay value for interactive storytelling systems by allowing a user to experience an animated story several times through different courses.

The paper is organized as follows. First we will review the related work and then formulate our approach in an overview section. We will then present the design of the interaction techniques and present results of early experiments before discussion and conclusion.

2. Related Work

We first review navigation techniques in 3D environments. We then review camera control techniques which assist designers in placing virtual cameras in 3D environments.

2.1. 3D navigation techniques

A navigation task for a user is the process of moving a first-person camera in a 3D environment while being guided by some visual features to see or some tasks to perform (a process different from exploration in which there is no task). Navigation tasks require a mechanism to guide the user towards computed or predefined task-dependent locations. The challenge in designing such a system lies in the capacity to lower the number of user inputs, while enabling enough user freedom. A number of systems have been proposed, essentially around the idea of guided virtual tours. In this review we will focus only on interactive systems, in opposition to systems which automatically compute a path given target locations (such as [OSTG09, ZSA03]).

Interactively guiding a user requires to apply some form of attraction towards a target location while avoiding obstacles. In this perspective, potential fields have shown to be an effective solution. Hong etal. [HMK*97] rely on the computation of such a potential field to compute a force that pushing the user away from boundaries, while progressing along the organ for virtual colonoscopy inspection. The approach has inspired other potentiel field based approaches [BJ09, Bec02]. Hanson etal [HW97] relied on a similar approach, however using a vector field representation to smoothly navigate on terrain surfaces.

A second category of approaches rely on planning techniques, that present the benefit of avoiding local minima that appear with potential or vector fields. Planning techniques need to first rely on an abstraction of a 3D environment, either based on topologic representations or regular sampling with various – often hierarchically organized – primitives such as spheres or cubes. A path is then computed through such primitives and smoothed with dedicated techniques. Elmqvist etal [ETT07] rely on a regular decomposition of a 3D landscape to then compute optimal paths through selected locations using a Travel Salesman Approach. A mass-spring physical system then guides the user along the computed path between the selected locations while enabling a certain level of freedom. Chittaro etal. rely on regular grid decomposition [CRI03] and propose that a virtual character acts as a guide which the user can follow in the 3D environment. Andujar etal. [AVF04] rely on an alternate representation – a geometric cell-and-portal structure of a 3D environment – to compute an optimal path with an A* planner, linking pre-selected targets to achieve a guided tour.

While such systems address the common problems in guided tours (avoid disorientation, avoid missing important locations, and not knowing whether the tour is completed [CRI03]), these are designed in mind for static environments (with exception of [BJ09, Bec02]), and don't target the idea of a *participative experience*.

2.2. Camera control techniques

Besides navigation techniques to guide users, the community around virtual camera control has also proposed a number of automated viewpoint computations techniques to convey dynamic events occurring in animated 3D events. Lino et al. [LCL*10] proposed a real-time cinematographic system which automatically computes camera angles, camera trajectories, and camera edits to portray a dynamically evolving sequence. The user however had no interaction in the process. Autonomous overview systems also proposed a method based on the analysis of human motions [ACoYL08, YLCL11], or in a more principled way by an analysis of social relations (proxemics), such as locations and distances between characters [YLL*12]. Finally Galvane [GCR*13] proposed to detect events in complex interactions such as crowds and portray them with a set of self-organizing cameras driven by steering behaviors so as to cover alternate viewpoints.

While such approaches provide a foundation to extract relevant events, and compute relevant viewpoints to portray these events, the user is not included in the process, and more importantly the intent is not to let the user navigate in the environment in a way that he/she can experience a narrative.

3. Approach

Our proposed approach takes for input an animated sequence as well as a set of viewpoints authored by a designer and placed in the timeline to portray selected key events in the story. Our interactive system then reacts to the user's displacements and attracts the user's first-person camera to the nearest candidate viewpoint when the time and location are all in the influential ranges of a key event.

