Introducing artistic tools in an interactive paint system

Koen Beets   Tom Van Laerhoven   Frank Van Reeth

Hasselt University - Expertise Centre for Digital Media
and transnationale Universiteit Limburg
School of Information Technology
Wetenschapspark 2, 3590 Diepenbeek, Belgium
{koen.beets, tom.vanlaerhoven, frank.vanreeth}@uhasselt.be

ABSTRACT

While paint systems have been around for a long time, systems capable of capturing the complex behavior of paint media like watercolor, gouache, Oriental ink, oil and acrylic paint have emerged only recently. However, concentrating on the simulation of paint and brush mechanics, these applications mostly provide just a minimal set of instruments assisting users in creating artwork.

We report on the extension of our physically-based paint system for watery paint with a set of versatile tools supplying users with more control during the painting process. We introduce, among others, the use of masking fluid, a special-purpose brush using patterns to steer paint diffusion, and the adoption of an absorbent, textured piece of paper to remove some wet paint from the canvas. Results show that images created with genuine paint, using real-life counterparts of some of these tools, can be closely reproduced with our application. Additionally, our digital tools can produce effects that are difficult or impossible to achieve with real paint, while retaining the spontaneous nature of the resulting images.

keywords: I.3.4 [Graphics Utilities]: Paint systems, I.3.5 [Computational Geometry and Object Modeling]: Physically based modeling, B.6.1 [Design Styles]: Cellular arrays and automata

1 INTRODUCTION

The large amount of research invested in creating digital equivalents of real-life painting is not surprising, given the fact that throughout the majority of our history people have been fascinated with creating images. The activity is accessible and very simple so that users of all ages can practice it; yet, mastering it is complex enough that it remains challenging, even to an expert. Digital interactive paint systems have the same extensive audience.

Working with an ideal paint system would have several advantages compared to the real process:

- the material is free, durable, and can easily be customized according to a user's needs;

- painting becomes even more accessible;

- making images is more practical; there is very little set-up time, no mess and no cleaning up afterwards;

- it allows post-processing and making perfect copies;

- during painting, digital tools allow all sorts of non-physical operations, undoing, removing pigment and water, ...

Despite this convincing list of advantages, so far paint systems have had little impact on how artists make images. Most still prefer to use the traditional methods because the digital results look too artificial. Also, while moving from a real-life canvas to the computer screen, an artist should gain an incredible amount of flexibility, which does not apply to a great deal of the applications the relatively long history of painterly rendering research has brought us.

In recent years, however, researchers are targeting exactly these problems, which already resulted in interactive systems that are able to capture the complex behavior of Oriental ink, acrylic and watercolor paint. Most of these systems provide just a minimal set of instruments complementing the brush and canvas model. Also, while trying to mimic the real-life equivalents of the virtual material, the possibilities of integrating tools that are non-physical or quasi-physical are not fully explored yet. In this work we propose several of these extensions to our simulation for watery paint; some have no correlation with the real world, while others are commonly used by artists but had not been introduced in any paint system yet.

The rest of this work is organized as follows: first we will state our contributions to the domain, followed by a short discussion on previous related work. Next, a quick review of our paint system for thin paint is presented. Section 5 discusses how some of the techniques we will add to the system are applied in real-life, with right after some notes on how we integrated them. Finally, results are shown and discussed in section 7 along with some conclusions and future work in section 8.

2 CONTRIBUTIONS

We extend our real-time interactive paint system for watery paint with the following tools:

Textured tissue removes pigment and water with a textured, absorbent piece of paper, leaving a distinct pattern.

Diffusion controller allows customizing or "steering" the diffusion process by means of a user-defined pattern.

Masking fluid sits on the canvas, protecting paint in all layers beneath it.

Selective eraser removes some/all pigment and/or water from an active paint layer.

Paint profile collects predefined palette and canvas parameters under a single identifier, facilitating the ability to switch between different paint media and styles.

