# A Cinematography System for Virtual Storytelling

Nicolas Courty[1], Fabrice Lamarche[2], Stéphane Donikian[3], and Éric Marchand[1]

[1] INRIA
[2] University of Rennes I
[3] CNRS
IRISA Rennes, Campus de Beaulieu, 35042 Rennes Cedex, France
Contact author Email {Donikian||Marchand}@irisa.fr

**Abstract.** In this paper we introduce a complete framework to automatically generate "cinematographic view" of highly dynamic scenes in real-time. The main goal of such a system is to provide a succession of shots and sequences (of virtual dynamic scenes) that can be related to pure cinema. Several major difficulties are to be considered. Above all, the real-time constraint makes impossible to consider scripted actions or planned phases that are common in real cinema (storyboards). Thus, one should consider, on one hand, the use of cinematographic rules (idioms) as a basis to describe camera behaviors and, on the other hand, camera control modules that can cope with dynamic scenes. Our system is based on the use of an image-based control of the camera that allows different levels of visual tasks and a multi-agent system that controls those cameras and selects the type of shot that has to be performed in order to fulfill the constraints of a given idiom. This capacity of adaptation constitutes the major novelty of our system. Moreover, it stands for a convenient tool to describe cinematographic idioms for real-time narrative virtual environments.

## 1 Introduction

Automatic cinematography can be seen as the ability to choose viewpoints (eventually animated) from which virtual environments are rendered, with narrative constraints. In this sense, a strong parallel with cinema can be drawn; camera shots and motions participate to the emotional commitment a user can get with narrative virtual environments. Since the beginning of the seventh art, film directors have implicitly established a set of rules and conventions that encode proper ways to tell stories in an accurate and explicit mode (see [2] for examples). Though, this set of rules is not sufficiently detailed to define a formal grammar which could be directly used in virtual storytelling applications. Indeed, given a complete three dimensional description of a scene that includes motion of various objects and characters, many possibilities exist to film with respect to the traditional way of filming. Moreover, we can assume that choosing among one of these possibilities depends on many factors (director's intentions, lighting purpose, disposition of the scene, etc.). Within this context, it appears to be a very awkward task to build such a system. Conversely, being able to shoot in an automatic way can be of great need for video games, virtual storytelling and virtual reality applications. This appears to be the next natural step of camera control problems.

In the contrary of cinema, when dealing with interactive environments such as video games or interactive fiction, it is not possible to use storyboards to characterize precisely the way each shot will be taken. In fact, due to the user interactions, the placement of cameras and actors can not be precomputed and, consequently, shots can not be planned in advance. Though it is always possible to set up offline some camera positions and switch between them at runtime (pre-composed “cut scenes”), this technique has proved to lack of expressiveness and forbids utilization of particular shots like over-the-shoulder shots for instance. This emphasizes the need of an autonomous cinematographic system in order to adapt to the dynamic world. In this paper, we introduce a flexible system allowing to encode in a generic way cinematographic rules while offering paradigms to deal with the real-time filming of narrative and interactive virtual environments.

## 2 Automatic cinematography

Many works have been done to automatically generate complete camera specifications, but among those works it is possible to distinguish two different types of approaches, considering or not real-time constraints. Hence, getting the exact flow of actions before they happen allows to run a planning process, which corresponds to the real cinematographic one [8, 4]. But, due to the user interaction, those approaches can not be used in real-time virtual environments as the flow of actions becomes unpredictable. This point constitutes the major difficulty in automatic cinematography, in the sense that the system has to generate a coherent and accurate viewpoint at each frame without knowing anything about the future of the actions. In [7], He *et al.* are the first to explore real-time automatic camera control. In [3], Bares uses a constraint-based approach for shots composition in a 3D interactive fiction. The FILM language[1] has two objectives: characterize the camera behavior of film idioms and adapt dynamically to the specific situation of any given shot. Funge [5] is one of the first to compare the camera system to an agent. Before the animation, a set of rules describing the camera behaviors are established, and during the animation its A.I. engine chooses between all those rules. Tomlinson [11] considered the camera as a creature, ruled by desires and intentions, that can act upon its environment (displacing lights for instance), in order to display the emotional content of the scenes.