3.1. System architecture

An overall architecture of the system is depicted in Figure 1. A user of the system acts as a participant observing the animated story through participation in a 3D virtual environment. As indicated in the left green boxes, we assume that we are given an animation script and its related animation designs such as 3D models, character animations, lighting, etc. In addition, the author of the story will also specify the key events in the story that require the participant's attention in order to understand the progress of story. With each key event in time, the author specifies several candidate viewpoints in different regions of the scene that can be used to set up a camera to render the animation and present the story to the participant in a clear way. Examples of the candidate viewpoints for a same scene are shown in Figure 2. The participant uses a common input modal such as keyboard to control the position and orientation of the first-person viewpoint. The input of user control is sent to a user model module to determine the intention of the participant on approaching or leaving a specific viewpoint. The result is then sent to the camera planning module together with the user control to computing the final command to the camera update, with which the scene is then updated.

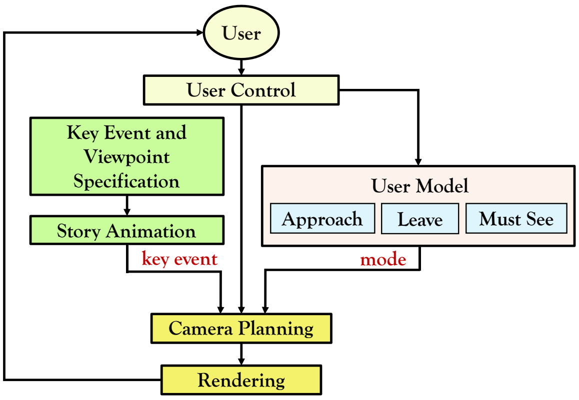

Figure 1: *System architecture*

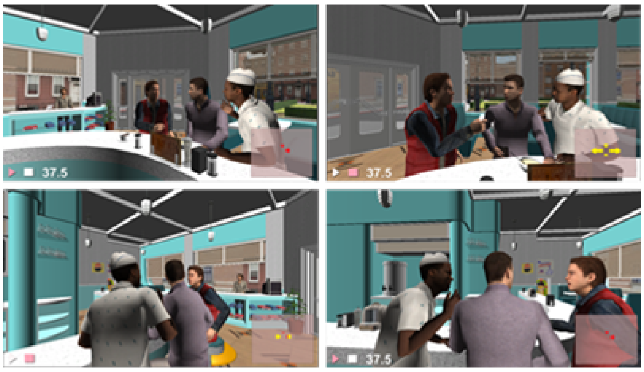

Figure 2: *A typical scene rendered from four different candidate viewpoints*

3.2. System control

In our system, we try to separate the head rotation from the orientation of the controlled avatar body. As in a typical navigation control in a 3D virtual environment, a user uses the keyboard to move the first-person camera attached to the user's eyes forward or backward by pressing the up and down arrow keys, respectively. Similarly, the left and right arrow keys can be used to rotate the camera counterclockwise or clockwise, respectively. The longer the keys are pressed, the farther the camera will be moved or rotated. When navigating in a real scene, a user usually can take advantage of the additional degree of freedom on head rotation to keep the eye focusing on the critical part of the scene and avoid missing an event. Similarly, in our system, we have unlocked the head rotation and confined it in a reasonable range. Consequently, the viewing direction and the body facing direction may not be the same.