With these increments we are able to create some distinctive effects that previously were difficult or impossible to produce.

3 BACKGROUND

In literature, several authors recognized the fact that interactive painting applications are in need of supporting tools, complementing the procedure of putting paint or ink on a canvas using some sort of brush model. This issue arrises within the domain of interactive painting systems, rather than the "automatic" painterly rendering systems that convert an input image to a painterly style equivalent.

Smith stressed the fact that one should make a difference between a digital paint *program* and a digital paint *system* [Smi01]. Although both can be described as just "applications processing pixels", the term "system" implies many more features and much more capability than just a "program" that simulates painting with a brush on a canvas. A paint system might even encompass several other sub-programs.

Related to this matter, Baxter *et al.* pointed out that current research in the area of painterly rendering mostly emphasizes the *art* of painting, the appearance of the final product, while paying less attention to the *craft* of painting [BSLM01], the way an artist uses different materials to express himself. His dAb paint system for painting with acrylic or oil combines a deformable 3D brush model with a force feedback input device to enhance the user's sense of realism, who can focus on the painting process itself. Using a minimalistic interface, however, it disregards some of the extra digital benefits that can assist a user during the creation of his artwork.

On the other side of the support-spectrum are the commercial paint systems like Adobe Photoshop and Corel Painter. The amount of complex tools integrated with these applications is overwhelming, making it virtually impossible for untrained artists to create digital paintings with the same ease as in the real world.

The recent work of Chu *et al.* on Oriental ink simulations [CT05] incorporates several special effects and controls that are hard or impossible to reproduce in real life. They provide control over paint wetness, allowing dried ink to flow again, or to fast-forward the drying process. An interesting option in their system is the possibility to tinker with the pigment advection algorithm, creating magnetic-like effects where pigment can be pushed or pulled. A "splash and spray" tool provides an alternative way of applying ink to the canvas, by emitting a pattern of ink drops using some simple physics scheme.

4 A CANVAS MODEL FOR SIMULATING WATERY PAINT

In this section we will give a brief overview of our recently introduced canvas model for interactive simulation of thin, watery paint in real-time [VLVR05].

Inspired by an implementation of a system that produced watercolor images from input images [CAS$^+$97], it is the first model that provides real-time painting experience with watercolor paint, targeting both realistic paint behavior and convincing rendering of semi-transparent glazes. In addition, it supports simulation of paint media related to watercolor, like gouache and Oriental ink.

Figure 1 depicts a schematic view on the canvas model consisting of three active layers and an unlimited number of passive layers. The passive layers are considered to be layers of dried paint that do not participate anymore in the dynamics simulation.
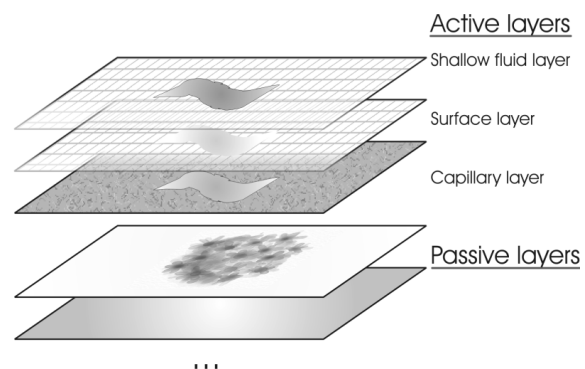

Figure 1: The canvas model, with three active layers and an unlimited number of passive layers.

Each layer consists of a 2D grid of cells on which operations take the form of cellular automata. In the

shallow fluid layer, the model adopts stable fluid dynamics algorithms based on the work of Stam [Sta99] to transfer pigment densities and water quantities on top of the canvas. Heuristic rules handle the deposition of pigment within the irregularities of the canvas surface (represented by the surface layer), as well as the evaporation, absorption and capillary diffusion of water inside the canvas structure (in the capillary layer).

