Diplomarbeitspräsentation

FAKULTÄT FÜR !NFORMATIK
Faculty of Informatics

A Layered Particle-Based Fluid Model for Real-Time Rendering of Water

Masterstudium:
Computergraphik & Digitale Bildverarbeitung

Florian Bagar

Technische Universität Wien
Institut für Computergraphik und Algorithmen
Arbeitsbereich Computergraphik
Betreuer: Michael Wimmer
Mitwirkung: Daniel Scherzer

Scope of the work

This thesis presents a physically based real-time water simulation and rendering method that brings volumetric foam to the real-time domain.

Foam is an important visual element in most situations where real-time fluids are used and there exists as yet no realistic real-time method for volumetric foam.

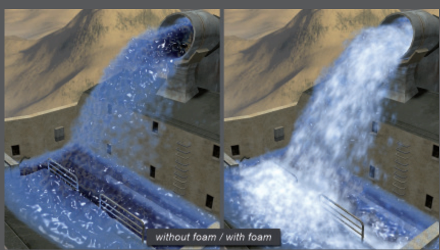
Our algorithm is based on a previously implemented screen space fluid rendering approach which renders the surface of particle-based fluids, smoothes the surface to prevent the fluid from looking jelly-like and is not based on polygonization.

Contributions

- Improved fluid rendering: We introduce an adaptive curvature flow filtering algorithm for smoothed particle hydrodynamics (SPH) rendering which accounts for perspective.
- New foam rendering: We introduce a physically based foam rendering method using Weber number thresholding and a volumetric layer -based rendering system.
- Our method is almost as fast as previous approaches, while providing higher image quality.
- In addition, the algorithm is simple to implement and integrate into existing rendering engines.

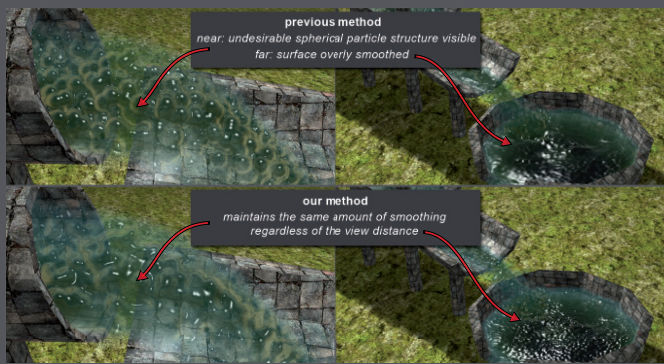
Results

Our approach provides more realistic fluid rendering at comparable cost to previous methods with the benefit of improved image quality especially at near or far viewpoints. Even foam does not significantly increase running time for our method.
We have used an Intel Q9450 CPU with a GeForce GTX 280 graphics card. The particle count is 64k and all images were taken at 1280x720 resolution.



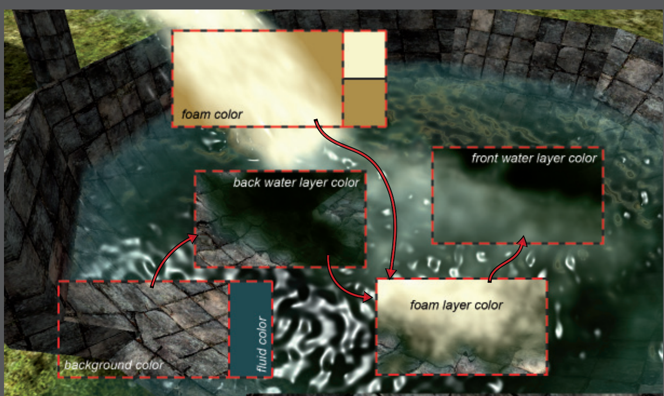
The average running time for our scenes is between 12 ms and 14 ms without foam and between 15 ms and 17 ms with foam (SPH simulation time not included).

Fluid Rendering: Adaptive Curvature Flow



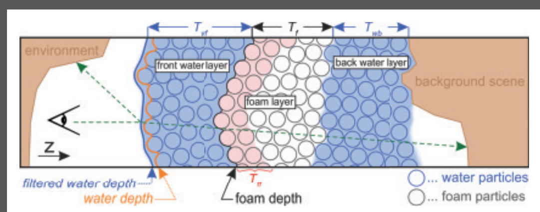
The first step in rendering a fluid using particles is to create the fluid surface by splatting the particles into a depth buffer. In order to avoid a jelly-like appearance due to the spherical particles, it is important to smooth the depth surface.
The previous method uses a fixed iteration count to smooth the surface which introduces view dependent artefacts. In contrast, our adaptive curvature flow filtering varies the number of iterations depending on the view distance, and thus produces a consistent fluid surfaces independent of the viewing distance.

Layer Compositing



The actual pixel color is calculated by volumetric compositing along a viewing ray back to front. Each layer causes attenuation depending on the thickness T of the layer.
In addition to the attenuation by the layers, we also calculate reflection of the environment and specular highlights.

Foam Rendering: Layer Creation



For rendering we have developed a layer-based model which is capable of treating all cases shown on the left side. By partitioning the fluid into a foam layer and two water layers, one in front and one behind the foam, we can simulate foam inside water, as happens at the end of a waterfall.
Since water and foam are volumetric phenomena, the amount of water respectively foam between two layer surfaces is determined in order to allow correct compositing and attenuation. The thickness T of a layer is determined by additively splatting every particle belonging to the volume into a buffer.

Kontakt: florian.bagar@aon.at, scherzer | wimmer@cg.tuwien.ac.at