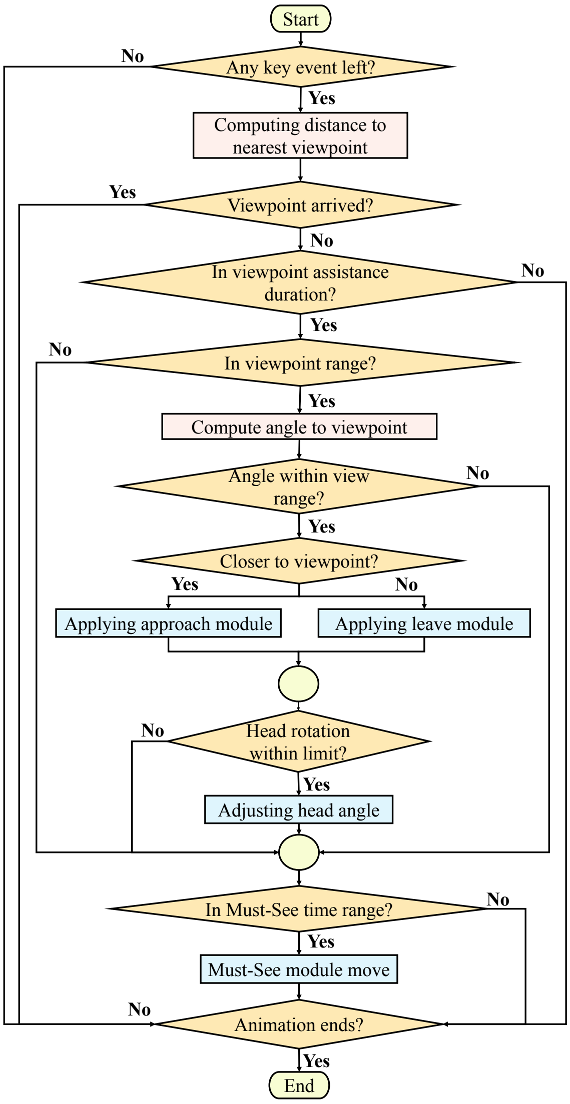

Figure 3: *Flowchart of System control*

System control is designed to assist a user in viewing an animated story by providing necessary adjustment on camera position and view angle to grasp a happening critical event. The flowchart of the system control is shown in Figure 3. When the system control is enabled, the system will voluntarily assist navigation control in the following way. Whenever a key event exists, the system will compute the distance of the current viewpoint from the nearest candidate viewpoint (out of multiple candidates) and see if it is within the influential range (a circle centered at the target viewpoint). Besides, the assistance will be activated only if the current time is within a time range before the key event takes place. The system will then further check if the target viewpoint is within the current view range in order to avoid moving backward. Depending on whether the camera is moving toward or away from the target viewpoint, the system will guess the user's intention on approaching or leaving the target viewpoint, and different assistance policies will be used to adjust the system control magnitude. If there exists discrepancy on the facing direction and the direction to the target viewpoint, the system will take advantage of the head rotation to turn the head toward the key event as early as possible as long as the head rotation is within the joint limits. Finally, if the time is up and the Must-See module is enabled, the system will perform a cut to transport the camera to the target viewpoint immediately in order not to miss the key event.

We assume that the position ($\boldsymbol{p}$) of a camera that is updated through its current velocity, $\boldsymbol{v}$ such that $\boldsymbol{p}' = \boldsymbol{p} + \boldsymbol{v} \times \Delta t$. The velocity vector, under the system-assisted mode, comes from two components: user control ($\boldsymbol{v_u}$) and system control ($\boldsymbol{v_s}$) using $\boldsymbol{v} = \boldsymbol{v_u} + \boldsymbol{v_s}$. The user control velocity is activated when the up and down buttons are pressed, and the camera is supposed to move forward ($\delta_p = +1$) or backward ($\delta_p = -1$) along the facing direction $\boldsymbol{u}(\theta)$. This is expressed as : $\mathbf{v_u} = \delta_p \times k_p \times \mathbf{u}(\theta)$. The system control velocity tries to move the camera toward the target viewpoint ($\boldsymbol{u_r}$), a normalized vector, with its magnitude determined by two factors: expected speed to reach the target viewpoint in time and the scaling factor for user intention (approaching or leaving).