## 3 Overview of the cinematography system

The main goal of our system is to provide one camera configuration for the 3*D* engine at each iteration of the application. Let us note that provided that the application would propose several view-ports of the virtual environment, the camera manager should give an equivalent number of camera configurations. Choosing the best camera configuration for the camera manager is performed through the use of information provided by other modules devoted to scenario, characters and environment management. This information can also be the result of direct queries. Those aspects will not be discussed in this paper. The camera manager is composed of a multi-agent system which controls several cameras that act, independently and with different goals, in the virtual environment.

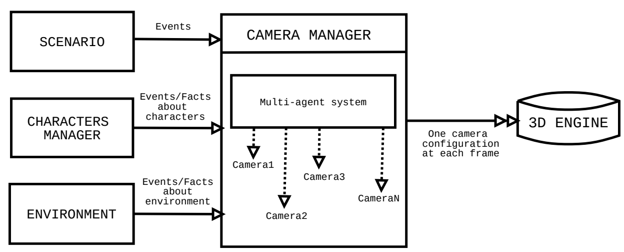


**Fig. 1.** Architecture of our system regarding to the other parts of the application

### 3.1 Camera Control

To perform positioning tasks and animation of the different camera, our system uses an image-based animation system, as proposed in [10]. Among other, this approach allows definition of multi-level tasks: a main task (or focusing task) that has to be respected at each frame, and some secondary tasks that can be executed providing that the main task is ensured. This feature allows flexibility in the control of the camera. Our camera module provides cinematographic primitives to control the camera. Several solutions may exist at the same time to solve a particular camera positioning task. Choosing between those solutions may depend on the desired effect as well as a particular context. Indeed, while evaluating the quality of a shot, the possible side effects due to the environment, like occlusions or lighting conditions, must be taken into account. In order to parameterize the different behaviors for the camera agent, the camera control module has to provide a lot of information about its status. Solutions to the detection of occlusions, and to possible strategies to avoid them exist [6, 10] and can be used efficiently in this context. Mainly, given an occlusion detection process, two possibilities are available: one can try to avoid the bad configuration, or if it is impossible (or if it results in a bad shot), the camera behaviors must adapt and choose for instance another shooting sequence. Using a camera control allowing different task levels is quite interesting in this context: while ensuring the nature of the shot (the image-based constraints), remaining degrees of freedom can be used to avoid bad configurations.

### 3.2 Modeling idioms and editing purposes

The language of film is a common international reference and it can be viewed as a structured style for conveying stories that is familiar to viewers. Even if stories of several movies are different, the related style is so structured that certain situations are almost filmed from the same way. This stereotypical way to capture specific actions into a predetermined sequence of shots is called a film idiom. Idioms are cinematographic rules which represent a codified way of chaining different points of view depending on the nature of the filmed scene. Starting from this assumption, an idiom can be interpreted as an abstract agent chaining shots on different points of interest in the scene. Focusing on those point of interests can be achieved in several manners from different viewpoints. For example, concerning a dialog between two characters, three different alternate kinds of view can be chosen (internal, external and parallel camera placements) and it is possible to pass from one type of view to another one by using the apex view of both actors (cf figure 2). Depending on both the duration and the location of a dialog, different sequences of shots can be obtained.

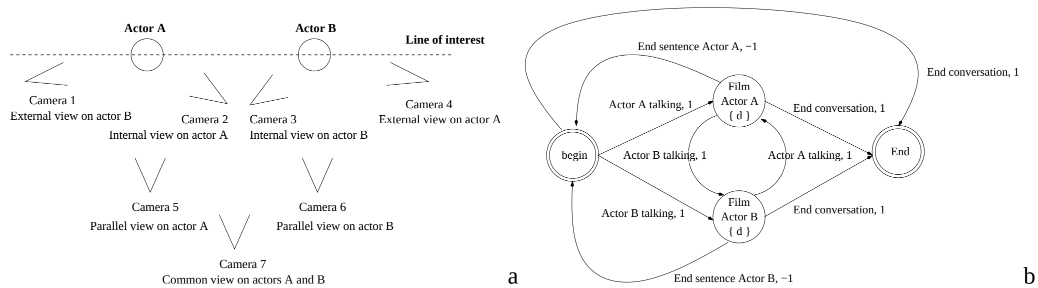


**Fig. 2.** (a) Possible camera placements for a conversation between two persons (b) Idiom template for a conversation between two actors. Transitions are labeled with their condition and degree of preference.

