

Automatic Video Editing For Video-Based Interactive Storytelling

Edirlei Soares de Lima,
Bruno Feijó,
Antonio L. Furtado
PUC-Rio – Departamento de Informática,
Rio de Janeiro – Brazil
{elima, bfeijo, furtado}@inf.puc-rio.br

Cesar Pozzer
UFSM – Departamento de Eletrônica e Computação,
Santa Maria – Brazil
pozzer@inf.ufsm.br

Angelo Ciarlini
UNIRIO – Departamento de Informática Aplicada,
Rio de Janeiro – Brazil
angelo.ciarlini@uniriotec.br

Abstract—The development of interactive narratives with the quality of feature films is the central challenge of what we can name **Video-Based Interactive Storytelling**. A promising approach to this question is the use of prerecorded videos with real actors. Amongst several technical challenges, this approach firstly requires automatic video editing methods for interactive narratives. However, this is a critical issue not fully covered in the literature. In this paper, we present a real-time editing method for interactive storytelling systems, which automatically generates the most adequate shot transitions, swaps video segments to avoid jump cuts, and creates adequate looping scenes. Moreover, these features consider the characteristics of the ongoing story.

Keywords - video editing; interactive storytelling; virtual cinematography

I. INTRODUCTION

Since the advent of projected motion pictures in the late 19th century, cinema has been promoting new forms of immersive experience – from the silent black-and-white film to the high-definition stereoscopic 3D projection. Each one of these forms has been focusing on the imaging and auditory side of the experience. Following a different track, films with interactive plots have been recently proposed as a new experience. The area of interactive storytelling may provide the proper foundation for this new art form. However, this research area has few works oriented to motion pictures.

Interactive storytelling systems try to handle the unpredictability of events by using 3D virtual worlds to dramatize the stories [1][2][3]. This approach provides the freedom to move the characters and cameras to any place in the virtual world, allowing the system to show the story events from any angle or perspective. Furthermore, 3D characters may have their properties easily changed, such as shape, personality, emotional state, and intents. However, this freedom usually sacrifices the visual quality of the real-time narrative. The visual result of any current interactive

storytelling system that uses 3D worlds to dramatize the stories is far from the excellent quality of a computer animated feature film.

The development of interactive storytelling systems with the quality of feature films is a difficult challenge, even considering the new technology used by the videogame industry. A promising approach to this problem is the replacement of virtual 3D characters by video sequences in which real actors perform predetermined actions. This approach, which we call “video-based interactive storytelling” in this paper, has showed some interesting results in the last years [4][5]. However, most of those results are domain specific applications or do not have the intention of presenting a general approach to handle the problems of a generic video-based interactive story.

We claim that the task of developing a general video-based interactive storytelling system should firstly solve two basic problems: real-time video compositing and real-time video editing. In this paper we focus on the second problem only. We present a method based on cinematography principles to automatically edit segments of videos in real-time, while the plot is being generated by an interactive storytelling system. The proposed method is able to automatically select appropriate video shot transitions, avoid jump cuts, swap video segments, create looping scenes, and produce smooth and continuous video sequences. Therefore, our method goes beyond automatic techniques to choose the most adequate shot boundary effects. For this reason we present it as a video editing method. The paper is organized as follows. Section 2 presents previous works. Section 3 presents the architecture of our storytelling system. Section 4 presents our method to automatically edit the segments of video. In section 5, we analyze the performance and accuracy of the results to demonstrate the efficiency of our approach. Finally, in section 6, we present concluding remarks.

II. RELATED WORKS

Many works have already been done with the objective of applying videos as a form of visual representation of interactive narratives. Bocconi et al. [6] present Vox Populi, a system that generates video documentaries based on verbal annotations in the audio channel of the video segments. In a similar work, Chua and Ruan [7] designed a system to support the process of video information management, i.e.: segmenting, logging, retrieving, and sequencing of video data. This later system semi-automatically detects and annotates shots for further retrieval based on a specified time constraint. Ahanger and Little [8] present an automated system to compose and deliver segments of news videos. In a similar approach, but not focusing on news, Nack and Parkes [9] present a method to establish continuity between segments of videos using rules based on the content of the segments.