$$\mathbf{v_s} = min\left(k_{s1} \times \frac{|\mathbf{d}_r|}{t_r} \times \left(1 + \delta_p \times \left(\frac{min(b, b_{max})}{b_{max}}\right)^2\right), v_{max}\right) \times \mathbf{u}_r$$

The expected speed is determined by the remaining time (t_r) to the key event and the distance (d_r) from the target viewpoint. The remaining time is subtracted by a constant k_t, specified by the designer, to reach the target viewpoint some time before the event happens, expressed as $t_r = max(tg - t_c - k_t, t_{min})$. The scaling factor is a quadratic function determined by the duration (b) that an arrow key is pressed when it is within the maximal duration (b_{max}). The range of the scaling factor is 0 to 2 with a nominal value at 1. In the approaching and leaving modes, the speed will be scaled up and down, respectively, by multiplying the scaling factor with the expected speed. When the keys are not pressed, the system control will still be in effect but just not scaled. The rotational part of the camera update is similar to the translation part described above except that expected rotational speed is computed according to the remaining rotational difference from the target viewpoint. The rotational part is expressed as:

$$\theta' = \theta + \omega \times \Delta t$$
$$\omega = \omega_u + \omega_s$$
$$\omega_u = \delta_\theta \times k_\theta$$
$$\omega_s = k_{s2} \times \frac{\alpha_r}{t_r}$$

3.3. Implementation and Interface design

We have implemented the computer-assisted navigation system proposed above in Unity3D. A snapshot of the graphical user interface showing the 3D scene and animation is shown in Figure 4. In the first-person camera view, a user can play and stop the animation by the buttons at the lower left corner of the screen. In addition to the regular control keys, the user can also call up a hidden panel on the lower right corner, showing the positional and rotational control vectors by the user (in yellow) and the system (in red), respectively. The final control vector is the vector sum of these two components. This visual panel is used for debugging and for illustrating the effect of system control to the users. Besides, in order to let the user be aware of the candidate target viewpoints, we put a spot light on each target viewpoint location as a hint to guide the user to move onto it. An example of candidate target viewpoints and its influential range (yellow circle) at a given time frame in the story line is shown in Figure 5.

Figure 4: *Snapshot of the user interface with control buttons and vectors*

4. Pilot Study

In order to evaluate the design of our system, we have conducted a pilot study to seek feedback from the users for future improvement.

4.1. Experimental settings

We have designed a sample story using a 3D animation from a fragment of an episode of the movie "Back to the Future" [GRLC15]. In the episode, the script includes two seated actors in dialog and another two actors moving around and intervening the dialog. We have set 11 key events in the 80 seconds of animation with a total of 25 viewpoints.

Figure 5: *Various viewpoints and their corresponding activation range for system control*

Table 1: *Evaluation of different runs from various aspects*

Navigation Conditions	Key Events Seen	Framing Satisfied	Plot Understood	Process Comfortable
Regular (1st run)	3.00 (1.15)	3.33 (0.94)	3.00 (0.82)	3.00 (0.82)
Regular (2nd run)	3.67 (0.75)	3.67 (0.75)	4.00 (0.00)	3.50 (0.50)
Assisted (1st run)	4.17 (0.37)*	4.17 (0.37)*	4.17 (0.69)	3.83 (0.37)*
Assisted (Second)	4.17 (0.69)	4.17 (0.69)	4.17 (0.69)	4.17 (0.69)*
Assisted (Must See)	4.83 (0.37)	4.00 (0.58)	4.33 (0.47)	2.83 (0.69)

* $p<0.05$

We have invited six users as our subjects to take part in this within-subject experiment. Each subject will participate in the animation as an observer for five runs, consisting of two runs with no system assistance, two runs with system assistance, and an additional run with the system assistance and the Must-See mechanism enabled. In order to reduce the learning effect, we have shuffled the first four runs such that the learning effect can be cancelled out. The fifth run with the Must-See module always came last. The users did not know which run is with the system assistance and were asked to fill in a simple survey at the end of each run with questions such as if the key events were seen, if they were satisfied with the framing, if they could understand the plot, and if they felt comfortable with the participatory process.

