Combining manipulation and navigation in virtual environments

B.J.H. van Basten
Center for Advanced Gaming and Simulation, Utrecht University, the Netherlands, basten@cs.uu.nl

Introduction

Character animation plays an important role in games, simulations and movies. Unfortunately, creating animations from scratch is a very tedious job and the resulting quality depends on the skills of the animator. Therefore a lot of research has been done on automatically generating realistic animation to assist the animator.

Although current algorithms are able to generate a wide range of motions, most of them do not take human behavior into account. Consider the following task: picking up a cup from a table while walking passed it. The resulting animations from current algorithms look far from natural because they do not consider factors that influence the path and animation. For instance, the end position of the path might determine which hand will be used for picking up the cup.

In an ideal situation, we would only like to instruct our character *what* to manipulate: our algorithm will generate a realistic path to the object followed by a realistic animation of the manipulation. So, our research is two-fold. We conduct experiments in our motion capture lab to determine a model that yields natural path and postures given specific situations and we develop algorithms that generate animations that are conform this model.

Related work

One of the best-known algorithms for motion synthesis (generation) are the *motion graph* [1] approaches. A motion graph is a directed graph that encapsulates the ways existing motion clips can be re-assembled into new motions. It contains original motion clips (for instance retrieved by motion capture) and generated transitions. These transitions connect two clips of original data by blending from one frame to another frame. In order to automatically determine these transitions several distance comparison metrics can be used to determine to what extent frames resemble each other. Then, after the motion graph is created, a graph walk can be applied to generate a new motion. Many motion graph variations exist that mainly restructure the graph and/or apply better search algorithms to synthesize motions more efficiently.

Unfortunately, the generated motions do not always fit our needs because a target object that we want to pick up might be at other positions than in the concatenated motions. By using motion interpolation [2] techniques we can generate picking up motions that correspond to the target object.

Recently, several methods have been developed that combine motion concatenation (such as the motion graphs) and motion interpolation. Heck *et al.* introduce *parametric motion graphs* [3]. Continuous "spaces" of motions are constructed that are connected in a graph-like structure. These spaces are basically comprised by a set of resembling motions that can be parameterized by interpolation techniques.

Although parametric motion graphs result into a continuous stream of parameterizable motions, it is still not possible to combine motions naturally, such as needed for walking past a table and picking up a cup. To address part of this problem, Heck *et al.* [4] present a technique to attach (splice) a lower-body motion with an upper-body action. Unfortunately, the upper and lower body motions are still fixed and except for some alignment between body parts, human behavior is still

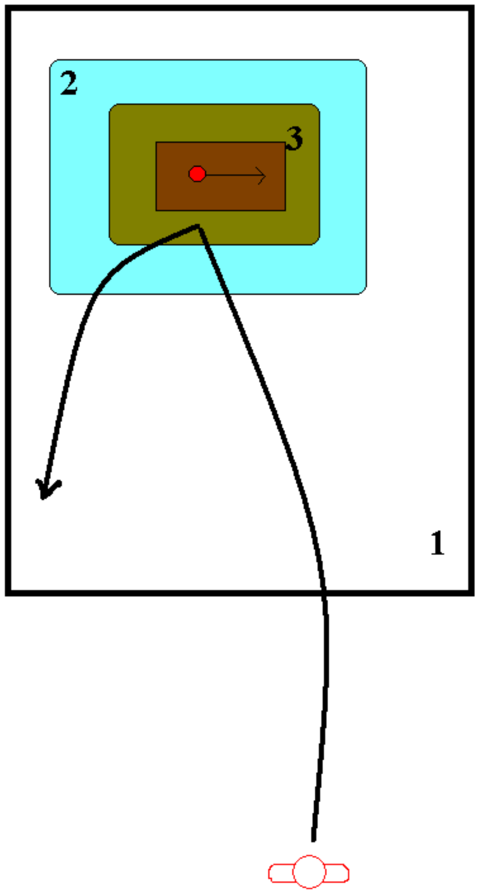

Figure 1. *Several zones can be distinguished in tasks consisting of navigation and manipulation*

not taken into account. Shapiro *et al.* [5] also split the problem into two parts. One part (the locomotion) is fixed; the other part (manipulation) is planned on top of the lower body locomotion. They use path planning techniques to plan the arm movement to manipulate or avoid objects. The locomotion, however, is still fixed and dependencies between the manipulation and navigation are not taken into account.

Model

When we consider a combination of manipulation and navigation such as depicted in Figure 1 (the character walks to the brown table, moves the red cup to the right and walks back) we can distinguish several zones.

At first, the character walks globally towards the table, when entering zone 1 (*navigation* zone) he adapts his path according to the task and goals he will have in the future. When entering zone 2 (*preparation zone*), the character will prepare his manipulation task, such as already reaching out. In the third zone, the character will perform the actual manipulation.

Many parameters will influence this task. The shapes of these zones and the shape of the path are all dependent on the goal, the starting point of the character, the properties of the table and cup etc. Our research focuses on determining and analyzing the factors that influence the shapes of these zones.

Once we have a proper understanding of these zones, we analyze the behavior in such a zone itself.

Experiments

In order to determine these zones, we are conducting several human behavior experiments in our motion capture lab. The resulting data does not only provide us with insight in human behavior but also allows us to derive numerical data in order to develop animation algorithms to simulate these kinds of tasks.

References

1. Kovar, L., Gleicher, M., Pighin, F. (2002). Motion graphs. *Proceedings of the 29th Annual Conference on Computer Graphics and interactive Techniques* (San Antonio, Texas, 23-26 July 2002), 473-482.
2. Wiley, D. J., Hahn, J. K. (1997). Interpolation Synthesis of Articulated Figure Motion. *IEEE Comput. Graph. Appl* **17**: 39-45.
3. Heck, R.,Gleicher, M. (2007). Parametric motion graphs. Proceedings of the 2007 Symposium on interactive 3D Graphics and Games (Seattle, Washington, April 30 - May 02, 2007), 129-136.
4. Heck, R., Kovar, L., and Gleicher, M. (2006). Splicing Upper-Body Actions with Locomotion. Computer Graphics Forum **25**: 459-466.
5. Shapiro, A., Kallmann, M., Faloutsos, P. (2007). Interactive motion correction and object manipulation. Proceedings of the 2007 Symposium on interactive 3D Graphics and Games (Seattle, Washington, April 30 - May 02, 2007), 137-144.