Scene to Adventure Game

Project Proposal

Author: Ross Berkland
Supervisor: Shaun Bangay

Department of Computer Science
Rhodes University

27 February, 2009

1. Introduction

Recently a project was undertaken which involved building a text-to-scene system. The main function of this involves automatically extracting key concepts from the text of a fiction book and using these concepts to generate a computer animated movie. We now wish to extend this to create virtual environments depicting the realm and behavior described in the book.

This project investigates the techniques which can be used to convert the description given in these books to adventure games. More specifically, we will be looking at how the key aspects of the story can be incorporated into the game by annotating various parts of the text with tags/markers. Other tasks include enriching the game experience by including multiplayer and artificial intelligence.

In order to achieve this several problems will need to be resolved. The first of which will be creating an environment and its terrain in which we will be placing the various characters and objects.

We will also need to create a constraint system which handles the situation when constraints can affect on another in various ways. Another issue which we face is that of game goals also known as objectives or tasks. Unlike previous text-to-scene systems which simply animated the scene being described, we will need to create a way to add goals to the constraints. The system will have to generate triggers when users embark on new quests or complete any of those which are already active. It will also be necessary for us to create a way in which we can associate behavior with the individuals in the game. The characters will have behavioral descriptions about them which will need to be extracted from the text and implemented in the game.

While techniques for speaker identification have been created, it will be necessary for us to extract the dialogue from the book and include it into the game. Doing this will require that we know which character is speaking, what they are saying and when they are saying it.

The freely available GLEST engine will be extended to accommodate the creation of our game world. The content of which will be managed by extending the Blender 3d modeling package with custom plug-ins.

2. Motivation

In their paper on adventure game development, Moreno-Ger, Sierra, Martínez-Ortiz and Fernández-Manjón [6] mention how the process of creating a video game has become extremely lengthy and costly. The teams behind many modern games can include over 100 people and creating fully fledged adventure games has become a near impossible task for a individuals. By partially automating this process the creation of adventure games could become much easier and far less time consuming.

This will also give the user to explore those aspects of the story's realm which the book does not cover. In addition, user created content may be added to existing, well known stories.

3. Related Work

An important part of the project work will be defining a technique to annotate the text with tags which will then be used to incorporate story constraints into the game. The way in which text-to-scene conversion systems annotate text can be used as a good starting point.

In 2001, Bob Coyne and Richard Sproat of the AT&T Labs Research division published a paper on their text-to-scene system WordsEye. The paper mentions how WordsEye uses a database of 3D models and poses to depict the objects in the scene and their actions. According to Coyne and

Sproat [1] the system not only accommodated spatial relations but also actions performed by the objects in the scene.

Glass and Bangay [2] describe a method for converting fiction text to a time based graphical representation by using constraints acquired from annotated text. This concept was then expanded by Glass and Bangay [4] in their paper *Automating the Creation of 3D Animation from Annotated Fiction Text*. The paper describes how the constraint systems and time intervals used in Glass *et al.*, [2] can be adapted to automate animation assembly through use of the Blender modeling package.

Johansson, Berglund, Danielsson and Nugues [5] found further application for the text-to-scene concept and created CarSim, a system which uses text to visualize and animate traffic accidents.

Johansson *et al.*, [5] claim that the system is not restricted to inventive narratives.

It will also be necessary for us to know what is being said and by which character. Glass and Bangay [3] describe a strategy for speaker identification in their paper *A naïve, salience based method for speaker identification in fiction books*.

Another important aspect of the project is creating a quality adventure game. Moreno-Ger *et al.*, [6] describe the process which one could take in creating an adventure game in their paper *A Documental Approach to Adventure Game Development.*

4. Intended Approach

It is quite possible that several of the following tasks will lend themselves to one another requiring that they be done in parallel. For this reason the following order is not final.

4.1. Annotation

We will begin by analyzing the most relevant and efficient way of annotating the text in terms of the project. Constraints will play a large role in the final outcome of the game and it is therefore necessary to have an annotation method which will accompany the various aspects of the game. This includes the characters, the map and its various regions, important objects in the world, dialogue and the events which characters are involved in.