The idea of a generic framework for the production of interactive narratives is explored by Urso et al. [5]. The authors present the ShapeShifting Media, a system designed for the production and delivery of interactive screen-media narratives. The productions are mainly made with prerecorded video segments. The variations are achieved by the automatic selection and rearrangement of atomic elements of content into individual narrations. The system does not incorporate any mechanism for the automatic generation of stories. Essentially, their approach is to empower the human-centered authoring of interactive narratives rather than attempting to build systems that generate narratives themselves. A similar approach is used by Shen et al. [10], in which the authors present a video editing system that helps users to compose sequences of scenes to tell stories by selecting video segments from a corpus of annotated clips.

In a recent research work, Porteous et al. [4] present a video-based storytelling system that generates multiple story variants from a baseline video. The video content is generated by an adaptation of video summarization techniques that decompose the baseline video into sequences of interconnected shots sharing a common semantic thread. The video sequences are associated with story events and alternative storylines are generated by the use of AI planning techniques. Continuity issues are, however, not tackled by this approach. As these video segments can be jointed in different orders, several continuity failures may occur, in particular because their system uses video segments extracted from linear films. The AI planning algorithm ensures the story continuity, but not the visual continuity of the film.

Previous works focus mainly in the creation of stories by ordering video segments, without concerning about the way these segments will be connected. In this paper, we present a method for automatic video editing in video-based interactive storytelling applications. The method, based on cinematography theory, automatically selects appropriated transitions, avoids jump cuts, and creates smooth and continuous interactive video sequences.

III. SYSTEM ARCHITECTURE

The present work is part of the Logtell Project [11]. Logtell is an interactive storytelling system based on temporal modal logic and planning under nondeterminism. It uses a hybrid planner that combines partial-order planning and task decomposition to efficiently deal with nondeterministic events, *i.e.* events that can have more than one outcome. Logtell conciliates plot-based and character-based approaches [1] by logically modeling how goals can be brought about by previous situations and events.

For each class of character, there are goal-inference rules that provide objectives to be achieved by the characters when certain situations are observed. For example, within a fairy tale genre, if a victim V is killed by a villain L , a hero H will want to avenge the victim.

$$\square [(victim(V) \wedge villain(L) \wedge hero(H) \wedge killed(V,L) \Rightarrow \diamond killed(V,H))]$$

where \square and \diamond are the temporal modal operators “always holds” and “eventually holds” respectively; and \wedge is the logical symbol of conjunction. This logic framework permits the system to produce consistent temporal and spatial facts when they are required by a method of automatic video editing (*e.g.* the date and place of a particular action in the story).

The story is generated in chapters. In each chapter, goals to be achieved are specified either by rules or by user interventions, and the planner tries to achieve them. Situations generated by the planned events and user interventions that occur while the chapter is being dramatized influence the next chapter and so on. The chapters are represented as contingency trees, where the nodes are nondeterministic events and the edges correspond to conditions that enable the execution of the next event. A nondeterministic event e_i is executed by a nondeterministic automaton (NDA) composed by actions a_i (Figure 1). Our method of automatic video editing starts working upon the data embedded in this automaton. This automaton contains information about possible sequences of actions and is open to audience’s interventions.

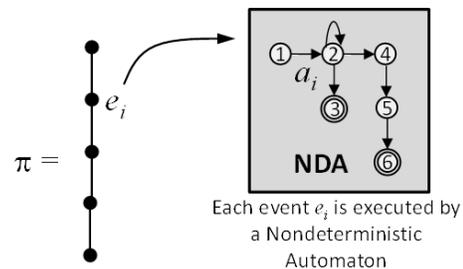


Figure 1. Plot π , events e_i , and nondeterministic automata with actions a_i . Double circles mean final states.

The audience can interact with the story by suggesting events to the planning algorithm (to be incorporated in the next chapter) and/or interfering in the nondeterministic

automata in a direct or indirect way. Details on plot generation and NDA specification in Logtell can be found in [11].

In the present paper, we are concerned with the proposal of a new video-based dramatization architecture that can be integrated with interactive storytelling systems. Any storytelling system can be used since it can provide the same sort of data found in Logtell, *i.e.*: (1) possible sequences of actions for an event; and (2) coherent spatial/temporal facts about a continuously-evolving story.

