Volume Painting

Abstract

In this paper, a new non-photorealistic rendering concept, volume painting is introduced. Volume painting is a three-dimensional generalization of classical painting. In volume painting, brushes move in three-dimensional space leaving paint objects behind. The material properties of these paint objects are determined by sampling the material properties of space. These samplings provide a noise that creates the painted look.

A prototype painting system based on the volume painting concept is described and images created using this volume painting system are presented.

1 Introduction

One of the most important goals in computer graphics is the development of new concepts and techniques for creating artistic [11, 14] or natural looking [7, 9] images and shapes. We have observed *three inherent properties* of painting that establish a good paradigm for obtaining both artistic images and shapes.

1. *Paintings are events in space-time.* Some of the richness of painted images comes from the application of many layers of different paint over the canvas. Since each layer of paint goes on over the previous layers, the appearance of a painting implicitly contains the history of painting process. In other words, D'arcy Thompson's comments [16] that organic forms are *events in space-time* apply equally well to the appearance of paintings.
2. *Paintings can be considered to be sampled and reconstructed images.* The reconstruction can be done by using materials such as oil paints, water paints, colored papers, etc. Regardless of the materials used, painters introduce predictable noise into the paintings. This noise is useful since it can make paintings interesting by stimulating the imagination of the viewers. It is useful to classify noise in paintings into shape and color noise.
    (a) *Color noise.* The perceived colors in a painting are not necessarily the colors of the paint. Impressionist painters, such as Georges Seurat and Paul Signac, are well-known for the methodical introduction of jittered paint spots to produce perceptual color. However, the use of noise to affect color perception has been used throughout the history of painting. Some painters such as Rembrant Van Rijn and Diego

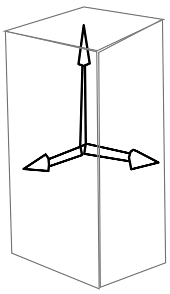
Velazquez even created a perception of gray by applying layers of chromatic paint [13]. Noise is also used in Computer Graphics to create the illusion of paintings from images [11].

   (b) *Shape noise.* Shape perception is not necessarily obtained by perfect description of the object. In many cases, partial description is enough to create a perception of shape [3]. Some painters, such as Paul Cezanne or Henry Matisse, are well-known for the introduction of such noise in shape. However, careful inspection reveals that noise for shape perception has also is a common device used by almost all painters [13].

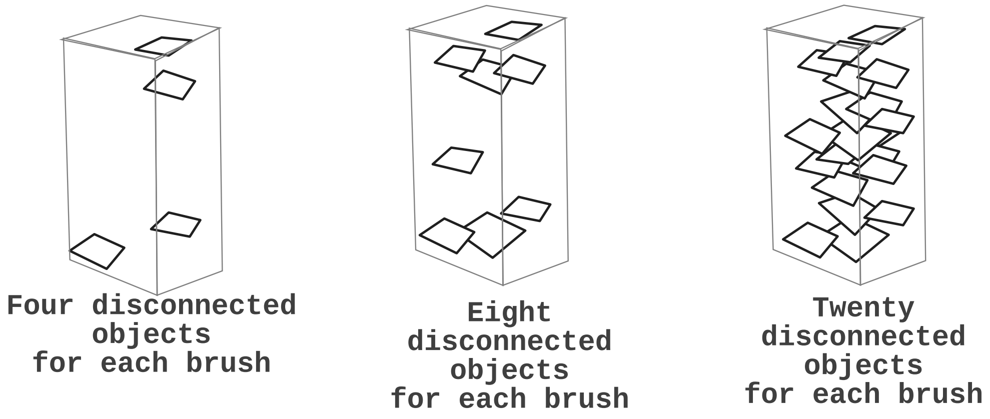
3. *Painting is a subjective process.* Painters work in highly individual ways. They do not stop painting until they are satisfied; they may use different brushes, different paints. These decisions of painters are subjective. Some painters even leave most of the canvas unpainted by stopping after a few brush strokes. On the other hand, some painters continuously apply new layers covering the canvas with several layers of paint.

We have developing a new algorithmic painting concept, we call volume painting. Volume painting is the generalization into three-dimensional space of the three inherent properties of classical painting. In other words, a volume painting is an event in space-time, it is produced by a sampling and reconstruction process that populates three-dimensional space with different types of objects and the volume painting process should not interfere the subjectivity of the painter. In other words, the goal in volume painting is to paint the volumes exactly like painters paint a canvas. A volume painter controls the motion of a set of *"virtual brushes"* that continuously move in three-dimensional space. Like real brushes, when these virtual brushes move, they leave some *"virtual paint"* behind them. This virtual paint may be graphical objects such as dodecahedra, tetrahedra, cylinders or quadrilaterals. Painters should be able to control the attributes of these graphical objects, such as size and material properties, directly or as functions of position and time. Based on the volume painting concept, we have developed a prototype volume painting system. An example of an image created with our volume painting system is shown in Figure 1.