Rendering of the canvas is handled by the Kubelka-Munk diffuse reflectance model.

All algorithms participating in the dynamics phase and the rendering phase of the simulation are implemented on graphics hardware as fragment shaders that operate on a set of textures containing the relevant simulation data.

For more information concerning the details of this model, we refer to literature [VLVR05].

5 REAL-WORLD ARTISTIC TOOLS

In this paragraph we describe how the techniques we implemented in our system correspond to working with water colour in real-life.

5.1 Textured tissue

In real-life, an absorbing piece of paper, a sponge, or even dried breadcrumbs can be used to remove an excess of pigment and water from the canvas and to leave an impression of the texture. This can be done right after applying the paint, but also when the paint is already dry. In that case, the dried paint is made wet again by rubbing over it using a wet brush. This way, some paint pigments come loose again, and can adhere to the absorbing piece of paper. The result is a print of the texture of the paper, which can create interesting visual effects. Figure 3 shows the application of an absorbing piece of paper.

5.2 Diffusion controller

The second contribution of this paper is a diffusion controller, which allows steering the diffusion of paint, using a special-purpose brush. In this way, we can obtain blooming colours also obtainable in real-world painting when painting using a technique known as "wet-in-wet". The English artist Joseph Mallord William Turner was the first to exploit extensively the wet-in-wet technique, where paint is applied on a wet piece of paper [Mac05]. Combined with the transparency effects that can be obtained with glazing, blooming can create vibrating visual effects.

In real-life, the amount of blooming depends on the wetness of the paper, on the pigment characteristics and on paint quality. The directions and amount of blooming can be influenced by holding the paper in the desired direction, because it is influenced by gravity, and by the amount of paint and water. Blooming is often used for skies and water in particular, but also to obtain interesting effects of atmosphere, light, and texture.
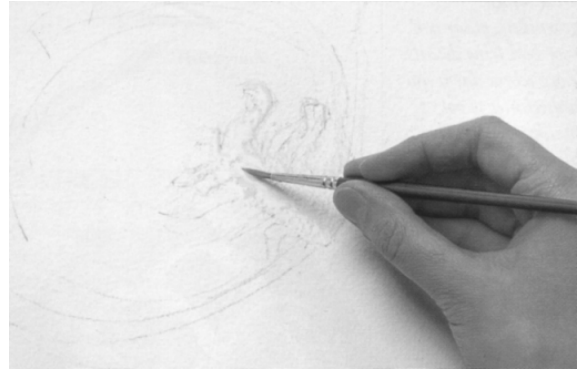

Figure 2: Applying masking fluid with a brush on the canvas [Smi98].

5.3 Masking fluid

Masking can be useful to prevent staining a specific area of the canvas, in several ways. A first application is to obtain sharp and exact contours in the painting. A second application is to protect large areas from paint when using broad washes.

Note that masking can be used both in areas with and without paint, and that it can be used in layers. In real-life painting, masking can generally be done using two different materials: tearable adhesive tape, and masking fluids.

A first method is to delineate a certain area of the paper by using adhesive tape, which can be teared to obtain specific shapes. This method however does not allow for very precise control. A second, and more precise, method is applying masking fluid with a brush on the canvas (figure 2). This fluid is a removable colorless liquid made from rubber and latex staining that can be used to mask areas of work that need protection when color is applied in broad washes. Once dry, these areas remain protected and cannot be penetrated by colour. Afterwards, the dry masking fluid can be removed by rubbing with the hand.

In our system, the masking fluid is more easily and precisely controllable. The tool retains its usefulness, even with the availability of Undo functionality, and the use of several layers. With the masking fluid, one can easily obtain sharp borders throughout several layers of paint, whereas it would be cumbersome to get this effect without.

5.4 Selective eraser

This tool has no exact counterpart in real world painting. There, it is only possible to remove part of the pigments which are deposited on the paper. Precise control is not possible, and selectively removing a specific pigment is not possible either.

