Character Splash System

Scott Cegielski
Dreamworks Animation

1 Introduction

One of the most common effects in the animated feature film Flushed Away are the splashes that result from a character's interaction with a body of water. The quantity of shots that required this effect warranted the development of a character splash system. The system's requirements were to automate the process of generating splashes from a character's interaction with a water surface while still retaining a large degree of artistic control. This system contains tools for generating splash emissions, localizing a fluid simulation, and generating a surface from the resulting particles. This system does not include a fluid dynamics simulator, but instead uses a previously developed in house simulation tool.

2 Splash Generation

In order for an animator to implement this effect efficiently, splashes from character water interaction are generated semi-automatically. This is done in several stages with user controlled parameters which can be animated over time and can also vary over the surface of the character using painted texture maps. First, the character's surfaces are converted with an evenly spaced distribution of particles using a proprietary tool. The spacing of the particles can be controlled based on how much detail is required from the character's motion. Each particle contains a local velocity vector of the character's motion at that position. A closest point calculation is then performed between each particle and the water surface with which the character is interacting. Using each particle's closest point distance, velocity, and user controlled parameters, the following calculations are performed. The particles are first culled based on user defined criteria such as a maximum distance from the water surface. The positions and velocities of the remaining particles are modified using several computed vectors which give the user a high degree of control for the resulting splash direction and amplitude. The resulting particles are then input into the fluid simulation program.

3 Local Fluid Simulation

In some shots, the character is moving at a high rate of speed due to the current in the water. This means that over the course of a shot, the character may travel quite a large distance. The fluid dynamics simulator requires a grid that covers the area of desired simulation. These shots would require a very large box to cover the range of motion of the character. Since detail is required in the simulation, the grid must also be subdivided into many cells. These two factors could lead to very long simulation times. Therefore, an alternative method was implemented to localize the simulation around the character. This is done by stabilizing the character, which involved tracking the center of it's bounding box at every frame and then inverting the tracked data. This resulted in the removal of the large scale motion of the character but retained the local motion such as limb movement. The character's general position ends up at the origin where a much smaller fluid grid is placed for the simulation. In order to retain the sense of the large scale motion in the fluid simulation, a vector force is included in the simulation. The direction of this force is derived from the frame to frame motion of the character and is recorded during the tracking phase. Therefore, if the large scale motion is along the positive Z axis, then the vector force applied to the simulation is in the opposite direction along the negative Z axis. After the simulation, the particles are then transformed so that on each frame they track with the character's pre-stabilized motion.

4 Surface Generation and Integration

Due to the miniature scale in this film, it was decided that the character splashes should be rendered as surfaces as opposed to particles. To create the surfaces, the particles from the fluid simulation are first converted to a density grid with an octree structure using a previously developed tool. For the conversion from the density grid into a polygon file, a program using the marching cubes algorithm is used. The surface is then smoothed using a surface relaxing program (Figure 1). Since the topology of the mesh changes each frame, special handling is required to get accurate looking motion blur. For each frame's polygon file, a second file is generated in which the vertices are deformed. The direction and amount of deformation on the vertices is determined by finding and using the velocity of the closest particle to the vertex from the simulated particle set. This second file is then used at each frame as the previous frame's position for the vertices and thus correct motion blur is achieved. Another requirement was that the splash surface blend into the water surface. This is done by iterating over the vertices in the splash's polygon file. For each vertex, a closest point calculation is performed to the water's surface. As the vertices approach the water's surface, they are deformed along a vector that is both tangential to the water surface and in the direction of the splash surface's normal. This creates what appears to be a meniscus between the two surfaces and when rendered, the splash surface and the water surface appear continuous (Figure 2).

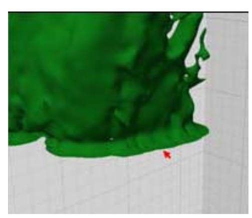

Figure 1: Splash Surface

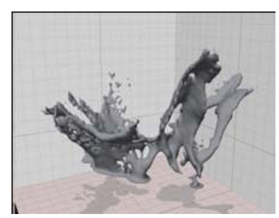

Figure2: Meniscus

5 Conclusion

This system provides an efficient and controllable method for generating splashes from character water interaction and has been used in many production shots. It was written in such a way that as more shots were completed, the system was further refined by adding more user controllable parameters and improving the results. Having the system has also unified the look of the character splashes throughout the film.

© DreamWorks Animation, L.L.C. 2006. This is the author's version of the work. It is posted here for your personal use. Not for redistribution. The definitive version was published in SIGGRAPH 2006 Sketches, https://doi.org/10.1145/1179849.1179873.