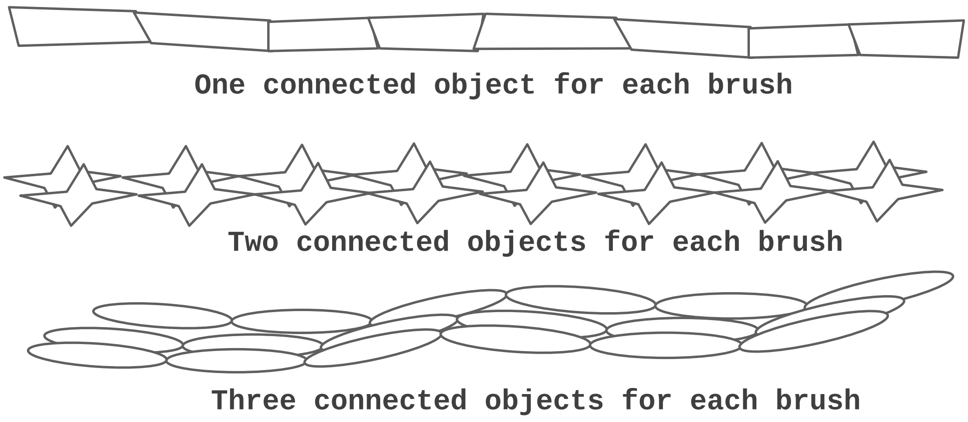
There is a great body of work in computer graphics such as cellular texture generation [9], voxel automata [12], or L-systems [15] which are closely related to the idea of volume painting



a. Coordinate System and Brush Size Prism



b. Airbrush Mode



c. Oil Paint Mode

Figure 1: Examples of paint.

since they create events in space-time. However, these are not exactly volume paintings since they do not include sampling and reconstruction noise. Painterly rendering [14] is also closely related to volume painting since it creates reconstruction noise by populating three-dimensional space with particles and attaching brush images to these static particles. However, since the particles do not move, in painterly rendering the resulting images are not events in space-time.

2 A Conceptual Framework for Volume Painting Systems

In this section, we introduce a conceptual framework for volume painting systems. We view the volume painting process simply as a set of brushes moving in three-dimensional space and leaving paint objects behind them. These paint objects have their own shapes and material properties that are defined by sampling material properties of three-dimensional space. Based on this description, we have identified the four basic elements of a volume painting system as layered surfaces, brush motion, spatial material properties and paint objects.

2.1 Layered Surfaces

We propose to use layered surfaces for the representation of solids. For instance, a solid sphere can be considered layers of spherical surfaces with different radii. In this example, the radius of a sphere can be used as a layer parameter to identify its surface.

A volume painting system should provide a representation for layered surfaces. Moreover, the layered surfaces should be two-manifolds in order to guarantee that brushes move freely on a surface without reaching a boundary.

The current surface that attracts the brushes will be called the target layer. In a volume painting system, the painter should be able choose the target layer by changing a layer parameter. If each layer is only partially covered by paint, it is possible to see several layers. Figure 1 demonstrates this characteristic.

2.2 Brush Motion

We have observed the power and simplicity of turtle geometry [2] comes from that separation of velocity vector into speed and direction. We suggest to use a similar approach to define the motion. In other words, in a volume painting system the current state of brush will be defined as a triplet $(p(t), s[x(t)], n[x(t)]$ where the integer t denotes the current time, unit vector $p(t)$ denotes the current position of the brush, unit vector $n[x(t)]$ denotes the current direction of the motion and the real number $s[x(t)]$ denotes the current speed and $x(t)$ denotes all the control parameters that affect the direction and the speed of the brush, including the layered surface on which the brush moves. We adopt the convention that time advances in unit steps, so that $t-1$ denotes the previous time and $t+1$ denotes next time. The next position of the brush, $p(t+1)$, is given by

$$p(t+1) = p(t) + s[x(t)]\; n[x(t)]. \tag{1}$$

If $s[x(t)]$ and $n[x(t)]$ are arbitrarily chosen, the brushes will roam in three-dimensional space without any goal. Instead, they need to be attracted by a target surface and should stay on the target surface when they reach it. In order to move the brushes towards the target surface, a volume painting system should provide equations for computing $s[x(t+1)]$ and $n[x(t+1)]$. In addition, the painter should be able to control the speed and direction of motion of the brushes in a simple and intuitive way.

2.3 Material Properties of Three-Dimensional Space

In order to sample the properties of space, it should be possible to compute the material properties of any given point p in three-dimension by using a shader function $m(p)$. These material properties are vectors that might include such characteristics as a diffuse color and shader coefficients.

2.4 Paint

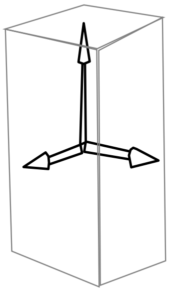
It is assumed that a set of graphical objects is attached to the brushes, which leave these objects behind as they move in space. We call these graphical objects *paint objects* or simply *paint*. Paint will generally be described as a set of simple shapes such as polygons. The shapes of paint can be approximations of any shape such as circles, spheres, tubes or even flowers and insects. During the volume painting process, painter should be able to change four attributes of the paint: (1) *the number of paint objects,* (2) *paint shapes,* (3) *paint sizes*, and (4) *material properties of paints.* The number of paint objects is analogous to the bristles of an actual brush. Paint shapes and material properties are analogous to the paint types such as oil paint or water color. Paint sizes are related the thickness of actual paints.

