Real-Time Non Photorealistic Paint Spreading using Stencil Volumes

Marc ten Bosch*

Adapting shadow volumes techniques to fluid rendering

A shadow volume contains all the points in the shadow of an object, and is created by extruding the contour edges of that object away from the light source. Heidmann [1991] adapted shadow volumes to hardware acceleration and as a result introduced **stencil volumes as a method of determining the shadowed regions of a scene**. After rendering the volumes **the stencil buffer holds a mask, which separates the pixels that lie outside the volumes from the ones that lie inside**.

I extend the stencil volumes method to a **different application**, namely the rendering of a fluid spreading on an arbitrary surface in real-time.

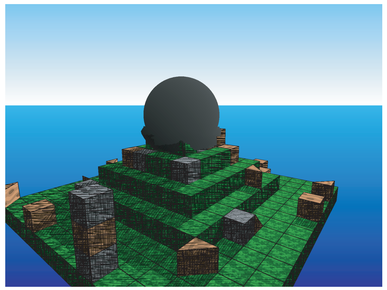
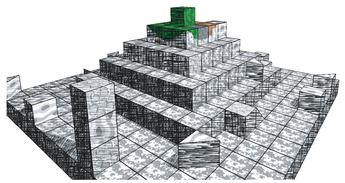
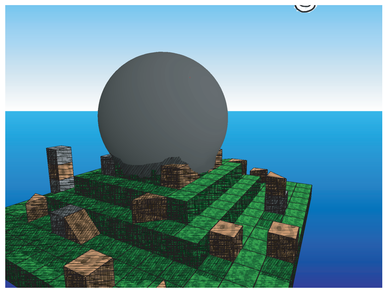
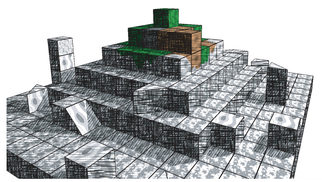
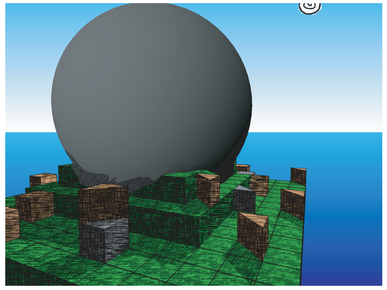
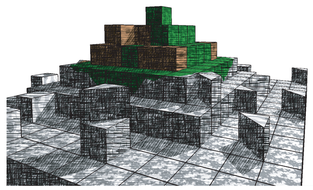
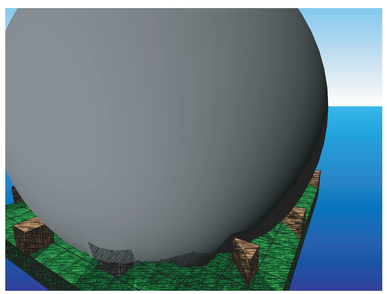
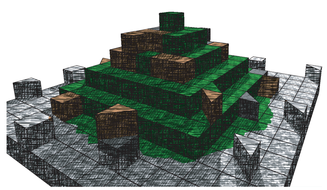
Paint Volumes

Paint volumes enclose the paint as it spreads out in three dimensions. Just as for shadow volumes, the surfaces of the objects that lie inside the volumes are rendered differently from the ones outside. For shadow volumes the inside is rendered in shadow, and the outside in light. For this particular application, and in order to achieve an artistic effect reminiscent of paint spreading on a canvas, the **inside of the volumes is rendered in color and the outside in monochrome**.

Construction and Update

The **paint volume is initialized to a spherical mesh that is updated using a physically-based system** of simple fluid dynamics.

The vertices are initially at rest, then randomly perturbed to simulate the material microstructure of the surface they lie on. An **elastic force** maintains the vertices around their original positions. Undesirable transient oscillations are damped by a **friction force**. Additional forces can be applied to take into account the surface's shape, and in particular to keep the volume in a certain region of space.

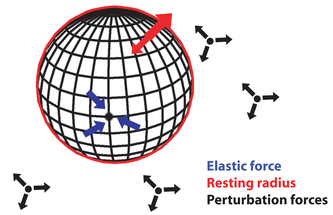









DigiPen
Institute of Technology

*mbosch@digipen.edu

Spreading

The volume grows and shrinks by **varying a desired resting radius**. To simulate additional paint being added to the volume, causing it to spread, the desired radius is slowly increased.



Rendering

For a look reminiscent of a drawing a simplified form of real-time hatching is used to convey lighting, on top of texture and color.

1. Both the **grayscale and colored versions** of the scene are rendered **in the same pass**, as each fragment's alpha component is set to the grayscale value of the final color.

$$\alpha = (r+g+b)/3$$

2. The stencil buffer is set by rendering the paint volumes.
3. The stencil test is set to keep only the fragments that lie outside the paint volumes (with stencil value equal to zero).
4. A **screen aligned quad** is rendered using the alpha saturate **blend mode** of OpenGL to copy the alpha component over the red, green and blue components.

$$(r,g,b,\alpha) \rightarrow (\alpha,\alpha,\alpha,\alpha,1)$$

Advantages over texture-based approach

This technique borrows from shadow volumes in that it is **not prone to aliasing problems**. It provides a dynamic effect **without the cost in memory** of storing individual data for each polygon. It **simplifies computations** by working on the vertex level as opposed to the fragment level. Moreover, it is **independent of mesh and scene complexity**.