5.5 Paint profile

In real life, a popular technique is "mixed media", which means a painting is created using a combination of different media, e.g. both Chinese ink and water colour, and even gouache. Using these profiles, it becomes much easier to integrate different media into a single painting.

6 DIGITAL ARTISTIC TOOLS

We will now take a closer look on how the techniques introduced in the previous section are implemented.

6.1 Textured tissue

The tissue is applied to areas on the canvas where paint is not completely dry yet. At this point the brush becomes a metaphor for the finger, applying pressure on the tissue as long as the brush touches the canvas surface.

The tissue is modeled as a texture in which one color channel is used for storage of a height field (the irregular surface of the tissue), and a second color channel marks transferred cells on the tissue, making sure the tissue does not remove all material from the canvas.

Water in the fluid layer of the canvas model is absorbed by the tissue and, at the same time, an amount of pigment is transferred from the canvas surface area to the tissue according to the magnitude of the pressure at that point. In our examples we used a circular pressure field with a Gaussian distribution. The amounts of material removed are not individually stored by the tissue but combined to a single "mark"-value. This procedure is shown in figure 5.

The operation is implemented using three fragment shaders. Two shaders remove pigment and water amounts from the canvas according to the tissue texture's x-channel, containing height field values between 0 and 1 (the irregular surface). A third shader marks the transferred cells in the y-channel of the tissue texture.

We assume that a "clean" tissue is used after lifting the brush from the surface. At this time, the marked cells on the tissue are cleared.

The left side of figure 4 depicts a close-up of the tissue used in our system. It was sampled from the piece of paper displayed in the center part of figure 3. The final result of this operation is a footprint of the tissue, as shown in the right part of figure 4.

6.2 Diffusion controller

The "diffusion controller" introduces per-cell blocking values in the simulation, forcing pigment densities to follow a predefined pattern during the diffusion process. There is no real-world tool that allows this kind of paint manipulation.

The simulation step that deals with the diffusion of pigment densities is handled by the diffusion component $\nu\nabla^2\vec{p}$ of the Navier-Stokes equation [Sta03]. Because of its simplicity and cheap execution on graphics hardware, we use the Jacobi iteration method to solve this Poisson equation and calculate the new pigment densities p^{new} at cell (i,j) each time step [VLVR05]:

$$p_{i,j}^{\text{new}} = (\alpha p_{i,j} + p_{i+1,j} + \\ p_{i,j+1} + p_{i-1,j} + p_{i,j-1})/(4.0 + \alpha) \quad (1)$$

with $\alpha = \text{cellarea}/(\nu\Delta t)$ and ν the diffusion rate. Intuitively, what happens is that all cells keep exchanging content, trying to cancel out differences in concentration with neighbouring cells (figure 7). This way, when dropping pigment densities into a wet area on the canvas, this particular diffusion algorithm would result in something similar to the top image in figure 6, which is not what we want and which certainly does not correspond to what happens in real-life (section 5.2).

If we introduce a scale factor $\beta_{i,j}$ at each point where cells exchange content, we can control the amount of pigment density that is passed from one cell to a neighbour while retaining conservation of mass:

$$p_{i,j}^{\text{new}} = (\alpha p_{i,j} + (\beta p)_{i+1,j} + (\beta p)_{i,j+1} + \\ (\beta p)_{i-1,j} + (\beta p)_{i,j-1})/(\beta_{\text{sum}} + \alpha) \quad (2)$$

with $\beta_{\text{sum}} = \beta_{i+1,j} + \beta_{i,j+1} + \beta_{i-1,j} + \beta_{i,j-1}$.

