A Framework for Simulation-Event Triggered Animations

R. Hauck[1] and U. Künzler[1]

[1]Bern University of Applied Sciences, Switzerland

Abstract

We present a generic framework which uses a CAVE virtual environment for the optimization of surgical workflows and operation room medical equipment arrangements. The framework combines an event-driven simulation with 3D character animations and focuses on the separation of the workflow logic from the geometrical representation of the 3D objects.

Categories and Subject Descriptors (according to ACM CCS): I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism —Animation I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism —Virtual reality

1. Introduction

Animated characters are an essential element in realistic 3D environment visualizations and in particular for an application such as our medical workflow simulation.
In the last few years, the use of information technology for surgical procedures enabled strong improvements in medicine. These improvements necessitate specialized personnel and need more technical devices, which lead to less space in an operation room. We developed a generic framework in the so-called *HOVISSE* project [KAE*09], which aims to organize the available space to adapt it optimally for a specific surgery. This is done by using a virtual 3D representation of an operation room instead of the today used 2D floor plans, which have difficulties considering spatial aspects as well as with identifying medical screen occlusion. The pre-operative planning of the positioning of the operation room equipment can be interactively optimized through the simulation of the operation room setup and intra-operative surgery workflow within a CAVE environment.

2. Framework architecture

The framework we developed is divided into several modules. It is driven by a workflow description, consisting of a set of actions, which refer to actors and objects. The main objective of this generic framework was to separate the program logic and the surgery workflow description from the geometrical representation of the referred actors and objects. This allows us to optimize these two entities independently. Therefore, the module with the program logic, the workflow engine, has no information about the exact location of persons or objects in the operation room; it just assumes they are part of the simulation. The workflow engine is responsible for the execution of a surgery workflow. This workflow definition is divided into different phases, which consist of high-level actions; the logical sequence is specified as a predefined, unordered set of actions, where every action has certain pre- and post-conditions. These conditions define the resulting execution sequence of actions in the workflow description. Once the workflow engine has determined the valid actions, they are passed by an XML-based network protocol to the simulation engine.
The simulation engine is responsible for the visualization of the operation room, which includes medical actors, visualized through animated characters, and medical equipment. The visualization is realized through our graphics software framework, developed mainly in C++ and OpenGL. Its main component is a scene graph, which is controlled by the simulation engine.

3. Simulation engine

The simulation engine controls several core components of the surgery workflow simulation such as the event management, the interaction of the user within a CAVE and the animation handling.
It applies the discrete event-based simulation technique. Each event corresponds to an action of the surgery workflow, sent by the workflow engine. These events are processed by

the animation module, a sub-module of the simulation engine, which looks up the respective 3D animation. The simulation engine executes the events, i.e. animating actor behavior or movements, or playing a 3D animation for medical equipment. Each 3D animation is assigned to an actor or object.

At any time, the simulation can be stopped to change the positioning of the objects and actors in the operation room and to restart the simulation afterwards. This allows the medical experts to influence the geometrical layout and to test different approaches to find the optimal arrangement. For this, the simulation engine has to provide a mechanism to store and reload specific spatial positions of the operation room equipment. This is done by traversing the scene graph using the visitor pattern and storing all relevant parameters for different phases of the surgical workflow or for a user defined breakpoint. Through this mechanism, the operation room arrangement can be optimized at each particular stage of the workflow. After the medical experts have changed the scene, they can replay this particular phase with the new conditions. The simulation engine then again traverses the scene graph and restores the previously saved positions. At the same time, the workflow engine is notified to restart at the given action in the workflow.

4. AI based animation handling

Animations of actors are an essential element of realistic simulations. For this, the simulation engine contains the animation module that controls all aspects of the 3D animations in the operation room simulation. Actions received from the workflow engine are converted into events that are passed to the simulation engine. For this, each action, which is assigned to either an actor or a medical device, is inserted into an event queue. This queue is processed every time the simulation engine raises an event to update the scene graph and to redraw the frame. Such an update step includes several tasks. First, if the current event is a move event, i.e. an actor moves from one location to another, the animation module has to calculate the path. For this, AI techniques based on Dijkstra's algorithm [Dij59] pre-calculate the resulting path, taking into account the position of other actors and medical devices simultaneously. This pre-calculation is only executed when a new event is received or if during the simulation two actor movement paths intersect and would lead to a collision. In this case, the simulation engine suspends the next steps for these actors and recalculates each path.

The result of the path planning algorithm is a sequence of positions, which lead to the targeted location. The simulation engine retrieves them and queries for each redraw of the frame the new positions of the actors. These positions are defined by the pre-solved path sequence and the difference between the current and the last iteration time.

Next, the corresponding 3D animation has to be determined. The animation module can look up the referenced animation in a catalogue, in which every 3D animation is mapped to an object. Using this reference, the animation module plays the 3D animation using the free 3D character animation library Cal3D [LD]. To synchronize all animations, the 3D character animations use the same calculated time difference between the current and the last iteration as the path finding algorithm uses for calculating the movement of the actors.

5. Results and future work

We have developed a generic framework that uses an innovative approach to separate the program logic and the workflow action handling from its geometrical representation. The use of 3D character animations supports a virtual surgery simulation by a more realistic visualization. Using the example of a surgical workflow, we showed that the optimization done in a CAVE is easier to evaluate and has several advantages compared to a traditional desktop system. Medical experts can stand in the virtual operation room and see the medical objects on a scale of one to one. Using head tracking, the point of view can be adjusted instantly by the user through changing his position. Additionally, the possibility to optimize the arrangement of operation room medical equipment pre-operatively achieves lower operation costs.

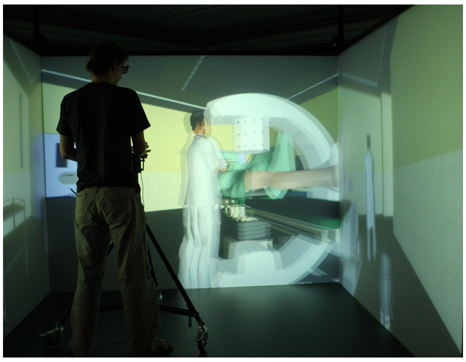

Figure 1: *Running the surgery workflow simulation in a CAVE using stereoscopic projection.*

Our future work will advance the framework to include an autonomous path planning technique, so that actors are able to move freely in the virtual environment in an intelligent manner instead of on a pre-calculated path.

References

[Dij59] Dijkstra E. W.: A Note on Two Problems in Connection with Graphs. *Numerical Mathematics 1* (1959), 269–271.

[KAE*09] Künzler U., Amrhein B., Eckerle J., Fischli S., Hauck R., Hoigné D., Witschi R.: Hovisse — haptic osteosynthesis virtual intra-operative surgery support environment. *Human Machine Interaction: Research Results of the MMI Program* (2009), 193–220.

[LD] Laurent Desmecht Loic Dachary B. H.: A skeletal based 3d character animation library. http://home.gna.org/cal3d/.