A local coordinate system is attached to each brush in order to compute the position and orientation of the paint objects to be left behind. A set of equations for computing the orientation of this coordinate system should also be provided by the volume painting system. The volume painter should be able to describe *brush size* which is given by the dimensions of a rectangular prism centered at $p(t)$ as shown in shown in Figure 2.a. The paint objects that are left behind will be positioned inside of this rectangular prism. The position of the objects inside of this rectangular prism is determined by the mode of operation. We identify two distinct operation modes airbrush and oil paint.

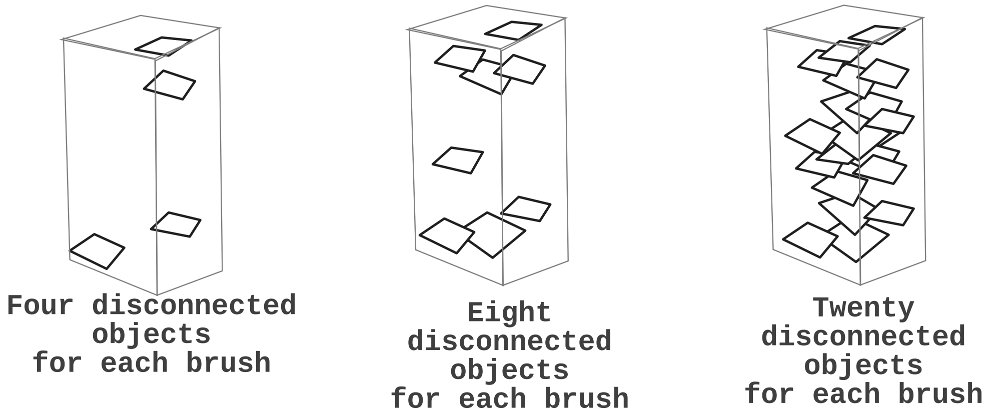
- *Airbrush.* In airbrush mode, objects are randomly placed inside of the prism regardless of the position of the paint in the previous state of the brush. Examples of positioning the paints in airbrush mode are shown in Figure 2.b.
- *Oil Paint.* In painting mode, each paint object has to be placed in the prism in such a way that there should be a connection between the previous, current and next paint objects. Examples of positioning the paints in oil paint mode are shown in Figure 2.c.

The essential element in volume painting is determination of the material properties of the paint objects that are left behind. This determination process consists of four stages.

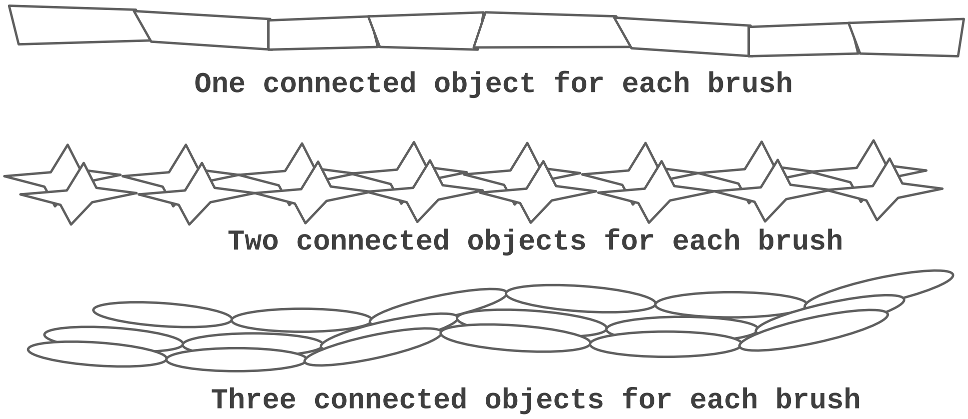
- The first stage is to attach a material property to each brush when brushes are initially created. These brush material properties will be used to generate material properties of graphical objects.
- The second stage is to compute the material properties of paints by adding noise to the material property of the brush. For instance, if the diffuse color of the brush is green, the diffuse color of a paint object attached to this particular brush may be generated by adding a little bit of blue or red to the green. During the volume painting process, the painter should also be able to change the amount of this noise.
- The third stage is to compute the material property of the three-dimensional position where the graphical object to be left. This material property is computed using the shader function $m(p)$.
- The last stage is to compute the material property of the graphical object is to be left behind. This material property is computed as a weighted average of the material property



a. Coordinate System and Brush Size Prism



b. Airbrush Mode



c. Oil Paint Mode

Figure 2: Examples of paint.

of the three-dimensional position and the material property of the paint object. The painter should be able to change the weights of the interpolation function in order to control the amount of noise.

Although the original material properties of the paints may differ, when they pass thorough the neighborhood of a particular point they will all share some of the material property of that point. The layers of paints which share the same point's material property create a mixture that tends toward the material property of that point.