The proposed dramatization architecture (Figure 2) is composed by three cinematography-inspired agents: Director, Scene Composer, and Editor. The Director is responsible for interpreting the story events (planner output), retrieving videos from the remote Video Database, and also for controlling the dramatization of the story events. The Scene Composer, using real-time compositing techniques (such as chroma key) is responsible for combining the visual elements (video sources) that compose the scenes, creating the illusion that all the scene elements are parts of the same video. The Editor agent, using cinematography knowledge of video editing, is responsible for keeping the temporal and spatial continuity of the film. This later agent accomplishes its mission by automatically selecting the appropriated transitions between video segments, avoiding jump-cuts and presenting the story events through a smooth and continuous video sequence. In the Video Database, each action is associated with a coverage and a master scene video (a structure that is detailed in the next section).

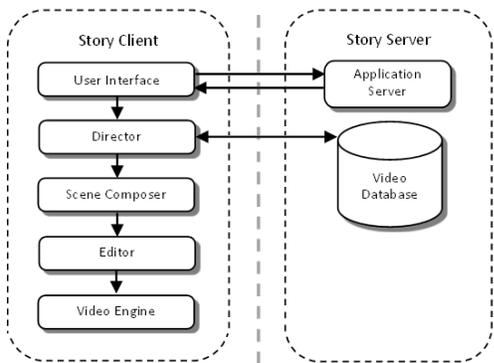


Figure 2. The proposed video-based dramatization architecture.

IV. THE CINEMATOGRAPHY EDITOR AGENT

A film can create its own time and space to fit any particular storytelling situation. Time may be compressed or expanded; speeded or slowed; remain in the present, go forward or backward. Space may be shortened or stretched; moved nearer or farther; presented in true or false perspective; or be completely remade in to a setting that may exist only on film [12]. Independently of all spatial and temporal awkwardness a filmmaker may purposely create in a film, it is always important to keep the spectator's mind oriented in time and space.

Cinematography defines a set of rules and principles that, when used correctly, ensure the basic spatial and temporal

continuity of a film. Some of these rules can be applied when the film is being shot and others during the editing process. Continuity editing is a set of editing practices that establish spatial and temporal continuity between shots and keep the narrative moving forward logically and smoothly, without disruptions in space or time [13]. Continuity editing is used to join shots together to create dramatic meanings. With an effective editor, the audience will not notice how shots of various frame sizes and angles are spliced together to tell the story. The best editing is usually the unobtrusive editing [12], that is, the one in which the audience does not notice that the editor joined the shots.

Our approach to create a computer program that is able to automatically edit video segments consists in translating cinematography principles and practices directly into logical rules. The main problem of using videos to dramatize an interactive narrative is the lack of freedom occasioned by immutable prerecorded segments of videos, especially during an automatic editing phase. For example, suppose the following conditions are detected: there are two video segments that cannot be joined together without sacrificing film continuity; both scenes are important to the story; and the order of the segments cannot be change. In this case, the only option that exists is to break the film continuity. Our solution to solve this problem is the use of the master scene filming method. In this method, we produce the master scene (a shot that includes the whole film set and uses a view angle that is different from other cameras) along with the coverage shots (shots that reveal different aspects of the action). In this way, the editor has always two shots of each scene. If the agent detects a problem in the coverage shots, a new shot of the same action can be extracted from the master scene (Figure 3). Furthermore, this method gives to the editor the freedom to creatively cut and alter the pacing, the emphasis, and even the point of view of the scenes. Filming a scene with the master scene method can be done using a single camera or multiple cameras [14]. The master scene method is used in probably 95% of narrative films shots today [14].

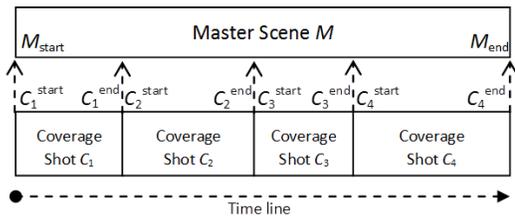


Figure 3. The master scene structure.