At the end of the five runs, the process is unblinded, and the subjects were informed of the system settings of each run such that they know which ones are with and without system assistance. Then, the subjects were asked to fill in another questionnaire with 14 questions to evaluate the overall system. All the questions were answered on a 5-point Likert scale from 1 to 5 for strongly disagree to strongly agree. Following the survey, we conducted an interview with open questions to acquire further feedback from the subjects. In addition, we have also recorded the inputs and the computed controls and user modes of each frame in the course of participating in the animation for further analysis.

4.2. Experimental results

In the survey at the end of each run, we collected the evaluation of the subjects on the system from four aspects with and without system assistance. The statistic data including arithmetic mean (average) and standard deviation (SD, in Parentheses) have been computed and reported in Table 1. The averages for the two runs without system assistance are almost below 4.00 (3.00-4.00) while the two runs with system assistance are almost above 4.00 except for one (3.83 in process comfortable). Three averages for the last run (Must-See enabled) are above 4.00 but the score for process comfortable is below 3.0 (2.83). The result reveals that, despite that this is a blind experiment, the subjects all prefer the navigation with system assistance and the second ones all receive higher scores than the first ones, which is expected. The scores of the first run with system assistance are higher than the scores of the first run without system assistance with statistical significance in the first, second, and fourth aspects. If we compare the scores for the second runs with and without system assistance, we find that only the last aspect is statistically significant. This may imply that system assistance is especially useful in the first run of the system when the user is still not familiar with virtual scene. Nevertheless, the subjects do not seem to appreciate the assistance of the Must-See mechanism that transport the viewpoint to the target when the time is up probably due to the visual disturbance of jump cut.

At the end of all runs, the subjects are asked to fill in a survey consisting of 14 questions. The survey results with average and standard deviation are shown in Table 2. In the first four questions, we asked the subjects if they allow the system to take over the control on the rotation and translation of the viewpoint and if their speeds are adequate. The only score below 4.0 is the appropriateness of the translation speed but it is also the one with the highest standard deviation, which means that the opinions vary more than the other scores. In general, the subjects are willing to let the system to take over the control to assist them to view the critical events in time, which is also reflected in the scores for questions 5 and 6. As a result, the subjects can understand the story plot better and the viewing quality is also improved as indicated in the scores of questions 7 and 8. Almost all subjects agree that they prefer the navigation experience with system assistance to the one without (question 9) and they consider the system having a great potential of application value (question 10). They also agree that the participative way of watching the animation is preferred to the traditional way (question 11). In addition, they are more willing to watch the same animation multiple times (question 12), which highlights the replay value of our system as an interactive storytelling system. The subjects also agree that the proposed participative way gives a better sense of presence than the traditional one (question 13). However, the better sense of presence or replay value does not necessarily imply that the subjects like the story more, as reflected in the score of question 14.

In addition to the subjective feedback through questionnaire, we have also collected the log of user behavior about how and when the system control was activated to assist the subjects in navigation. A

Table 2: *Overall evaluation questionnaire (Part I)*

Questionnaire	Avg	SD
1. Did you allow the system to take over control on view angle?	4.67	0.75
2. Was the rotational speed of the system-assisted view angle moderate?	4.50	0.50
3. Did you allow the system to take over translational control?	4.00	0.82
4. Was the speed of system assisted movement moderate?	3.83	0.90
5. Was the system helpful in assisting you to control the camera view?	4.33	0.47
6. Did the system help you see the key events of the story?	4.50	0.50
7. Did the system help you understand the plot of the story?	4.00	0.58
8. Did this system's assistance improve your viewing quality?	4.33	0.47
9. Do you prefer the system with assistance to the one without?	4.50	0.50
10. Do you think the system has potential for further application development?	4.33	0.75
11. Do you prefer to watch an animation by the "Participative" way rather than by the "Traditional" way?	4.00	1.15
12. Are you willing to watch the same animation many times by the "Participative" way?	4.33	0.75
13. Compared to the traditional way, does the "Participative" way enhance the sense of presence?	4.33	0.94
14. Do you like to story more by watching the animation with the "Participative" way?	3.83	0.37