3 The Prototype Volume Painting System

We have developed a prototype volume painting system, in order to show the viability of this concept and to introduce the images in this paper. In our system, we use implicit representations to provide layered surfaces and brush motion; and material properties of space are given by solid textures. This volume painting system is obtained by adding layered surface and solid texture representations into an earlier implicit surface painting system [1] and by improving the brush motion equations. The new equations for brush motion is simpler and more intuitive.

3.1 Layered Surface Description

In our volume painting system, the layered surfaces are given by implicit equations in the form of

$$\mathcal{S}(f-a) = \{ \ p \ | \ f(p) - a = 0 \ \}.$$

where $p = (x, y, z)$ is a point in three-dimensions, f is a C^1 function from $\Re^3$ to $\Re$ and layer parameter a is a real number. The volume inside surface $\mathcal{S}(f-a)$ is given by

$$\mathcal{V}(f-a) = \{ \ p \ | \ f(p) - a \leq 0 \ \}.$$

Since, for any two real numbers $a_1 < a_2$, $\mathcal{S}(f-a_1) \in \mathcal{V}(f-a_2)$ this representation provides an infinite set of layers.

3.2 Implicit Surface Painting Equations

Characteristics of implicit representations [6, 18, 5] are quite suitable for development of equations for brush motion.

- The gradient vector ∇f gives the normal vector to the surfaces $\mathcal{S}(f-a)$. (Note that gradient is independent of layer parameter a.) We can use this normal vector to move in three-dimensional space to reach $\mathcal{S}(f-a)$. This idea is used by Bloomenthal [6] in order to sample implicit surface, $\mathcal{S}(f)$.
- Any vector perpendicular to ∇f will be on the tangent surface of $\mathcal{S}(f-a)$. By using this vector, it is possible to write an equation to make the brushes float over surfaces. Witkin and Heckbert [17] used this fact in order to achieve a uniformly sample a complex surface using local repulsion between *floating* particles and letting particles be born and die.

In order to develop implicit painting equations which will give control to the painter, we also use the gradient vector ∇f.

Let three orthonormal vectors n_0, n_1 and n_2 describe the coordinate system attached to brush. In our volume painting system, we choose the gradient vector ∇f as one of these normal vectors

$$n_0 = \frac{\nabla f}{|\nabla f|}.$$

We compute the other orthonormal vectors n_1 and n_2, by using the previous direction of brush motion $n(x(t-1))$ as

$$n_2 = \frac{n_0 \times n(x(t-1))}{|n_0 \times n(x(t-1))|},$$

$$n_1 = n_2 \times n_0.$$

The orthonormal vectors n_1 and n_2 are used to find the current direction of brush motion $n[x(t)]$. In our system, the painter only provides the rotation angle $\theta(t)$ in order to provide simple control of the current direction of brush motion. The unit vector denoting the goal direction is then given by

$$r = \cos\theta(t)\; n_1 + \sin\theta(t)\; n_2.$$

If the brushes move in the direction of r, they can never reach the target surface. We need another vector to attract the brushes towards the target surface. The vector $v(f-a)$ that is given as

$$v(f-a) = -\frac{(f-a)}{|\nabla f|} n_0 = -\frac{(f-a)\nabla f}{\nabla f \bullet \nabla f}$$

attracts the brushes towards the target surface at the direction of surface normal ∇f. This vector will be zero when the brushes are on the target surface. Thus, the current direction vector can be computed simply by normalizing the linear combination of the vectors r and $v(f-a)$[1]

$$n[x(t)] = \frac{r + b(t)\ v(f-a)}{|r + b\ v(f-a)|}, \tag{2}$$

where scaling factor $b(t)$ is a positive real number that can be controlled by the painter. It is possible to view $b(t)\ v(f-a)$ as a feedback term to make the particles attract and stay on the target surface $\mathcal{S}(f-a)$.

In our volume painting system, the current speed $s[x(t)]$ is controlled directly by the volume painter, therefore, it will simply be denoted as $s(t)$. If we rewrite the equation 1, we obtain

$$p(t+1) = p(t) + s(t)\ n[f - a(t),\ b(t),\ \theta(t)]. \tag{3}$$

Equation 3 shows all variables that can be controlled by the painter for computing the next position of the brush. In equation 3, the value of $b(t)$ is not extremely critical, but, there exists an interdependency between surface curvature, $s(t)$ and $b(t)$. Depending on the values of $s(t)$, $b(t)$ and the surface curvature, the brushes either converge towards the target surface, or move in an orbit around the surface, or make a periodic motion as illustrated in Figure 3. Even if the brushes move on a distant orbit, the painters will not immediately notice any difference, since the effect will be the same as using a different layer parameter $a(t)$. However, if a periodic behavior occurs, it will be noticed immediately as a ripple. Such ripples which are the result of high curvature regions can be eliminated by choosing smaller values for either $s(t)$, $b(t)$ or both. The value of $a(t)$ simply controls the target surface, therefore $a(t)$ is also not extremely critical for the motion of the brushes.

