Ember, Toward Salience-Based Cinematic Generation

Bradley A. Cassell
North Carolina State University
890 Oval Dr Raleigh, NC 27606
bacassel@ncsu.edu

R. Michael Young
North Carolina State University
890 Oval Dr Raleigh, NC 27606
young@csc.ncsu.edu

Abstract

Automatic cinematic generation for virtual environments and games has shown to be capable of generating general purpose cinematics. There is initial work that approaches cinematic generation from the perspective of narrative instead of low level camera manipulation. In our work, we further extend this idea to take into consideration a model of user memory. Models of reader comprehension and memory have been developed to attempt to explain how people comprehend narratives. We propose using these models of narrative comprehension and memory to augment a system for cinematic generation so that it can produce a cinematic that can communicate character deliberation to the viewer by maintaining the salience of specific events.

Introduction

Cinematics are prevalent in our lives. We watch them on TV, in the movies, and in our video games. Most, if not all, of the cinematics in these domains are constructed by human directors, actors, cinematographers, and artists. Games, however, give rise to a new challenge of having custom cinematics based on player choices. To address this challenge, game creators typically create multiple cinematics beforehand and the system picks the appropriate one to display based on the user's choices in-game; however, with games becoming increasingly story driven it will become infeasible to construct all possible cinematics beforehand. A system that automatically generates cinematic sequences could solve this problem.

Most work in the area of cinematic generation has focused more on low level problems of camera placement and less on the cognitive and narrative effects on the viewer. However, human cinematographers either implicitly or explicitly build shot sequences to manipulate the viewer's mental state (Branigan 1992). Work by Jhala and Young (Jhala and Young 2010) viewed cinematics as a hierarchical discourse structure, much like the approaches used by the natural language generation community to model linguistic discourse (e.g. (Moore and Paris 1993; Mann and Thompson 1988)). Further improvement to this method can be achieved by augmenting it to use a viewer mental model directly.

Related Work

Camera Control

Current cinematic generation research falls into two main categories: 1) focusing on the low level problems of camera manipulation and 2) viewing generation as a narrative construction problem. Systems that focus on the low level problems typically use constraint solvers or intelligent agents (Christianson et al. 1996; Bares et al. 2000; Tomlinson, Blumberg, and Nain 2000; Elson and Riedl 2007). They usually use well known cinematic constraints, such as the rule of thirds or not crossing the line, to guide camera shot selection. Work by Christianson et. al. (Christianson et al. 1996) encodes cinematic idioms, or standard sequences of shots, into a declarative camera control language (DCCL). First, it uses a sequence planner to construct a scene tree for the action in the cinematic. It then uses DCCL to guide shot construction so that the resulting cinematic follows cinematic principles.

Tomlinson et. al. (Tomlinson, Blumberg, and Nain 2000) developed a system that uses an intelligent agent to film action within a virtual 3D space. The system constructs a cinematic by responding to events within the 3D space. Each virtual actor in the environment is able to tell the virtual agent how important it thinks it is and can request to be filmed. The virtual agent then responds to these requests and decides which character is the most important to film at any given time.

The system developed by Bares et. al. (Bares et al. 2000) uses user input in the form of storyboards to specify constraints on a 3D camera that films events in a 3D game. The system has a robust story boarding interface that allows artists to define constraints on the camera. These storyboards are then used by a constraint solver to determine where to place the camera in a real time interactive environment.

El Nasr (El-Nasr 2004) created a system which not only designs camera shot sequences, but also designs the entire visual experience. The system has four parts: a character system, a lighting system, a camera system, and a director agent. The three subsystems, character, camera, and lighting, all propose various actions to fulfill a communicative goal and the director agent unifies these into a consistent plan.

Systems that view cinematic generation as narrative con-

struction often use methods similar to those used in the natural language generation community to model linguistic discourse. Darshak (Jhala and Young 2010), for example, views cinematics as containing a hierarchical structure similar to how the work of Moore and Paris (Moore and Paris 1993) views linguistic discourse. This hierarchical structure uses abstract and base shots to encode cinematic principles. Darshak then uses a decompositional planner to create a shot sequence using these abstract shots and their decompositions into base shots.

Cognitive Models of Memory and Narrative Comprehension

Cognitive psychologists have developed models of narrative comprehension that attempt to explain how people understand narratives and store the information presented by them.

One model of human memory is the Attention Working Memory model developed by Landauer (Landauer 1975). In this model, information is said to be stored in a finite 3D lattice of locations. Each location is connected to the locations above/below, left/right, and forward/back of it. A pointer navigates this space randomly. As the pointer moves, new information encountered is stored at the location of the pointer.

If at any point information needs to be recalled, a search is done in the 3D space with the point's current location as the starting point of the search. The search is done up to a specified radius away from the center location. If the information is found, it is said to be salient. If it is not found, it is not salient.