typical example of user and system controls in a duration of 14 seconds is shown in Figure 6. The blue line is the control magnitude issued by the user while the orange line is the control supplied by the system. When the user controls his viewpoint to enter the influential range of a target viewpoint and move toward the viewpoint, such as at 7.8sec, the system activates the approach user mode by supplying increasing system control. The system control remains and assists the user moves toward the target viewpoint for the key event happening at 13.2sec even after the key was released later. In the second half of the example, the user tried to move away from the target viewpoint of the second key event (at 19sec) and activated the leave mode. In the beginning, the system still tried to move the camera to the target viewpoint until the key was pressed long enough, in which case, the system control dropped to zero (at 17.8sec).

In Figure 7, we show the distances of the current viewpoints of the six subjects from the target viewpoint location for the same period of time (49-55sec). The corresponding system controls for these subjects at the same period of time are also shown in Figure 8. The distances of the subjects from the target viewpoint are all different at 49sec, and they started to move toward the target also at different time. For example, subjects B, C, E, and F started to move at 51, 50.5, 50, and 50, respectively. The behaviors of three subjects worth notice. First, subject B was the farthest from the target but also the latest one to move. Therefore, we observe a ramp starting at 50.5sec that issued the strongest system control (highest speed) in order to bring the viewpoint to the target in time. Second, subject F was the closest to the target at the beginning; however, he tried to move away to another candidate target viewpoint starting from time 50.0sec. The distance went up until 51.5sec and then drop to zero again gradually. Third. subject E is one of the closest to the target viewpoint at the beginning and gradually moves toward it. Consequently, the system control observed in Figure 8 for subject C is also the mildest.

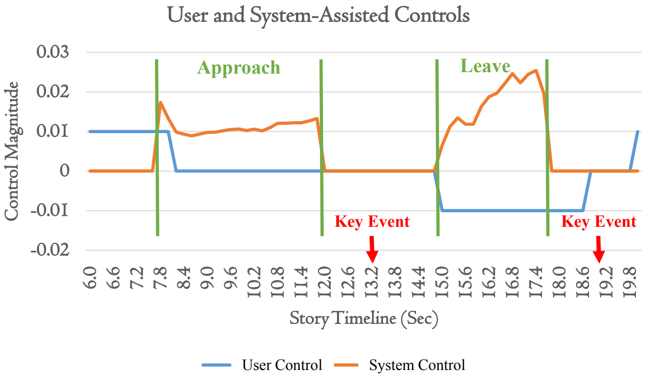

Figure 6: *An example of user and system controls in the approach and leaving mode*

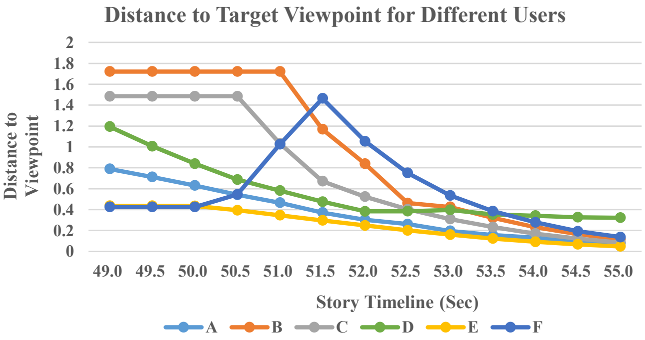

Figure 7: *Comparison of distance to the target viewpoint location for different users*