The overall shape of the brush motion curve is mostly affected by the values of $s(t)$ and $\theta(t)$. On a planar surface $\theta(t) = 0$ gives a straight line, on the other hand $|\theta(t)| < \pi/2$ with constant $s(t)$ draws a regular polygon which approximates a circle. Since brushes have states and the length of $s(t)$ can change, even parametric L-systems [15] can be used to control brush

---

[1] The equation 2 can also be obtained by using Witkin and Heckbert's approach [17] and simplifying their motion equation. Since in volume painting the motion of one brush does not have to be dependent on the positions and motions of other brushes, we do not use a local repulsion. In addition, shape parameters and time derivatives of the shape parameters can also be ignored since we do not have to change the shapes of the layered surfaces. These eliminations makes our painting equations extremely simple.

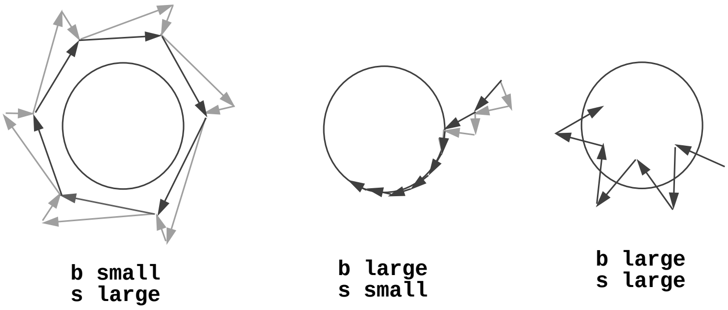


Figure 3: The relationship between $s(t)$ and $b(t)$ for a given curvature.

motion in order to draw a tree-like curve over the implicit surface. Instead, we use a very simple control by using the equation

$$\theta(t+1) = \theta(t) + c\,\delta\theta, \tag{4}$$

where c is a uniform random number between 1 and -1 and scale factor $\delta\theta$ is a positive real number. The values $\delta\theta$ and initial angle $\theta(0)$ are provided by the painter. Likewise, $\delta\theta$ and $\theta(t)$ can be changed during the painting process. The value of scale factor $\delta\theta$ is essential for a good coverage of $\mathcal{S}(f-a)$. If $\delta\theta$ is very small, the brush draws similar curves without covering most of the surface. On the other hand, if $\delta\theta$ values are bigger than $\pi/4$, the brushes tends to stay in a local area. Figure 4 shows how the values of $\delta\theta$ can affect the uniformity of trajectory distribution.

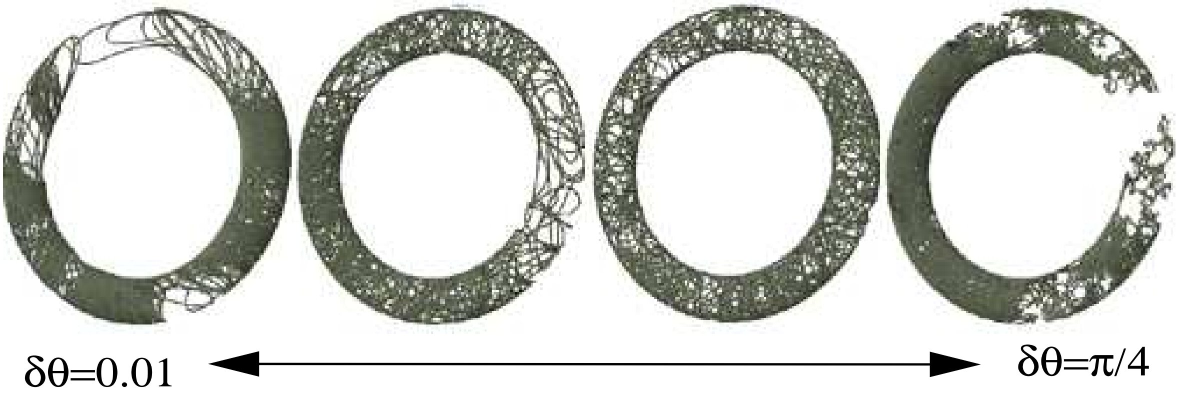


Figure 4: Trajectory of only one brush over a torus for different $\delta\theta$ values.

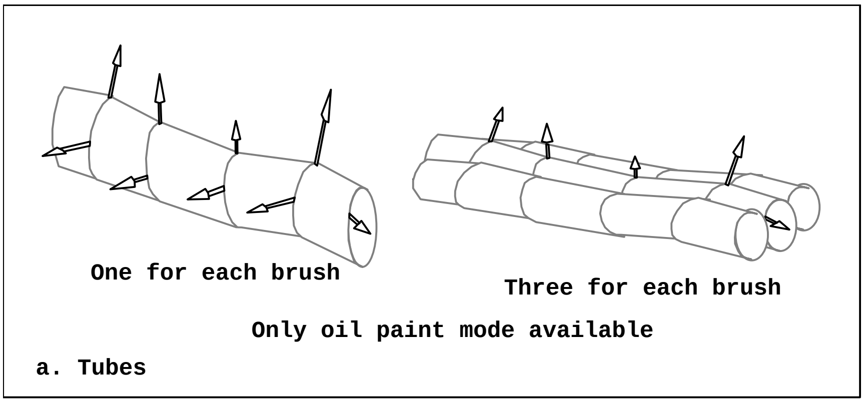
3.3 Description of Material Property of Space

In our applications, instead of general shaders, we use only solid textures for the description of diffuse color. We also use texture map equations in order to describe solid textures. For instance, if we ignore the radius, spherical texture mapping can be used as a solid texture. When we use texture mapping, there will be reconstruction and aliasing artifacts. Since this noise cannot be controlled by painter, it must be eliminated. We avoid these artifacts by approximating continuous image data by using splines [4] through the image sample points.