The basic purpose of the Editor agent is to join the selected video segments and keep the film continuity. A cinematography principle used by conventional editors to join and maintain the continuity between segments of videos is the use of adequate scene transitions. There are four basic ways to transit from one shot to another [15]:

- **Cut:** Consists of an instantaneous change from one shot to the next. It is most often used where the action is continuous and when there is no change in time or location.

- **Dissolve:** Consists of a gradual change from the ending of one shot into the beginning of the next shot. The dissolve is correctly used when there is a change in time or location, the time needs to be slowed down or sped up, and when there is a visual relationship between the outgoing and the incoming images.
- **Wipe:** Consists of a line or shape that moves across the screen removing the image of the shot just ending while simultaneously revealing the next shot behind the line or shape. The wipe is correctly used where there is a change in the location and when there is no strong visual relationship between the outgoing and the incoming frames.
- **Fade:** Consists of a gradual change from a fully visible image into a solid black screen (fade-out) and a gradual change from a solid black screen into a fully visible image (fade-in). The fade is used at the beginning/end of a film, scene, or sequence.

Another important principle of cinematography is obeyed when two segments of videos are joined together: cuts should be unobtrusive and sustain the audience's attention on the narrative. One way of complying with this principle is to avoid jump cuts [12]. A jump cut is often regarded as a mistake in classical editing [14]. It usually occurs when two very similar shots of the same subject (person or object) are joined together by a cut, producing the impression that the subject "jumps" into a new pose. A jump cut produces a disorientation effect, confusing the spectators spatially and temporally. There should be a definite change in image size and viewing angle from shot to shot.

We propose the use of a similarity scale to classify the transition between two consecutive shots and make the appropriate editing decision. Firstly, given two video segments C_x and C_y , we calculate the histogram C_x^h of the last frame of C_x and the histogram C_y^h of the first frame of C_y . Then we use a metric such as a correlation coefficient to determine the degree of similarity between shots:

$$F_{corr}(C_x, C_y) = \frac{\sum_i (C_x^h(i) - \overline{C_x^h})(C_y^h(i) - \overline{C_y^h})}{\sqrt{\sum_i (C_x^h(i) - \overline{C_x^h})^2 \sum_i (C_y^h(i) - \overline{C_y^h})^2}}$$

where $C_x^h(i)$ and $C_y^h(i)$ are the histogram values in the discrete interval i and $\overline{C_x^h}$ and $\overline{C_y^h}$ are the histogram bin averages. High values of correlation represent a good match between the frames, that is: $F_{corr}(C_x, C_y) = 1$ represent a perfect match and $F_{corr}(C_x, C_y) = -1$ a maximal mismatch.

Using the histogram correlation coefficient, we define three classes of similarity $\{S_1, S_2, S_3\}$. These classes can be expressed by the following rules:

If $F_{corr}(C_x, C_y) \in (\beta, 1]$ then similarity class is S_1 .

If $F_{corr}(C_x, C_y) \in [\alpha, \beta]$ then similarity class is S_2 .

If $F_{corr}(C_x, C_y) \in [-1, \alpha)$ then similarity class is S_3 .

The similarity scale and the classes of similarity are illustrated in Figure 4. The similarity class S_1 represents a class of high similarity between frames and no editing procedure is required. Transitions of videos with similarity S_1 are not noticed by the audience. The similarity class S_2 represents a condition that causes jump cuts. In this case, no shot transition can be applied and the Editor agent should use the master scene. The similarity class S_3 represents a class of low similarity between frames. In this case, the transition between videos can be done using a cut, dissolve, wipe or fade, without causing jump cuts.

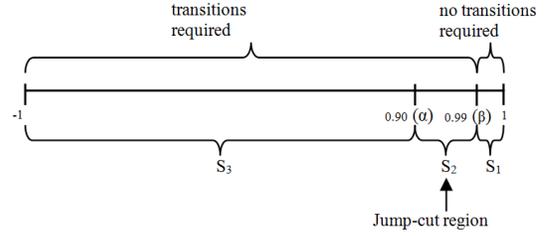


Figure 4. Similarity Scale (the values of α and β are experimental).

When a transition of two video segments is classified as similarity class S_2 , the video sequence must be changed to avoid the jump cut. As mentioned above, we adopt the use of the master scene method to give to the editor an extra video source of the same action shot with a different camera angle (Figure 3) – what guarantees an S_3 situation. In this way, when a transition is classified as similarity class S_2 , the editor can use the extra camera angle provided by the master scene video to show the same action without breaking the film continuity.

