GALA 2007 submission document

Title:	Intelligent Character in a Shooting Game
Authors:	Chun-Chieh Chen Master Student, CS Department, (Graduated July, 2007) National Chengchi University, Taiwan Tsai-Yen Li Professor, National Chengchi University, Taiwan
Contact data:	**e-mail:** g9425@cs.nccu.edu.tw **url:** *http://imlab.cs.nccu.edu.tw* **phone:** 886-2-29387170 **fax:** 886-2-29390414 **postal address:** *Computer Science Deparmtent, National Chengchi Univ. 64, Section 2, Zhi-Nan Rd., Taipei, Taiwan*
URL:	http://imlab.cs.nccu.edu.tw/~jckim/project3/
Movie file submitted:	*chen.avi*
Reference teacher:	Prof. Tsai-Yen Li **affiliation:** Computer Science Department, National Chengchi University **e-mail:** li@nccu.edu.tw **phone:** 886-2-29387642
Process of development:	The system was developed in 2007, as part of the master thesis by the first author.
Resources used:	MS DirectX 9 library for graphics and planning programming, CMU motion library (http://mocap.cs.cmu.edu/)
Resources required:	The interactive system was developed and tested on a regular PC with MS Windows OS and 1GB of RAM.

GALA 2007 submission document

Intelligent Character in a Shooting Game

1. The application and context of the work

Realism and interactivity are always tradeoffs in real-time computer animation. In most 3D games today, due to the limited computing power available on a target platform, designers usually need to sacrifice the flexibility of the motions of the characters in the game to provide real-time interactivity. For example, most of the digital characters today, when interacted with the user, can only display canned motions triggered in a preprogrammed situation. The system has to constrain the motions of the character to some areas of the given environment that have been considered beforehand and therefore restrict the flexibility of how a character can react to a dynamic environment or to the user.

As the computing power increases because of hardware development, we think we now can invest more computing cycles to increase the interactivity of the digital actors with the user in a game. We consider that an ideal digital character in a highly interactive game such as shooting game should be capable of generating its own body motions on the fly according to the interactions with the user. In other words, the character should be smart enough to figure out how to avoid the user's threatening actions such as shooting. If such an intelligent character is made available, interesting interactive games with more dynamic contexts should be able to emerge and bring the fun of interaction to a new level. In this work, we aim to design an interactive system that allows the users to interact with such an intelligent character. We use a shooting game to demonstrate the interactivity of the digital character with the user.

2. Novelty

Human characters have been widely used in the entertainment industry such as various types of 3D games. Most recent researches in real-time human animation take a data-driven approach [3][5][6] since the captured motions in this approach are the most natural. However, the motion-captured data also have the problem of less flexible for adapting to different environments. In this work, we attempt to make good use of existing motion libraries and synthesize them in real time according to the interaction with the user.

In order to help the digital character to select appropriate motion clips from the motion library, we have to design a real-time system that can maintain the available collision-free motions that the character can use in the near future. We have design a structure called Feasible Motion Tree (FMT) in real time by traversing the motion clips organized in a motion graph [4]. FMT represent the feasible (collision-free) motions that are available for selection at the given situation including information such as character position, gun position, and environmental obstacles. We have developed an effective strategy to traverse the motion graph and maintain the FMT such that the most useful motions can be checked first. We use a time-budgeted approach to maintain FMT in an incremental fashion such that the computation for maintaining this tree can be distributed and incorporated into every frame of the control loop.

3. The architecture

The system takes keyboard or mouse inputs from the user and converts them into high-level inputs as in most 3D games. The system checks collisions of the full body of the smart character with the obstacles (including the bullets fired by the user) and attempts to generate appropriate motions to avoid potential collisions. The planning and interactive system has been fully implemented based on the Microsoft DirectX graphics library. The motion library and motion graph were built based on the motion capture library provided by CMU[2]. Detail description of the architecture can be found in [1].

4. Performance

The animated character can interact with the user and be animated in real time despite the high computational complexity of collision checks and motion planning. For environments consisting of thousands of polygons, the system still can maintain real-time frame rate of over 30fps. In fact, in our budget-based approach, the frame rate can be set by the user and the system will make good use of the remaining CPU cycles after graphics rendering to perform the planning. Therefore, the system performance is scalable for computers with different computing powers.

References

[1] C.-C. Chen, T.-Y. Li, "Intelligent Third-Person Control of 3D Avatar Motion," *Proc. of the 7th International Symposium on Smart Graphics*, Japan, June 2007.

[2] CMU Graphics Lab Motion Capture Database, http://mocap.cs.cmu.edu/

[3] T.H. Kim, S. I. Park, and S. Y. Shin, "Rhythmic-Motion Synthesis Based on Motion-Beat Analysis," *ACM Transactions on Graphics*, 22, 3, pp. 392-401, 2003.

[4] L. Kover, M. Gleicher, F. Pighin, "Motion Graphs," in-*Proc. of SIGGRAPH 2002*, 2002.

[5] J. Lee, J. Chai, P. Reitsma, J.K. Hodgins, and N. Pollard, "Interactive Control of Avatars Animated with Human Motion Data," *Proc. of SIGGRAPH 2002,* 2002.

[6] J.J. Kuffner Jr. *Autonomous Agents for Real-time Animation*. PhD thesis, Stanford University, Stanford, CA, December 1999.