3.4 Paint Objects

The paint shapes used in our system are quadrilaterals and tubes. Both are given by a set of triangles.

- *Tubes.* Tubes are cut cones which are only used in oil paint mode. Examples of tubes in oil paint mode are shown Figure 5.a. The tubes can be rendered in two different ways.
    - *Constant diffuse reflection coefficient.* In this case, all the triangles that describe a tube have the same diffuse reflection coefficient. In this case, the rendered tube does not give any information about the inside and outside of the surface.
    - *Variable diffuse reflection coefficient.* In this case, the diffuse reflection coefficient of a triangle that approximates a tube is computed by using the cosine of the angle between the triangle's own normal and target surface gradient. In this way, the color of a triangle facing the inside of the target surface is guaranteed to become darker than the color of the triangles facing outside. In other words, using a variable diffuse reflection coefficient helps to differentiate between the inside and outside of the surface. Most of the images in this paper are rendered using a variable diffuse reflection coefficient.
- *Quadrilaterals.* Quadrilaterals are used in both oil paint and airbrush modes. In oil paint mode, quadrilaterals create the effect of ribbon or tape. They are always placed perpendicular to the surface normal and gradient vectors at the vertices are used as vertex normals for the shading. Examples of quadrilaterals in airbrush and oil paint modes are shown Figure 5.b.



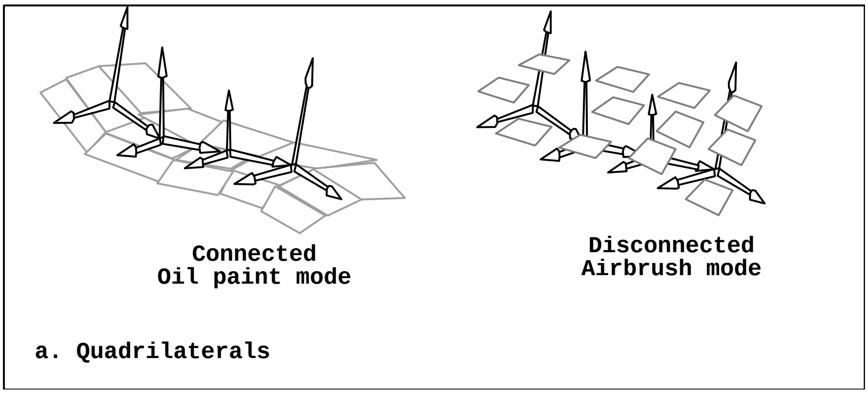


Figure 5: Examples of paint objects.

In oil paint mode, the volume painter can choose to interpolate the material properties along the connected pieces. This interpolation makes noise more natural. Most of the 'oil paint' images in this paper are rendered by interpolating the material properties.

In oil paint mode, the angle $\theta(t)$ should never be bigger than $\pi/2$ since obtuse angles create overlapping paint objects as shown in Figure 6.

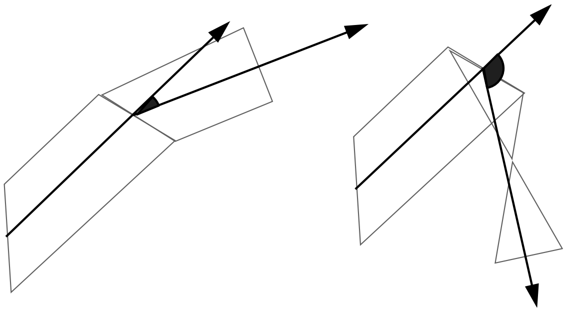


Figure 6: Angles bigger than $\pi/2$ creates overlapping objects in oil paint mode.

During the volume painting process, the painter is able to change the number of tubes and quadrilaterals, and their sizes and transparencies. It is also possible to change from tubes to quadrilaterals, or even to make the object type change with a control based on the values of the solid texture functions.

The painter also describes the *brush size* by the dimensions of the rectangular prism centered at $p(t)$.

3.5 Noise Creation

In our system, shape and color noise are created in several ways.

- *Noise by speed.* Since faster speed means lower sampling, speed can introduce both shape and color noise. In our system, noise by speed is obtained by increasing the value of $s(t)$.
- *Noise by size of paint objects.* Large object size can create both color and shape noise. This noise is more effective in airbrush mode. In paint mode the large object sizes create undesirable images.
- *Noise by brush sizes.* Brush size is given by the dimensions of a rectangular prism centered at the current brush position. If this brush size is large, the paint can be placed away

from the target surface, thus creating shape noise. This effect is useful only in airbrush mode. In paint mode the large brush sizes can create undesirable images.