This attention working memory model was used by Walker (Walker 1993) in her Design World system. Walker used this model as part of a system that could produce *informationally redundant utterances* (IRUs) in dialogs. IRUs are utterances that do not present new information in a dialog but are nonetheless present. Walker outlines several uses for these:

1. Providing evidence supporting mutual beliefs and acceptance (Walker 1993).
2. Making a proposition salient for the discourse participants (Walker 1993).
3. Supporting the beliefs that certain inferences are licensed (Walker 1993).

We draw a parallel between these IRUs and *informationally redundant shots*. These are shots in movies and cinematics that do not add any new significant information, but are nonetheless present in the cinematic discourse. Redundancy is often, but not always, used in film to indicate the focus of attention of a character. We specifically focus on the use of these redundant shots to make propositions salient for the discourse participants, or the viewers of the cinematic.

Narrative Discourse

Explaining character deliberation in film is more difficult than in written discourse. While in written discourse an author can just write out the character's thoughts or write an internal monologue these methods are generally not used in film. Directors and cinematographers often use special shots, called internal shots, to show that a character is thinking. In addition to these, other shots need to be used to present what the character is thinking about. Many times these shots show objects or elements in the world that are important to understanding what the character is thinking about. One special case of this is a shot that shows redundant information, or information that has already been presented to the viewer. In this case these special shots are used to foreground prior information about the story to help explain a character's thought process. This is often used in cases of character deliberation, or when a character is considering changing a course of action. This type of discourse element also occurs in written narrative discourse and conversations, as explained by Walker (Walker 1993).

System Requirements

Here we present Ember, a cinematic generator that creates cinematics that can maintain salience between shots that need it. We focus here on the generation of shots that specifically communicate a character's internal deliberation in the story. Often this involves the use of informationally redundant shots. Ember consists of three parts:

- Story structure passed in as input
- A Narrative Comprehension Model
- A Generation Algorithm

The story structure taken as input is a partial order causal link (POCL) (Weld 1994) plan data structure that represents the story world actions being filmed. This data structure is augmented in that it is required to contain instances of character deliberation within the story (described below).

The second part of Ember, the narrative comprehension model, keeps track of what viewers have seen and how salient, or more easily recalled, prior story actions are. This model is a simplified model of memory and recall that will be replaced with a more expressive model in future work. Currently, the model bases salience solely on how recent an event has occurred.

Finally, the cinematic generator uses the story plan and memory model to generate a cinematic sequence that can contain redundant shots to foreground important information about character deliberation. During generation, Ember uses the memory model to determine what elements about a character's deliberation are not salient for the viewer and then foregrounds them if needed. This process may create informationally redundant shots if the deliberation elements have been shown before in the cinematic. These three parts of Ember are explained in more detail in the following sections.

Character Deliberation in Story Generation.

Initial work on Ember has not focused on how to generate the input stories that contain instances of character deliberation. Instead it has focused on how to construct the cinematic generator given a story line that contains deliberation actions. However, some specifications have been designed

for the process of constructing the input story plans required by Ember. Specifically, the story planner will need to reason about character beliefs so that it can include instances of character deliberation. The resulting plans must contain instances of *deliberation steps*. A deliberation step is a step labeled as *Deliberate* and may have character beliefs as preconditions or effects. These steps are intended to model character mental acts of deliberation regarding choices they make.

Ember's Narrative Comprehension Model

The narrative comprehension model used in Ember is a simplified model of memory used to characterize aspects of the viewer's mental state. In this model, propositions that are communicated to a viewer by a camera shot are added one by one to the front of a list. The current focus is said to be at the front of this list. As propositions get added to the front, they push previously encountered proposition further away. When recall is needed, a check is done to see if the proposition being searched for is within a specific distance from the front of the list. This *salience length* is variable and is set before run-time. If a step is within the salience length of the front of the list, it is considered salient at the point of current focus.

Ember's Cinematic Generation Process

Cinematic generation in Ember uses the modified POCL story plan as input and constructs a camera plan that films the events in this story plan. The process used is similar to the way Darshak (Jhala and Young 2010) constructs cinematics. Camera shots are instantiated planning operators and can be abstract or primitive. Abstract camera shots have various decompositions that describe sequences of shots that accomplish the effects of the abstract shot. As camera steps get added to the plan they are linked to plot steps in the input story plan that accomplish what the camera steps are filming, thus linking the camera steps to the story steps they film.

Ember

Ember augments the methods Darshak uses to generate shot sequences to also include shots that explain character deliberation. Ember proceeds as Darshak does. However, when steps are added to the plan and they are linked to story plan steps, Ember checks to see if the story steps are deliberation steps. If so, Ember determines what information is needed to be salient for the viewer to understand the steps, checks if this information is salient using the simplified memory model, and then foregrounds it if it is not salient. To do this, two new constructs are defined for use during planning: *salience links* and *open salience precondition flaws*. The flaw maintains that no camera step, prior to the occurrence of the step the flaw is on in the camera plan, that films a story step that satisfies the precondition in the flaw, is linked to the step the flaw is on.

