

# Fluid Simulation as a Tool for Painterly Rendering

Sven C. Olsen\*  
Swarthmore College

Bruce A. Maxwell†  
Swarthmore College

We follow in the tradition of stroke based non-photorealistic rendering pioneered by Haelberli [1990]. Inspired by expressionistic paintings such as Van Gogh's *Starry Night*, we have begun experimenting with fluid simulation as a tool for brush stroke alignment.

Using simulated fluid fields in the context of stroke based non-photorealistic rendering allows us to produce stroke behaviors which are visually appealing, and which offer expressive potential beyond that of gradient based techniques (such as those used in [Litwinowicz 1997] and [Hertzmann 1998]).

## 1 Algorithms

We have implemented a painterly rendering system very similar to that in [Litwinowicz 1997]. We can create animations based on video clips or still images.

### 1.1 Region Definition

We want to be able to render certain regions of source image, for example the sky or parts of the background, using strokes which are aligned and advected by a fluid field. It is generally not desirable to use fluid-based rendering methods in areas of the video containing more detailed features. Therefore we divide each source image into regions, and specify what fields will be used to govern stroke motion and alignment in each region. Regions are identified using the color space undersegmentation algorithm presented in [Comaniciu and Meer 1997].

### 1.2 Thin Plate Fields

As an alternative to fluid-based alignment, we use a smoothed vector field derived from the gradient of the image. For each region, a set of gradient values is calculated by treating all pixels in the region as intensity 1, all pixels outside the region as intensity 0, and then applying Sobel convolution matrices. We then divide the image into an  $M \times M$  grid, and average all values in each grid box to create a single vector. Vectors perpendicular to the averaged values are then used as control points for two thin plate splines; and together the splines define a field over the region.

### 1.3 Stroke Motion

In video based renderings, we typically use either fluid or optical flow fields to advect strokes between frames. When creating videos from still images, we move non-fluid controlled strokes using the thin plate fields discussed above (advecting only the fluid strokes creates a sense of discontinuity between the moving and static stroke regions).

When strokes are advected they tend to become too sparse in some regions and too dense in others. In order to combat this effect, a quality controlled Delaunay triangulation is used to determine locations where stroke centers should be inserted. If the distance between any two stroke positions is below a given threshold, one is eliminated.

### 1.4 Stroke Rendering

Random permutations are applied to stroke intensity, color, width, length, and alignment. Different permutation parameters may be used depending on the region that the stroke is currently in (crossing a region boundary causes the stroke's permutation variables to be

reinitialized). A region may use curved brush strokes, in which case the alignment permutation is ignored, and a curved stroke is rendered by advecting points through the alignment field.

## 1.5 Fluid Simulation

We simulate a fluid velocity field using a semi-Lagrangian solver as in Stam [1999]. We use vorticity confinement as in [Fedkiw et al. 2001], sometimes increasing the confinement forces well beyond what is physically reasonable in order to exaggerate the turbulence of the field.

We have been experimenting with ways in which the fluid fields can be informed by properties of the image. We usually consider non-fluid rendered regions to be boundaries to fluid flow. In the poppy example image, the fluid velocities were initialized by shooting bursts of force out from the averaged gradient values, running the simulation for a few time steps, and then adding in a small portion of the thin plate field (thus ensuring that the resulting field is non-zero at all points).

It is sometimes useful to control strokes with hybrid fields defined as a weighted sum of the fluid and thin plate fields. We vary the weights at a given point according to the distance to the nearest thin plate control point. Thus we can create a field that is aligned with the region boundaries, and which exhibits fluid-like behavior farther from the region edges.

## 2 Performance

We developed our application on an Athlon 1.4 ghz machine with a GeForce2MX video card. The slowest component of the algorithm is stroke rendering. The rendering time for our demo images was typically 5 seconds per frame. The next most expensive operation is stroke density control, which takes about 2 seconds per frame. Total time to process a single frame is around 8 seconds.

## References

- COMANICIU, D., AND MEER, P. 1997. Robust analysis of feature spaces: Color image segmentation. In *1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)*, IEEE Computer Society, 750–.
- FEDKIW, R., STAM, J., AND JENSEN, H. W. 2001. Visual simulation of smoke. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, ACM Press, 15–22.
- HAEBERLI, P. 1990. Paint by numbers: abstract image representations. In *Proceedings of the 17th annual conference on Computer graphics and interactive techniques*, ACM Press, 207–214.
- HERTZMANN, A. 1998. Painterly rendering with curved brush strokes of multiple sizes. In *Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, ACM Press, 453–460.
- LITWINOWICZ, P. 1997. Processing images and video for an impressionist effect. In *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, ACM Press/Addison-Wesley Publishing Co., 407–414.
- STAM, J. 1999. Stable fluids. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, ACM Press/Addison-Wesley Publishing Co., 121–128.

\*e-mail:sven2718@verizon.net

†e-mail:maxwell@swarthmore.edu