- *Noise by weighted average of material properties.* The weights in computing a weighted average of the material property of the three-dimensional position and the material property of the paint object control the amount of noise. Since the material properties of paint objects are created randomly, by given more weight to the material properties of paint objects, more noise can be introduced. In our system, volume painter can change these weights.

- *Noise by uncovered shape.* By not entirely covering a shape, volume painter can introduce shape noise. In our prototype system, completely covering one target surface depends on the chosen $s(t)$ and $\delta\theta$ values of equations 2, 3 and 4. A small $s(t)$ value provide a better approximation of the surface but it takes longer time to cover the whole surface. Likewise, choice of $\delta\theta$ values affect the time to cover the surface. In addition, the surface area of target surfaces and paint size affect the coverage time.

4 Implementation

Our volume painter is a simple menu driven interactive system which is implemented in C using the GL graphics library. The system also supports stereo viewing with stereo glasses. The following figures demonstrate the capabilities of our volume painting system. Each one of the images in these figures were created in less than one minute on an SGI O2 workstation. They are direct screen captures during the painting process.

Figure 7. This figure shows toroidal layered surfaces which are painted by changing a layer parameter during the volume painting process. Paint objects are tubes, each brush carries seven tubes and each tube has a constant diffuse reflection coefficient.

Figure 8. In this figure, the layered surfaces that resemble a face are described by a set of exponential functions. Similar exponential functions are used to define the solid texture that produces the colors of the facial features. Figure 8.a shows an example of shape noise. Paint objects are tubes, each brush carries seven tubes and each tube has a constant diffuse reflection coefficient. Figure 8.b is obtained by layering several surfaces. Each

layer surface is partially covered in order to reveal inside layers. Figure 8.b also shows ripples around high curvature areas such as the nose, ear and mouth. Since, in Figures 8.a and 8.b material properties along the connected pieces are not interpolated, we see sudden changes in color. In all the figures except this one, the material properties are interpolated in oil paint mode. As it can be seen in other figures such as Figure 15.b, this interpolation makes noise more natural. Figure 8.c shows the low pass effect of painting outside layers with transparent quadrilaterals. Note that all the details seen in Figure 8.b are now covered.

Figure 9. This figure shows examples of shape noise, where the noise is achieved by not entirely covering the surface. The layered surfaces are spheres. The solid texture is defined by parallel projection of a portion of the Mona Lisa. The images in Figure 9.a are painted by using only one connected tube for each brush and each tube has a variable diffuse reflection coefficient. During the painting process the layer parameter was not changed. In Figure 9.b everything is exactly similar to the one in Figure 9.b except that it is painted by using *seven* connected tubes for each brush instead of one. Furthermore, the radius of each tube is smaller than the radius of the tube used for creating Figure 9.a.

Figure 10. In figure 10.a we see several examples of partly covered layered surfaces. The layered surfaces are spheres and the solid texture is defined by parallel projection as in Figure 9. The images are painted by using seven connected tubes for each brush and each tube has a variable diffuse reflection coefficient. During the painting process, both the sizes of tubes and the layer parameter were continuously changed. If this object is viewed from another direction, the image of the Mona Lisa becomes unrecognizable. Figure 10b. shows a detailed example.

Figure 11. This figure shows the painting effect in airbrush mode. Noise is generated by using large quadrilaterals. The layered surface description and solid texture are the same as in Figures 9 and 10. The lower image shows a detail of the first image.

Figure 12. Figure 12.a shows the noise which is created by using fast speeds for brushes. The first image is painted by using connected tubes, the second one is created by using connected quadrilaterals in oil paint mode. The solid texture is defined by a spherical texture map equation. Figure 12.b shows that the noise in Figure 11a can be reduced

by using slower speeds for brushes. Figure 12.b also gives a comparison of oil paint and airbrush modes. The upper image in Figure 12.b is painted by using one connected tube for each brush and each tube has a variable diffuse reflection coefficient. The lower one is created in airbrush mode with quadrilaterals.

Figure 13. Figure 13.a shows examples of controlling the brush type with a solid texture. During the painting process if the solid texture color is almost equal to the background color (yellow) then the volume painting system uses quadrilaterals as paint objects. Otherwise, it uses tubes. Thus, we obtain different qualities of painting in the same image. The solid texture is defined by a spherical texture map equation. Color noise results from a weighted average of the diffuse color of each paint object and the diffuse color of its position described by the solid texture. Figure 13.b shows an example of this color noise. In the first image, the color noise is almost eliminated by using 0.9 as the weight of the solid texture. In the second image, more noise is introduced by using 0.6 as the solid texture weight.

Figure 14. This figure shows another example of controlling the brush type with solid texture. In this case, if the solid texture color is equal to black then the volume painting system uses tubes as paint objects. Otherwise the paint system uses quadrilaterals. The upper image is obtained by accepting dark gray as black. In the lower image, only dark black is accepted as black.

