AN AUTOMATIC REAL-TIME CAMERA CONTROL ENGINE FOR THE EXPLORATION OF ARCHITECTURAL DESIGNS

Carlos Calderon and Nicholas Worley
University of Newcastle upon Tyne
School of Architecture, Planning and Landscape
Claremont Tower, NE1 7RU
carlos.calderon@newcastle.ac.uk
n.j.worley@newcastle.ac.uk

Abstract

This paper is concerned with the use of real-time camera engines in architectural virtual environments as a method of enhancing the user's experience and as a way of facilitating the understanding of architectural concepts. This paper reports on an initial prototype of a real-time cinematic control camera engine for dynamic virtual environments in the architectural domain. The paper discusses the potential of the system to convey architectural concepts using well known architectural concepts such as rhythm and proposes a series of future improvements to address those limitations. Keywords: virtual environments, camera control, design process, filmaking.

1. Introduction

Manipulating the viewpoint or the camera control problem is fundamental to any interface which must deal with a three dimensional environment. This problem has been extensively researched in the computer science domain and a number of articles have discussed different aspects of the problem in detail [1][2].

In general, participants often have problems comprehending and navigating the virtual 3D environment [3] and, in particular, they struggle to perceive and understand the architectural concepts embedded in a design. For example, to experience rhythm in architecture is to observe "variations on a theme within a rectilinear pattern" [4] and, thereby, when you feel that a line is rhythmic means that by following with your eyes you have an experience that can be compared with the experience of rhythmic dancing. We are accustomed to perceiving architectural rhythm at a human perspective, that is, at walking pace and eye-level height. We believe that within a VE the limitation of having to navigate from "standard" camera modes (first person point of view; a particular character's point of view or from a free roaming mode) makes architectural concepts, such as rhythm, more difficult to comprehend and communicate. In fact, similar types of "communication" problems have been faced by cinematographers for over a century. Over the years, filmmakers have developed a set of rules and conventions that allow actions to be communicated comprehensibly and effectively.

This paper addresses the problem of communicating architectural concepts in 3D real-time virtual environments by creating a camera mode which incorporates cinematic principles. Basically, what we are trying to create with a new camera paradigm for walkthroughs is similar to what we are trying to create in an actual movie. We are trying to give the viewer/audience better understanding about the scene we are in. In order to achieve this, we are interested in defining a new camera mode which is "more useful" in the exploration of architectural concepts. In other words, most of the applications which use interactive 3D graphics as a platform are portrayed from a first person point of view, a particular character's point of view or from a free roaming mode but these camera modes are not always appropriate to facilitate the understanding of architectural concepts. Thus, we are trying to create a camera mode which gives the viewer a better understanding of the architectural design concepts.
Real-time control camera engines which incorporate cinematographic concepts face two main difficulties [3]: first, the encoding of informal rules of cinematography in a formal language and, second, a real-time camera control engine does not allow post-process editing. This means that whereas a traditional director works from an agreed script, a real-time camera control engine must deal with the events occurring in real-time the virtual environment to produce camera placements which can be seamlessly integrated into a meaningful sequence of shots and then into a "movie".

The paper is structured as follows: in the first section we introduce the principles and related work in camera engines; in the next section, we present our current implementation; and in the final part of the paper we report on future improvements.

2. Related work

Recent advancements in computer science have explored paradigms for automatically generating complete camera specifications in virtual 3D environments. Karp and Feiner [5] developed an animation-planning system that can customise computer-generated presentations for a particular viewer or situation. Christianson et al. [6] presented an interactive system which plans a camera sequence based on a simulated 3D animation script. All these techniques use an off-line planning approach to select the sequence of camera positions [3]. Off-line planning techniques take a pre-existing animation path and calculate the best camera placements. In this investigation, by contrast, we are concerned with real-time camera placement as the interactively controlled action proceeds. That is, systems which concentrate on finding the best camera placement when interactive tasks are performed. This type of systems were pioneered by Drucker et al. [7], they show how to set up optimal camera positions for individual shots. In our case, we are not only interested in real-time camera control system but also in incorporating cinematographic expertise for camera placement. He et al [3] were the first ones to present a paradigm for automatic real-time camera control which incorporated cinematic knowledge which could be used across application domains. We have used their implementation as the starting point for our system. Finally, it must be said that in the architectural realm initial steps have been taken to investigate ways in which filmmaking can be used for the development of *off-line* architectural animations [8] [9]. Both, Temkin [8] and Alvarado [9], argue that if we are to evolve beyond the average fly-through animation new ways of seeing and composing in time, which can be used to inform the process of architectural design, ought to be developed [8]. As previously explained, this investigation is concerned with finding new methods of communicating architectural design but in the realm of 3D real-time graphics which, in turn, poses a series of new challenges.

3. Example: tracking shot and rhythm

The following section attempts to illustrate how our initial prototype initial prototype (of a real-time cinematic control camera engine for dynamic virtual environments in the architectural domain) has been implemented by explaining a working example from the user's perspective.

3.1. A working example

We believe that realisation of architectural concepts emerges from the interaction of the players movements with specific areas of the 3D environment and that it can be enhanced by using an appropriate camera mode. The interaction of the user with the camera is not predetermined and only occurs through the events taking place in the virtual world. Let us illustrate this with an example.

Out aim in this first example is to illustrate the architectural concept of repetition (rhythm) to the user by using a cinematographic technique called the tracking shot [10]. The idea is that the user will approach the colonnade and upon entering an event is triggered in the VE which is recognised by the system and as a result, the new camera mode takes over. In this case, it simulates a tracking shot in which the camera is set along a perpendicular from the line of interest (the path that the user will takes as he travels through the colonnade) and then moves with the actor (the user's representation) maintaining the same orientation which keeps the actor near the center of the screen. A more detailed explanation of how cinematographic techniques can be encoded in the system is presented in the next section.