5. Conclusions and Future Work

In this work, we have proposed a novel interactive storytelling system allowing a user to navigate in a 3D environment and participate in an animated story as an observer. Unlike previous work in guided navigation, we have designed a computer-assisted control system with time constraint to assist a user to move to a good viewpoint before a specific key event happens. In the pilot study conducted for the implemented system, we have obtained positive feedback from the users about the system-assisted control mechanism as well

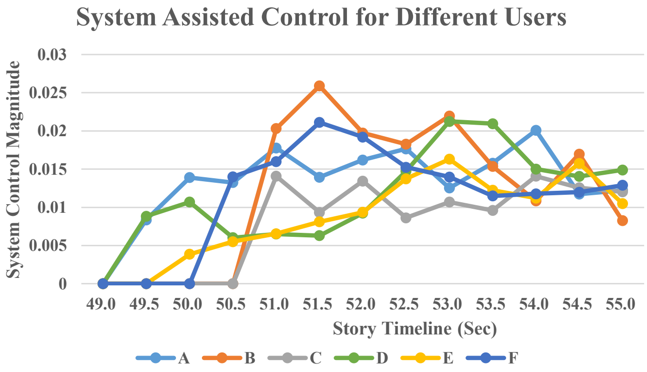

Figure 8: *Comparison of system-assisted control for different users*

as the participative way of experiencing an animated story. From the experimental data, we learn that although the system-assisted mechanism may not make the users like the story more, the users all prefer the novel design to the traditional one. It also increases the the replay value and the sense of presence in such an experiential passive storytelling system.

In the future, we would like to extend the experiment to a larger scale to validate the effectiveness of the proposed model statically. In addition, we would like to design a more personalized user model about the trait and intent of a user in order to further customize the system control to assist the user in navigation. Currently, we only allow the user to perform continuous navigation to a target viewpoint except for the "Must-See" mechanism that transports the camera to the target viewpoint immediately. From the experimental result, we have learned that the users may not like the design of Must-See, which may result in a jump cut. In the future, we will consider adding an interface to allow a user to select appropriate viewpoint position at distance that does not result in a jump cut. We also hope to extend the key event to a period of time instead of a point in time in the current implementation such that one can stay at a viewpoint location for a longer time to have a better observation. Furthermore, we hope to enhance the system by allowing the user to participate in the animated story in a more active way and have interactions with the animated characters controlled by the system.

References

[ACoYL08] Assa J., Cohen-or D., Yeh I.-C., Lee T.: Motion Overview of Human Actions. *ACM Transactions on Graphics (TOG) - Proceedings of ACM SIGGRAPH Asia 2008 27*, 5 (December 2008). URL: http://dl.acm.org/citation.cfm?doid=1457515.1409068, doi:10.1145/1409060.1409068. 1, 2

[AVF04] Andújar C. G., Vázquez P. P. A., Fairén M. G.: Way-Finder: Guided Tours Through Complex Walkthrough Models. *Comput. Graph. Forum 23*, 3 (2004), 499–508. 1, 2

[Bec02] Beckhaus S.: *Dynamic Potential Fields for Guided Exlporation in Virtual Environments*. PhD thesis, Fakultät für Informatik, University of Magdeburg, 2002. 2

[BJ09] Burelli P., Jhala A.: Dynamic Artificial Potential Fields for Autonomous Camera Control. *Artificial Intelligence and Interactive Digital Entertainment* (2009). 2

[Bou14] Boubekeur T.: ShellCam: Interactive geometry-aware virtual camera control. In *IEEE International Conference on Image Processing* (2014). 1

[CON08] Christie M., Oliver P., Normand J.-M.: Camera Control in Computer Graphics. *Computer Graphics Forum 27*, 8 (2008).