A blocking factor $\beta_{i,j} = 0.0$ effectively cancels out pigment exchange with neighbouring cells, while $\beta_{i,j} = 1.0$ corresponds to no blocking at all. These values are stored in a texture (the blocking texture) and used by the fragment shader that implements the diffusion algorithm to look up per-cell blocking values. Figure 6 shows a few examples of blocking patterns we used in painting with black ink. The second pattern from the top is drawn by hand and is mainly used in the results because of its natural appearance. Once the brush tip touches the canvas such a pattern is written into the blocking texture, locally "steering" the diffusion process in the next simulation steps. In practice, we combine values of the height field of the canvas with values from the blocking pattern to create a combined blocking value. This takes into account the fact that the canvas surface also affects the way pigment diffuses. The percentages in figure 6 indicate how much the pattern contributes to the total blocking value.

This procedure is able to mimic the blooming effect in the brush's footprint, as described in the previous section. In extending this technique to create similar

Figure 3: Applying a textured tissue to an area of wet black ink. The result is a print of the texture of the paper.

effects in a stroke once the brush is dragged on the canvas, we have to make sure the blocking pattern is applied at regular intervals. Moreover, the pattern has to disappear after some time period in order to avoid interferences with neighbouring strokes. This issue is handled by assigning each pattern in the blocking texture an age-value. Each time-step a small value is subtracted from the cells in the blocking texture, reducing them to zero over time.

The right image in figure 6 demonstrates some strokes drawn in black ink with the diffusion controller brush.

6.3 Masking fluid

The masking fluid forms a protective, solid layer on top of the canvas, securing all layers below. While painting with this special kind of pigment, instead of adding densities to the textures containing pigment data, we mark affected cells in a mask layer that has no role in the dynamics phase. This layer is drawn on top of the canvas.

Because the passive layers do not participate in the dynamics phase of the simulation, we only have to consider the effect of the masking fluid on the three active layers containing water quantities and pigment densities.
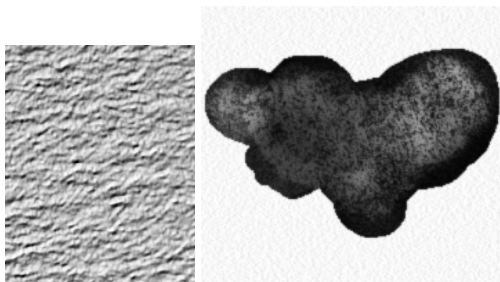

Figure 4: (*left*) A fragment of the textured tissue used in our system, rendered with per-pixel bump mapping. (*right*) A computer-generated example showing the resulting image after application of the textured tissue tool on a wet area of black ink. A distinct print of the tissue is still visible.
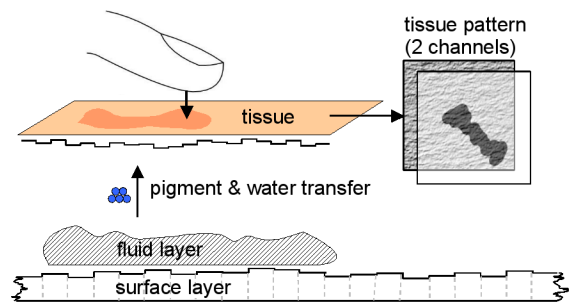

Figure 5: Removing water and pigment from the canvas surface with an absorbent tissue.

Similar to real masking fluid it is possible to cover it while painting; the paint just should not reach the canvas. This condition is assured by the ability to remove all pigment in the masked area when starting a new layer, or when removing the masking fluid.

When starting a new layer, all water quantities and pigment densities in the active layers covering the mask pigment are first removed, leaving the canvas untouched in that area. The same procedure is followed when removing the masking fluid.

The masking fluid is distinguished from strokes drawn with regular pigment by decorating it with whites crosses on red circles, as indicated in figure 8.

6.4 Selective eraser

A more versatile tool than the classic eraser that wipes paint from the canvas is the "selective eraser". It can reduce or remove densities of a specific pigment from a mixture, both from the fluid layer and the surface layer. Similarly, it is possible to reduce or remove water quantities from the fluid layer and the capillary layer.