Glass *et al*. [4] specify a constraint system which we will use as our starting point.

4.2. Familiarity with the game engine

It will be necessary to familiarize ourselves with the engine being used. This will be done through several practical tests and an investigation of the GLEST manual.

4.3. Creating the world

Before we can implement the characters and constraints, it is necessary to create the game world. For test purposes, we will start with a generic game world which will serve as a foundation for further development. This will be done with the help of the GLEST map editor.

4.4. Managing the constraints

Once the constraints systems have been specified, we will need to create a way in which to manage them. We will create a constraint system which takes into account the interval (time) of the event occurring, as well as the objects and characters involved. These constraints will have to be incorporated with the game engine. This will be done by attributing various classes and objects with various elements of the constraints. For example, the behavior which the text describes for a particular character will have to be incorporated into the characters properties.

4.5. Implementation

Incorporating the constraints will possibly be the largest and most time consuming task to be done. Manipulating the game engine to accommodate the constraints will require the creation of new data structures and classes which can be used to abstract the various objects, events and timing associated with those events. We will also need to interface the blender plugins responsible for managing certain game content with the engine.

4.6. Possible Advancements (Additional game value)

If time allows we would like to incorporate artificial intelligence and multiple players into the game. Fortunately, both of these are already included in the GLEST engine to some degree. However, it will be necessary to several changes.

We will have to create a method for attributing the computer controlled players with those character details specified in the book. We will test various methods to accomplish this such as annotating the character descriptions with tags which can be incorporated into the NPC (non-player character) class.

In order to accommodate multiple players we will look at various methods for ensuring that the text which was used to describe the game is consistent amongst each of the clients.

5. Evaluation

The primary goal of the system is quite simply to use the input text to create a game world which is consistent with that described in the book. We require that the game world is populated with all of the characters and objects in the book along with the various goals which will define the story and its progression. The player must be able to take part in the story and enroll in and complete various quests.

Our secondary goal is that of artificial intelligence implemented into the game. We require that the NPC's accurately depict the characters and the way in which the book describes their behavior.

Finally, we would like to see multiplayer functionality in the game which should allow various users to play in the same game world at the same time.

6. Outcomes

By extending the existing text-to-scene technology, we would like to create a system which has the ability to generate a rich game world populated with characters, objects and places from the given text. The events in the game should follow those mentioned in the text but should also allow the user to use his/her own initiative in completing tasks.

On successful completion of implementation we would like to incorporate the ability for multiple users to exist in the game world along with non-player characters.

References

[1] Coyne, B. and Sproat, R. *Wordseye: an automatic text-to-scene conversion system*. In Proceedings of the 28th annual conference on Computer graphics and interactive techniques. New York, USA, pages 487–496, 2001.

[2] Glass, K., and Bangay, S. *Constraint-based conversion of fiction text to a time-based graphical representation*. In SAICSIT '07: Proceedings of the 2007 annual research conference of the South African institute of computer scientists and information technologists on IT research in developing countries, pages 19-28, Fish River Sun, South Africa, October 2007.

[3] Glass, K., and Bangay, S. *A naive salience-based method for speaker identification in fiction books*. In PRASA 2007: Proceedings of the 18th Annual Symposium of the Pattern Recognition Association of South Africa, pages 1-6, November 2007.

[4] Glass, K. and Bangay, S. *Automating the creation of 3D animation from annotated fiction text*. In Proceedings of the IADIS International Conference on Computer Graphics and Visualization 2008, pages 3-10, July 2008.

[5] Johansson, R. et al. *Automatic text-to-scene conversion in the traffic accident domain*. In IJCAI-05: Proceedings of the 19th International Joint Conference on Artificial Intelligence. Edinburgh, Scotland, pages 1073–1078, 2005.

[6] Moreno-Ger, P. et al. *A Documental Approach to Adventure Game Development*. In Science of Computer Programming 67(1). 3-31. 2007 A Documental Approach to Adventure Game Development. Science of Computer Programming 67(1), pages 3-31. 2007.