To adapt to the dynamic world, a multi agent system offering good synchronization properties [9] is used. The key concept of this architecture is to exploit redundancy of points of view satisfaction on a given scene in order to automatically select the best camera in respect with idioms description, knowing that each camera is described as an autonomous agent. The synchronization facilities offered by the multi-agent system are used to ensure consistency during the film, between the concurrent cameras but also between the running idioms. Agents are organized as a hierarchy of state machines. To allow an agent to manage concurrent behaviors, sub-agents are organized in sub-state machines. Thus, each state machine is either an atomic state machine or a composite state machine. Moreover, the notion of idiom can be abstracted in order to provide idiom templates. Those templates describe the abstracted rules consisting in chaining different points of view, filming different points of interest, on a given scene. The key idea is that for the purpose of description, the template does not need to know explicitly the set of cameras. The figure 2.b shows an example of state machine corresponding to the idiom template consisting in filming a conversation between two actors.

## 4 Results

Our example (cf Fig. 3) shows a simple dialog between two persons. The movement of the White character is controlled by a user, whereas the purple character is autonomous. The idiom used to generate this sequence is the one depicted in figure 2.b. For this example a simple ray-casting occlusions detection system was used to detect whether a shot was good or not. Since no occlusions were detected, the two characters are filmed with an over-the-shoulder shot (Camera1 and Camera4 in figure 2.a). In the images 3.a the two characters are standing still, but in the images 3.b the white character is moving backward toward the exit door. One can observe that the over-the-shoulder shot is realized even when the white character is moving. As the virtual primitives involved in the image-based constraint belong to the two virtual characters, we can assume that they will stay in the field of view. If an occlusion had been detected, this over-the-shoulder shot would have been transformed into a close-up shot (Camera2 or Camera3 in figure 2.a).

## 5 Conclusion

In this paper a complete cinematographic system adapted to real-time applications such as virtual Storytelling is introduced. The camera control is handled with our image-based constraints framework, and provides a multi-level control of the cameras (through the use of different level of visual tasks at the same time), and is well suited to avoid bad configurations (like occlusions) that are very likely to happen in dynamic environments. Our multi-agent system allows to exploit redundancy that can exist in the different ways of filming scenes, thus performing a trade-over between traditional ways and existing rules of cinematographic productions and dynamic environments constraints. This constitutes one of the major novelty in comparison with prior works. Moreover, this system constitutes a fast and intuitive tool to design cinematographic behaviors (idioms) that conveys narrative informations with respect to film language.

**Drawbacks and perspectives** Producing shots that can be related to real cinematographic ones still remains a difficult task. In movies, film directors may use some artefacts to enhance quality of conveying a story. For instance, it exists some meaningless actions that may not be shown, like going through a door, walking from one place to the other, etc. Those “compressions” of the time-line cannot be handled in a real-time application. Directors often position actors, or even adapt the environment to produce the best shot. This is not as well possible in real-time applications where a user expects a full control on what is going on. Moreover, lighting issues (classically called the “photo” problem) are quite important considering the resulting quality of a shot. For this problem, it is possible to enhance our cinematographic system with a lighting module that could position or manipulate different light sources. It also exists for this problem a set of conventions and rules that could be expressed as behaviors for a lighting module. Those aspects are part of the perspectives of our work.