[CRI03] Chittaro L., Ranon R., Ieronutti L.: Guiding visitors of web3d worlds through automatically generated tours. In *Proceedings of the Eighth International Conference on 3D Web Technology* (New York, NY, USA, 2003), Web3D '03, ACM, pp. 27–38. URL: http://doi.acm.org/10.1145/636593.636598, doi:10.1145/636593.636598. 2

[ETT07] Elmqvist N., Tudoreanu M. E., Tsigas P.: Tour generation for exploration of 3D virtual environments. In *Proceedings of the 2007 ACM symposium on Virtual reality software and technology* (2007), ACM, pp. 207–210. 2

[GCR*13] Galvane Q., Christie M., Ronfard R., Lim C.-K., Cani M.-P.: Steering Behaviors for Autonomous Cameras. In *ACM SIGGRAPH conference on Motion in Games* (2013), MIG '13 Proceedings of Motion on Games. 2

[GRCS14] Galvane Q., Ronfard R., Christie M., Szilas N.: Narrative-Driven Camera Control for Cinematic Replay of Computer Games. In *Motion In Games* (Los Angeles, United States, Nov. 2014). URL: https://hal.inria.fr/hal-01067016.

[GRLC15] Galvane Q., Ronfard R., Lino C., Christie M.: Continuity editing for 3d animation. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence* (2015), AAAI'15, AAAI Press, pp. 753–761. URL: http://dl.acm.org/citation.cfm?id=2887007.2887112. 4

[HDKG08] Hachet M., Decle F., Knodel S., Guitton P.: Navidget for Easy 3D Camera Positioning from 2D Inputs. In *IEEE Symposium on 3D User Interfaces* (2008).

[HMK*97] Hong L., Muraki S., Kaufman A., Bartz D., He T.: Virtual voyage: interactive navigation in the human colon. In *SIGGRAPH '97: Proceedings of the 24th annual conference on Computer graphics and interactive techniques* (New York, NY, USA, 1997), ACM Press/Addison-Wesley Publishing Co., pp. 27–34. doi:10.1145/258734.258750. 2

[HW97] Hanson A., Wernert E.: Constrained 3D navigation with 2D controllers. In *IEEE Visualization* (1997), pp. 175–182. 1, 2

[KKS*05] Khan A., Komalo B., Stam J., Fitzmaurice G., Kurtenbach G.: HoverCam: interactive 3D navigation for proximal object inspection. In *SI3D '05: Proceedings of the 2005 symposium on Interactive 3D graphics and games* (New York, NY, USA, 2005), ACM Press, pp. 73–80. doi:10.1145/1053427.1053439.

[LCL*10] Lino C., Christie M., Lamarche F., Guy S., Olivier P.: A Real-time Cinematography System for Interactive 3D Environments. In *SCA '10 Proceedings of the 2010 ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (Madrid, Spain, July 2010), pp. 139–148. URL: https://hal.archives-ouvertes.fr/hal-00535865, doi:10.2312/SCA/SCA10/139-148. 2

[OSTG09] Oskam T., Sumner R. W., Thuerey N., Gross M.: Visibility transition planning for dynamic camera control. In *Proc. SIGGRAPH/Eurographics Symp. on Computer Animation* (2009), pp. 55–65. 2

[YLCL11] Yeh I.-C., Lin C.-H., Chien H.-J., Lee T.-Y.: Efficient Camera Path Planning Algorithm for Human Motion Overview. *Computer Animation and Virtual Worlds 22*, 2-3 (2011). 1, 2

[YLL*12] Yeh I.-C., Lin W.-C., Lee T.-Y., Han H.-J., Lee J., Kim M.: Social-Event-Driven Camera Control for Multicharacter Animations. *IEEE Transactions on Visualization and Computer Graphics 18*, 9 (2012). 1, 2

[ZSA03] Zancanaro M., Stock O., Alfaro I.: Using cinematic techniques in a multimedia museum guide. In *Proceedings of Museums* *and the Web 2003* (March 2003). URL: `citeseer.csail.mit.edu/zancanaro03using.html`. 1, 2