Because all simulation data is stored in float-texture objects, the implementation of this tool translates to a simple fragment shader that can be customized for the manipulation of one or more specific color channels of the data texture.

Despite its simplicity, the selective eraser has proofed it's usefulness as an error-correction tool.
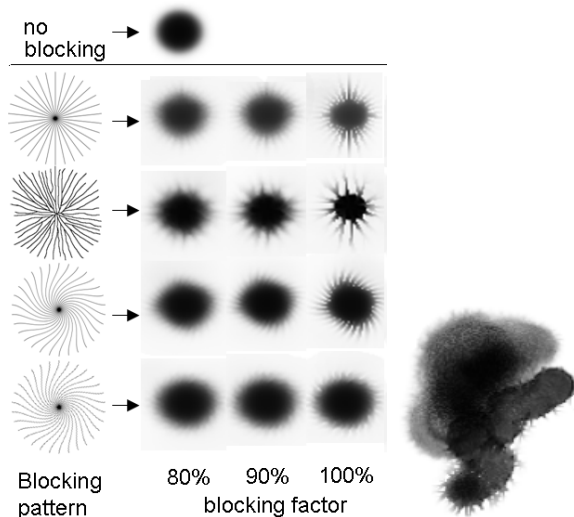
Figure 6: (*left*) Examples of blocking patterns used to "steer" pigment in the diffusion process. Each row shows the effect of 80%, 90% and 100% contribution of the pattern to the total blocking value. (*right*) A computer-generated example of some strokes drawn with the diffusion controller brush. The hand-drawn pattern was used to influence pigment diffusion, giving it its natural look.
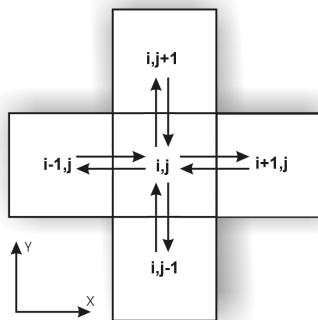

Figure 7: The unmodified diffusion algorithm makes cells exchange contents with neighbouring cells until an equilibrium is reached.

6.5   Paint profile

Our paint system is capable of simulating several related paint media: watercolor, Gouache and Oriental ink. Each one depends on a different setup of the simulation [VLVR05].

For example, Oriental ink requires a highly textured and absorbent canvas, and the dense carbon particles are so heavy they can be carried inside the canvas. Gouache on the other hand resembles watercolor but is applied in thick opaque layers on the canvas, creating flat color areas. Its palette contains a white pigment, which is not present in a watercolor palette. Therefore we have a large amount of settings that relate to the canvas, the fluid flow, the palette and the brush model.
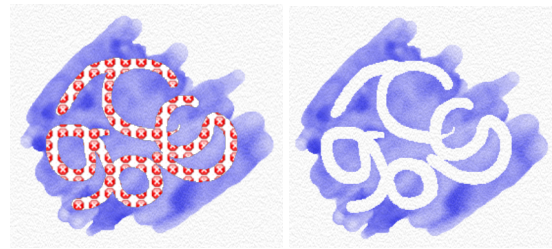

Figure 8: Using the masking fluid, which was previously applied by a brush (*left*), to cover the canvas beneath. To distinguish the masking fluid from regular pigment, it is decorated with a distinctive pattern. After removing the masking fluid the canvas remains untouched (*right*).

To cope with these different environments we introduce the notion of a paint profile, which combines all such properties and relates them to a specific name. The result is that we can easily change to a different paint media with a single mouse click and that we can even mix different styles in a single painting. At present our system supports three profiles: watercolor, Gouache and diffuse Oriental ink.

7   RESULTS

All computer-generated images presented in this work were drawn interactively and in real-time on a simulation grid of 512×384 cells, processed on a Intel(R) Pentium4 2.40 GHz system equipped with a NVIDIA GeForce FX 5900 graphics card. A Wacom tablet interface was used as an input device. As described in [VLVR05], a frame rate of 20 Hz is attained by processing only the areas of the canvas that correspond to "dirty" cells in an overlay grid.