The planner can repair open salience precondition flaws by adding shot steps to the plan or using other steps already in the plan that fulfill the salience requirements of the step with the flaw.

Camera Planning Algorithm

Camera planning takes a partial Ember camera plan, a story plan, a domain, and a planning problem as input. The system then uses a modified version of Darshak's DPOCL-T algorithm to build a cinematic sequence. To begin, the partial Ember plan contains just the plan's initial and goal state steps. The goal state step has open precondition flaws on each of its preconditions.

Expanding the plan involves adding new camera steps to satisfy open preconditions. When a step is added to the plan, it may need information from other steps to be salient. In cases like these, salience flaws get added to indicate the need for a shot to establish salience. At each iteration, the planner can choose to repair these salience flaws in the current plan by inserting shots that foreground this required information or by creating salience links to prior shot steps.

When the planner chooses to fix an open salience precondition flaw it must insure that a camera step that has the literal in the flaw, $(Bel\ V\ q)$ for example, as an effect is salient at step s that contains the flaw. To do this the planner nondeterministically chooses either to add a new camera step with effect $(Bel\ V\ q)$ to the plan and add a salience link between it and s, or to add a salience link between a step already in the plan, that has effect $(Bel\ V\ q)$, and s.

When a step is added to the plan that would allow for more steps to occur between the steps of a salience link than are allowed by the specified salience length, that step is said to *threaten* the salience link. This is resolved much like a causal link threat is resolved in conventional POCL planners (e.g. (Weld 1994)). The offending step is either promoted after the end step in the salience link or demoted before the beginning step in the link.

Future Work and Challenges

Ember is still in its early stages of research. Several challenges still remain to be solved with this approach. First, the planner currently ensures salience for all preconditions in a deliberation step. However, it may be the case that not all preconditions are required to be salient. There may be a subset of non-salient preconditions that could be foregrounded for better results. In addition, Ember uses a shot for each precondition literal that needs to be salient. It may be possible to have a shot film more than one, or all, non-salient preconditions. The current memory model is also far too simplistic and needs to incorporate more than just recency in its calculations of salience.

Conclusion

Most cinematic generation systems are able to produce useful cinematics, but they lack the ability to reason about the effects the cinematics have on a viewer's mental state. In this paper, we have presented support for the use of a memory model during cinematic generation. Future work will strive to complete this system and test its ability to correctly generate shot sequences that describe a character's mental state while making decisions.

References

Bares, W.; McDermott, S.; Boudreaux, C.; and Thainimit, S. 2000. Virtual 3D Camera Composition from Frame Constraints. *Proceedings of the Eighth ACM International Conference on Multimedia* 177–186.

Branigan, E. 1992. *Narrative Comprehension and Film*. Routledge.

Christianson, D.; Anderson, S.; He, L.; Salesin, D.; Weld, D.; and Cohen, M. 1996. Declarative Camera Control for Automatic Cinematography. 148–155.

El-Nasr, M. 2004. An interactive narrative architecture based on filmmaking theory. *International Journal on Intelligent Games and Simulation* 3(1).

Elson, D. K., and Riedl, M. O. 2007. A Lightweight Intelligent Virtual Cinematography System for Machinima Production. In *Artificial Intelligence and Interactive Digital Entertainment*.

Jhala, A., and Young, R. M. 2010. Cinematic Visual Discourse: Representation, Generation, and Evaluation. *IEEE Transactions on Computational Intelligence and AI in Games* 2(2):69–81.

Landauer, T. 1975. Memory Without Organization: Properties of a Model With Random Storage and Undirected Retrieval. *Cognitive Psychology* 7(4):495–531.

Mann, W. C., and Thompson, S. A. 1988. Rhetorical Structure Theory: Toward a Functional Theory of Text Organization. *Text-Interdisciplinary Journal for the Study of Discourse* 8(3):243–281.

Moore, J., and Paris, C. 1993. Planning Text for Advisory Dialogues: Capturing Intentional and Rhetorical Information. *Computational Linguistics* 19(4):651–694.

Tomlinson, B.; Blumberg, B.; and Nain, D. 2000. Expressive Autonomous Cinematography for Interactive Virtual Environments. In *Proceedings of the Fourth International Conference on Autonomous Agents*, 317–324. ACM.

Walker, M. A. 1993. *Informational redundancy and resource bounds in dialogue*. Ph.D. Dissertation, University of Pennsylvania, The Institute for Research in Cognitive Science.

Weld, D. 1994. An introduction to least commitment planning. *AI magazine* 15(4):27.