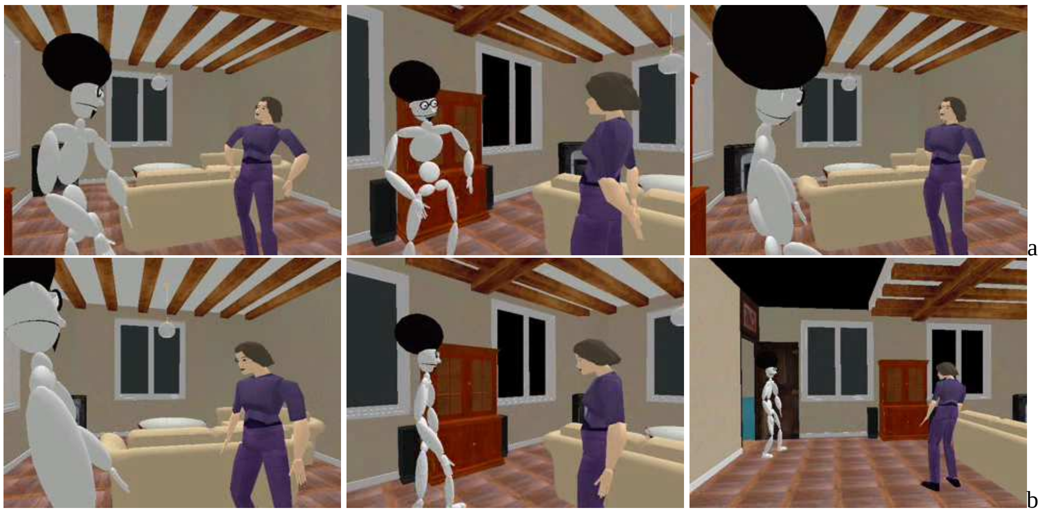


**Fig. 3.** Dialog between two characters (a) characters are standing still, talking with each others (b) the white character (controlled by a human user) starts moving back to the entrance door

## References

1. D. Amerson and S. Kime. Real-time cinematic camera control for interactive narratives. In *AAAI’00*, 2000.
2. D. Arijon. *Grammar of the Film Language*. Communication Arts Books, New York, 1976.
3. W.H. Bares, J.P. Grégoire, and J.C. Lester. Realtime constraint-based cinematography for complex interactive 3d worlds. In *Tenth National Conference on Innovative Applications of Artificial Intelligence*, pages 1101–1106, 1998.
4. D.B. Christianson, S.E. Anderson, L.W. He, D.H. Salesin, D.S. Weld, and MF.. Cohen. Declarative camera control for automatic cinematography (video). In *Proc. of the 13th Nat. Conf. on Artificial Intelligence and the Eighth Innovative Applications of Artificial Intelligence Conference*, pages 148–155, Menlo Park, August 1996.
5. J. Funge, X. Tu, and D. Terzopoulos. Cognitive modeling: Knowledge, reasoning and planning for intelligent characters. In *Proc. of SIGGRAPH 99, Computer Graphics Proceedings*, pages 29–38, Los Angeles, 1999.
6. N. Halper, R. Helbing, and T. Strothotte. A camera engine for computer games: Managing the Trade-Off between constraint satisfaction and frame coherence. In *Proc. of Eurographics’01*, pages 174–183, Manchester, UK, September 2001.
7. L.-W. He, M.F. Cohen, and D.H. Salesin. The virtual cinematographer: a paradigm for automatic real-time camera control and directing. In *Proc. of SIGGRAPH 96, in Computer Graphics Proceedings*, pages 217–224, New-Orleans, August 1996.
8. P. Karp and S. Feiner. Automated presentation planning of animation using task decomposition with heuristic reasoning. In *Proc. of Graphics Interface ’93*, pages 118–127, Toronto, Ontario, Canada, May 1993.
9. F. Lamarche and S. Donikian. Automatic orchestration of behaviours through the management of resources and priority levels. In *Proc. of Autonomous Agents and Multi Agent Systems AAMAS’02*, Bologna, Italy, July 15-19 2002. ACM.
10. E. Marchand and N. Courty. Controlling a camera in a virtual environment. *The Visual Computer Journal*, 18(1):1–19, February 2002.
11. B. Tomlinson, B. Blumberg, and D. Nain. Expressive autonomous cinematography for interactive virtual environments. In *Proc. of the 4th Int. Conf. on Autonomous Agents*, pages 317–324, Barcelona, Spain, June 2000.