The transitions of video segments classified in the similarity class S_3 and the segments recovered from the master scene (class S_2) lead us to the second step of the editing process: the selection of appropriated transitions. To identify the most appropriated transition to join two segments of videos, we formulate a set of rules based on the cinematography literature to classify the video segments into the four basic classes of transitions. Considering the functions $L_T(x)$ and $L_S(x)$ that return the temporal and spatial locations of a video segment x based on its plot action chain, we define the following transition rules:

$$\forall C_x \forall C_y (L_T(C_x) = L_T(C_y)) \wedge (L_S(C_x) = L_S(C_y)) \Rightarrow T_{cut}$$

$$\forall C_x \forall C_y ((L_T(C_x) \neq L_T(C_y)) \vee (L_S(C_x) \neq L_S(C_y))) \wedge (F_{corr}(C_x, C_y) \geq 0.75) \Rightarrow T_{dissolve}$$

$$\forall C_x \forall C_y ((L_T(C_x) = L_T(C_y)) \wedge (L_S(C_x) \neq L_S(C_y))) \wedge (F_{corr}(C_x, C_y) \leq 0.75) \Rightarrow T_{wipe}$$

$$\forall C_x \forall C_y (\neg C_x \vee \neg C_y) \Rightarrow T_{fade}$$

Video segments classified as class T_{cut} are ready to be processed and the direct cut can be executed when the end of the first video segment is reached. The video segments

classified as class T_{dissolve} , T_{wipe} or T_{fade} must pass through another analyzer to determine the duration of the transition. In a conventional editing process, the editor usually uses the duration of a transition to represent the temporal variation that occurs during the transition. Based on this idea and considering t the exhibition time (usually in minutes or seconds) and T the story time (usually hours or years), the duration of the transition t_d is given by:

$$t_d(C_x, C_y) = t_{\min} + \left(\frac{(\Delta L_T(C_x, C_y) - T_{\max})(t_{\max} - t_{\min})}{T_{\max} - T_{\min}} \right)$$

where T_{\max} and T_{\min} represent respectively the maximal and minimal temporal variation in the story time, and the variables t_{\max} and t_{\min} represent the maximal and minimal duration of the transition. Usually $t_{\min} = 0.5$ and $t_{\max} = 2.0$ seconds (the minimum and maximal time for a transition, using dissolve transition as a reference).

Figure 5 illustrates the process of computing the transition between two video segments using the method above.

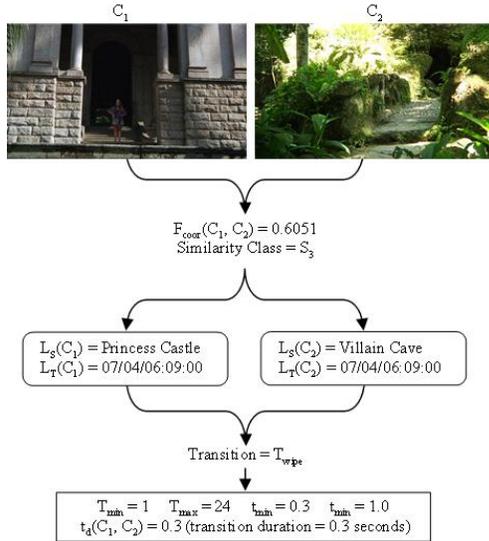


Figure 5. Example of a transition computation in our experimental comedy short film using Logtell [11].

The proposed method for automatic video editing is robust enough to handle looping states (such as the state 2 in Figure 1). These cases often fall within the S_2 similarity range and the Editor agent extracts an alternative video of the same action from the master scene video. In this way, the method can generate, for instance, a continuous fighting scene that gracefully switches between different viewpoints.

Also, we introduce a mechanism to guarantee variations in looping videos. When the same video is showed to the viewers for more than three times, the Editor agent computes a stochastic cut point in the coverage video to transit to the same point in the master scene video. These nondeterministic transition points generate more natural scenes and, consequently, loops in video are hardly noticed by the viewers.