The simulation results are displayed at higher resolution than the simulation grid by applying simple bilinear interpolation. Unfortunately, this also causes blurring of sharp stroke boundaries. One solution would be to increase the resolution of the simulation grid, at the cost of a slower simulation. A better approach is presented by Chu *et al.* [CT05], which consists of an image-based technique they call "boundary trimming". It restores sharp edges by reconstructing them with an implicit curve that is used to "trim" excessive pixels. This post-processing step could be integrated in our simulation to produce higher resolution images at low cost.

Figure 9, a painting in Oriental ink based on the work of Marie-Anne Bonneterre, demonstrates the use of both the textured tissue and the diffusion controller. The black background was taken from a real painting made with black ink.

The two results in figure 10 show how the masking fluid can be used to protect existing layers while painting a darker glaze on top.

Figure 9: A computer-generated image in Oriental ink, based on the original work of Marie-Anne Bonneterre. The distinct patterns on the flags are caused by usage of the textured tissue in combination with the diffusion controller. The black background was scanned from a real painting made with black ink.

Figure 10: Applying masking fluid to "oranges", a watercolor work in progress (*left*), in order to place a message in a part of the orange that is darkened (*right*).

8 CONCLUSION AND FUTURE WORK

We augmented the capabilities of our paint system by means of several useful tools. Each one contributed to the expressiveness of the painting experience, facilitating the creation of images that lack the traditional artificial look of computer-generated images, as was shown in the results section.

There is more to a pigment than just the colour alone. Characteristics are e.g. the amount of staining, transparency, granulation, and spreading quantity. By using watercolor mediums, these extra characteristics can be modified. Future work includes investigating how we can offer these paint modifiers to the user efficiently.

9 ACKNOWLEDGMENTS

We also would like to thank Marie-Anne Bonneterre and José Xavier for providing us with some real artwork and for sharing their expertise on the domain.

REFERENCES

[BSLM01] William Baxter, Vincent Scheib, Ming C. Lin, and Dinesh Manocha. dab: interactive haptic painting with 3d virtual brushes. In *Proceedings of ACM SIGGRAPH 2001*, pages 461–468. ACM Press, NY, USA, august 2001.

[CAS$^+$97] Cassidy J. Curtis, Sean E. Anderson, Joshua E. Seims, Kurt W. Fleischer, and David H. Salesin. Computer-generated watercolor. In *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 421–430. ACM Press/Addison-Wesley Publishing Co., NY, USA, 1997.

[CT05] Nelson S.-H. Chu and Chiew-Lan Tai. Moxi: real-time ink dispersion in absorbent paper. In *Proceedings of ACM SIGGRAPH 2005*, volume 24. ACM Press, NY, USA, july 2005.

[Mac05] Bruce MacEvoy. Wet in wet. Handprint: watercolour website, `http://www.handprint.com/HP/WCL/tech23a.html`, 2005.

[Smi98] Stan Smith. *The complete watercolour course*. Collins & Brown, second edition, 1998.

[Smi01] Alvy Ray Smith. Digital paint systems: An anecdotal and historical overview. *IEEE Annals of the History of Computing*, 23(2):4–30, 2001.

[Sta99] Jos Stam. Stable fluids. In Alyn Rockwood, editor, *Siggraph 1999, Computer Graphics Proceedings*, pages 121–128, Los Angeles, 1999. Addison Wesley Longman.

[Sta03] Jos Stam. Real-time fluid dynamics for games. In *Proceedings of the Game Developer*, march 2003.

[VLVR05] Tom Van Laerhoven and Frank Van Reeth. Real-time simulation of watery paint. In *Journal of Computer Animation and Virtual Worlds (Special Issue CASA2005)*, volume 16. J. Wiley & Sons, october 2005.