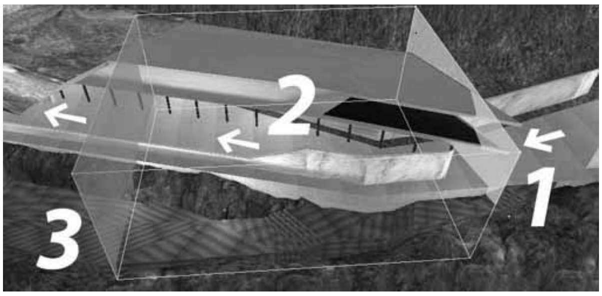

Figure 1: Overview of the example setup. Three areas can be identified: area 1 user's approaching colonnade; area 2 – within the volume boundaries- colonnade; area 3 user's outside colonnade

Figure 1 demonstrates the setup of the example. As the user approaches position 1 in Figure1 the act of the user entering the volume (the wire frame box in Figure1) is recognised by the system as an event. Figure 2a depicts the user's approaching the colonnade from a "standard" camera view: a third person point of view. Once the event has been identified, the camera moves into the tracking position shot (see Figure 2b) whilst the user remains within the volume boundaries (see position 2 in Figure1). Finally, the user's action of leaving the volume (see position3 in Figure 1) is recognised by the system as an event and the system, in turn, responds by returning the camera to a third person point of view (see Figure 2c). It must be noticed that with the volume defined and a camera position assigned to it the user can enter from any side and the camera will still execute the tracking shot position from the correct angle. This allows an environment in which the user can roam around with no predetermined direction while experiencing the architecture cinematically.

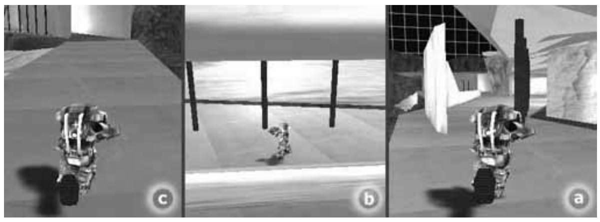

Figure 2: a) user's approaching colonnade; b) camera moves into tracking position; c) user's leaving colonnade

We believe that through the use of the cinematographic technique of the *tracking shot* the user views the virtual environment from a more meaningful perspective in the architectural sense and gains a better understanding of the architectural concept inherent in the building design. In our case this means that viewing the colonnade from a perpendicular perspective as the user walks through immediately draws the attention of the user to the 'variations on a theme within a rectilinear pattern'.

The full potential of this type of system derives from the user exploring the whole map and encountering different architectural concepts that are realised through different cinematographic techniques.

4. Conclusion and future work

In this paper, we have presented an initial implementation of a real-time camera engine in virtual environments as a method of communicating architectural design. Basically, our prototype can be seen as a method of encapsulating camera tasks, which follow the rules the cinematography (i.e. tracking shot), into will defined units called "camera modules" (i.e. tracking shot module). We therefore have presented the initial steps for a camera mode which we believe could give the viewer a better understanding of the architectural design concepts.

There are a number of areas for future work. For efficiency reasons, it seems that camera modules could be a robust approach for a real-time setting. Stereotypical formulas for capturing scenes as sequences of shot could be broken down into camera modules such as panning, following and so on. These then would have to be combined to produce a meaningful sequence. We are therefore currently experimenting with a more generic system architecture which incorporates the figure of a virtual director. That is, our virtual director would recognise events which occurred in the virtual environment in real-time and, from a library of modules (i.e. pan, follow, track, etc), select which camera module best fit. This would enable the system to seamlessly integrate shots into a meaningful sequence. We are expecting problems in correctly identifying unexpected occlusions as they happen in real time; encoding heuristics for good shot selection; and in translating a series of disjointed sequences into a movie.

Acknowledgements

We would like to thank Karl Nyman for helping us with the technical aspects involved in the development of this system.

References

1. Ware, C. and Osborne, S, Exploration and Virtual Camera Control in Virtual Three Dimensional Environments. Proceedings of the 1990 Symposium on Interactive 3D Graphics (Snowbird, Utah)1990.
2. Mackinlay, J. R., S. Card, et al. Rapid Controlled Movement Through a Virtual 3d Workspace. *Computer Graphics* 24(4): 171-176. 1990.
3. He, L. Cohen, M. and Salesin, D. The virtual cinematographer: A paradigm for automatic real-time camera control and directing. In Proceedings of the ACM SIGGRAPH'96, 217-224. 1996.
4. Rasmussen, Steen Eiler, Experiencing Architecture, MIT Press, pg 127. 1962
5. Karp, P. and S.K. Feiner. Issues in the automated generation of animated presentations. Graphics Interface '90. 1990
6. Christianson, D. B., Anderson, S. E., He, L., Cohen, M.F., Salesin, D.H., Weld, D.SDeclarative Camera Control For Automatic Cinematography". In proceedings of AAAI '96, 148-155. 1996.
7. Drucker, S. and Zeltzer, D. CamDroid: A system for implementing intelligent camera control. In proceedings of the 1995 Symposium on Interactive 3D Graphics*, 139-144.* 1995.
8. Temkin, A. Seeing Architecture with a Filmaker's Eyes. *In proceedings of ACADIA 2003: connecting-crossroads of digital discourse,* 2003
9. Alvarado, Rodrigo García. and Castillo, Gino Alvarez. Técnicas cinematográficas para las animaciones arquitectónicas. In proceedings of SIGraDi 2003*,* 2003.
10. Daniel Arijon. Grammar of the Film Language Communication Arts Books, Hastings House, Publishers, New York, 1976.