V. EVALUATION AND RESULTS

The capacity of the proposed method to select appropriated transitions for video segments was evaluated through the following strategy: we compared the results of our automatic video editing method with the decisions made by human editors of well-known movies.

Firstly we analyzed the initial scene of the movie *The Lord of the Rings: The Return of the King* [16]. The scene starts with Déagol falling in the river Anduin and finding the ring, and ends with Frodo and Sam following Gollum through the Vale of Morgul. The test sequence had approximately 8 minutes and a total of 94 shots manually separated into individual video files (where the frames that contain transition effects were eliminated by hand). We chose this sequence because the scene starts in the past and gradually progresses to the present story time, enabling the evaluation of different temporal transitions. Given the ordered sequence of shots of the movie, we computed (for each consecutive shot) the transition between the shots (see Figure 6) and then compared the result with the original transition used in the movie. In this case, there is no need of a story planner, because the movie is linear. In our interactive storytelling system, the temporal and spatial information used by the algorithm is automatically calculated, but in this test we manually annotated this information in each shot. The result of the test is shown in Table I.

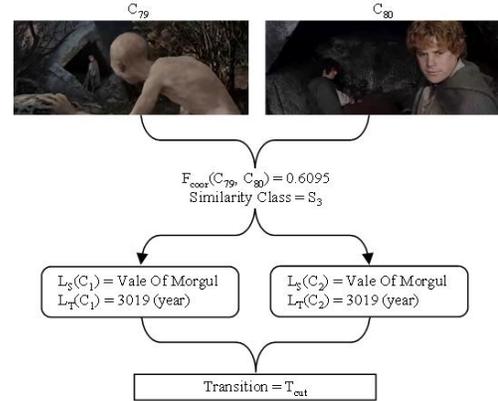


Figure 6. Example of a transition computation between two shots (C_{79} , C_{80}) of *The Lord of the Rings: The Return of the King*. Copyrighted images reproduced under “fair use” policy.

TABLE I. COMPARISON OF THE ORIGINAL TRANSITIONS IN *THE LORD OF THE RINGS: THE RETURN OF THE KING* WITH TRANSITIONS FROM OUR METHOD.

Transitions	Cut	Dissolve	Wipe	Fade
Original	86	6	0	2
Our Method	84	7	0	2
Hits Rate	97.6%	85.7%	100%	100%

We found two transitions that do not match with the original transitions. The first one is a cut classified by the Editor agent as a dissolve. Analyzing the video segments it is difficult to justify why the actual editor chose a cut, because

it is clear that the shots occur in different times and the cut causes some disorientation in the audience. This disorientation does not occur using a dissolve transition. In the other mismatch, the Editor agent classified the transition as a jump cut. Indeed, visually analyzing the film we see that there is a jump cut in a very short fighting scene. Actually we cannot affirm that the human editor is wrong in these cases, because each editor has his/her own style.

As a second evaluation test, we selected a movie by another director in a different film genre: the classic Psycho, directed by Alfred Hitchcock [17]. We analyzed the last scene of the film. The scene starts with Lila going to investigate Mrs. Bates' house and finishes in the end of the film, when Mary's car is pulled out of the swamp. The test sequence had approximately 14 minutes and a total of 153 shots manually separated in individual video files. The results of this test are shown in Table II.

TABLE II. COMPARISON OF THE ORIGINAL TRANSITIONS IN PSYCHO WITH TRANSITIONS FROM OUR METHOD.

Transitions	Cut	Dissolve	Wipe	Fade
Original	150	2	0	1
Our Method	150	2	0	1
Hits Rate	100%	100%	100%	100%

The automated video editing method presented in this paper does not involve complex computing tasks; however, its computing complexity grows according to the resolution of the analyzed video segments. To check the performance of the proposed method we tested its execution in the most common video resolutions. For each resolution, we computed the necessary time to get the image frames, computed the histograms, calculated the histogram correlation and classified the transition. This process was executed in a sequence of 40 video segments and the average time was computed for each resolution. We ran this test in an Intel Core i7 2.66 GHZ CPU, 8 GB of RAM using a single core. The results of the performance tests are shown in Table III.

TABLE III. PERFORMANCE RESULTS WITH DIFFERENT VIDEO RESOLUTIONS.

Resolution	704x480	1280x720	1920x1080
Time(ms)	21.268	29.455	59.170

VI. CONCLUSIONS

In this paper we propose a real-time method for automatic video editing to be used in video-based interactive storytelling systems. Our approach is based on translating cinematography editing theory directly into logical rules to automatically select appropriated videos transitions, avoid jump cuts, generate looping scenes, and create smooth and continuous video sequences. The evaluation and performance tests revealed the efficiency of the proposed method, although more extensive tests in a more dynamic context are still required.

As a further work we are planning some improvements in our video-based interactive storytelling system, such as the

use of video compositing techniques to manipulate the video segments and compose the scene elements in real-time. Another future work is the use of machine learning techniques for the implementation of different editing styles.

ACKNOWLEDGMENT

This work was partially supported by CAPES (Coordination for the Improvement of Higher Education Personnel, linked to the Ministry of Education) under grant RH-TVD 01/2007 No. 133/2008 and CNPq (National Council for Scientific and Technological Development, linked to the Ministry of Science and Technology) under grant PQ 304090/2009-3. Also we are grateful to the anonymous reviewers for the valuable suggestions.

REFERENCES

- [1] M. Cavazza, F. Charles, and S. Mead, "Character-based interactive storytelling," IEEE Intelligent systems, vol. 17, no. 4, pp. 17-24, July 2002.
- [2] M. Mateas and A. Stern, "Structuring content in the facade interactive drama architecture," Proc. of the Artificial intelligence and interactive digital entertainment conference, pp. 93-98, 2005.
- [3] A.E.M Ciarlini, C.T. Pozzer, A.L. Furtado, and B. Feijo, "A logic-based tool for interactive generation and dramatization of stories," Proc. International Conference on Advances in Computer Entertainment Technology (ACE 05), Valencia, pp. 133-140 (2005).
- [4] J. Porteous, S. Benini, L. Canini, F. Charles, M. Cavazza, and R. Leonardi, "Interactive storytelling via video content recombination," Proc. of the International Conference on Multimedia 2010, Firenze, Italy, pp. 1715-1718, 2010.
- [5] M.F. Ursu, I.C. Kegel, D. Williams, M. Thomas, H. Mayer, V. Zsombori, M.L. Uomola, H. Larsson, J. Wyver, "ShapeShifting TV: Interactive screen media narratives", Multimedia Systems, vol. 14, no. 2, pp. 115-132, 2008.
- [6] S. Bocconi, F. Nack, and L. Hardman, "Using Rhetorical Annotations for Generating Video Documentaries," Proc. of the IEEE International Conference on Multimedia and Expo, July 2005.
- [7] T.S. Chua and L.Q. Ruan, "A Video Retrieval and Sequencing System," ACM Transactions on Information Systems, vol. 13, no. 4, pp. 373-407, 1995.
- [8] G. Ahanger and T.D.C. Little, "Automatic composition techniques for video production," IEEE Transactions on Knowledge and Data Engineering, vol. 10, no. 6, pp. 967-987, 1998.
- [9] F. Nack and A. Parkes, "The Application of Video Semantics and Theme Representation in Automated Video Editing," Multimedia Tools and Applications, vol. 4, no. 1, pp. 57-83, 1997.
- [10] Y.T. Shen, H. Lieberman, and G. Davenport, "What's Next? Emergent Storytelling from Video Collections," Proc. of International Conference on Human factors in computing systems, pp. 809-818, 2009.
- [11] Logtell Project Web Site: <http://www.icad.puc-rio.br/~logtell/>.
- [12] J. Mascelli, The Five C's of Cinematography: Motion Picture Filming Techniques, Los Angeles, US: Silman-James Press, 1965.
- [13] B. Brown. Cinematography: Theory and Practice: Image Making for Cinematographers, Directors, and Videographers. Burlington, US: Focal Press, 2002.
- [14] J.G. Butler. Television: critical methods and applications. New Jersey, US: Lawrence Erlbaum Associates Publishers, 2002.
- [15] R. Thompson, Grammar of the Edit. Oxford, Focal Press, 1993.
- [16] New Line Cinema, The Lord of the Rings: The Return of the King, 2003.
- [17] Universal Studios, Psycho, 1960.