Figure 15. In these images, the layered surfaces are described by a beta-spline [4] function with three variables. The solid texture is a three-dimensional checkerboard. Figure 15.a shows an example of airbrush mode. The lower image shows in detail the airbrush effect at a contour. Each brush carries nine quadrilaterals and the size of the quadrilaterals is extremely small. Noise results mainly from the sizes of the brushes. Brush size is given by a prism which is much longer in the direction of the surface normal. Figure 15.b shows an example of oil paint mode where the image is painted by using two connected tubes for each brush and each tube has a constant diffuse reflection coefficient. Noise in Figure 15.b is created by a fast brush speed.

5 Discussion

In a volume painting system, the noise can be reduced significantly by covering the surface completely with slower speed, smaller brush and paint sizes. However, exactly like classical painting, the results will look like overworked paintings. Although, there is a close relationship between volume painting and classical painting, volume paintings are different than classical paintings in three ways.

- When painters do not paint a portion of the canvas we can still see the canvas. However, in a volume painting, if a painter does not paint a certain portion of a surface it leaves a hole. A similar effect has been achieved by Marcel Duchamp by painting a large glass [8].
- Volume paintings can be considered as sculptures which can be viewed in three-dimensions using stereo views. Our prototype system provides stereo viewing. Such stereo views especially reveal the potential of volume painting as a new painting paradigm. These paintings can also be transformed into stereo photographic prints.
- In the current implementation of our painting system, we do not preserve geometry information for later use. Instead, we only preserve the images we like. However, the concept volume painting would allow to produce a scene description with detailed geometry and shading information. This scene description could be rendered by using different renderers to get additional effects. The resulting scenes could even be animated by attaching skeletons and deforming the space.

Our choice of implicit equations in the development of the prototype volume painting system was based on well-understood behavior of implicit surfaces. However, it is possible to develop a volume painting system by using either parameteric representations such as tricubic tensor-product splines or polyhedral representations.

Volume painting can be used in visualization of multivariate data by considering it a shader function, $m(p)$, to be sampled by volume painter. Another application of volume painting can be visualization of four dimensional objects by using a more complete solid representation

$$\mathcal{S}(f) = \{ \;\; (x, y, z, a) \;\; | \;\; f(x, y, z, a) = 0 \;\; \},$$

and by still viewing a as a layer parameter.

6 Conclusion

In this paper, we have introduced the concept of volume painting and developed a prototype volume painting system. Despite the fact that our system provides only a small subset of the features that can be provided, it was still possible to easily create interesting images in a short time.

References

[1] Names of Authors are omitted, This paper is included as a supplement, "Implicit Painting of CSG Solids", 1998.

[2] H. Abelson and A. A. diSessa, "Turtle Geometry", *M.I. T. Press,* Cambridge, 1982.

[3] R. Arnhaim, "Art and Visual Perception, A Psychology of the Creative Eye", *University of California Press,* Berkeley, Los Angeles and London, 1974.

[4] R. H. Bartels, J. C. Beatty and B. A. Barsky, "An Introduction to Splines for use in Computer Graphics and Geometric Modeling", *Morgan Kaufman Publishers,* California, 1987.

[5] J. I. Blinn, "A Generalization of Algebraic Surface Drawing", *ACM transaction on Graphics,* vol 1, no. 3, pp. 235-256, 1982.

[6] J. Bloomenthal and B. Wyvill, "Interactive Techniques for Implicit Modeling", *Computer Graphics (1990 Symp. on Interactive Techniques,* vol 24, no. 2, pp. 109-116, 1990.

[7] J. Dorsey and P. Hanrahan, "Modeling and Rendering of Metallic Patinas", *Computer Graphics,* vol 30, no. 3, pp. 387-396, 1996.

[8] M. Duchamp, "The Bride Stripped Bare by Her Bachelors, Even", *Painting made by oil, varnish, lead wire, lead foil and dust on two glass panels,* 1915-1923.

[9] K. Fleisher, D. Laidlaw, B. Currin and A. Barr, "Cellular Texture Generation", *Computer Graphics,* vol 29, no. 3, pp. 239-248, 1995.

[10] P. A. Firby and C. F. Gardiner, "Surface Topology", *John Wiley and Sons,* NewYork, 1982.

[11] P. E. Haeberli, "Paint by Numbers", *Computer Graphics,* vol 24, no. 3, pp. 207-214, 1990.

[12] N. Green, "Voxel Space Automata", *Computer Graphics,* vol 23, no. 4, pp. 175-184, 1989.

[13] W. Januszczak, M. Beal, E. Bowes, A. Callen, S. Hackney, J. McCleery, C. Meredith, C. Villers, N. Watkins, J. Collins, J. WElchman, D. Chandler and D. A. Anfam. "Techniques of The Great Masters of Art", *Chartwell Books, Inc.* New Jersey, 1985.

[14] B. J. Meier, "Painterly Rendering for Animation", *Computer Graphics,* vol 30, no. 3, pp. 477-486, 1996.

[15] P. Prusinkiewicz and A. Lindenmayer, *Algorithmic Beauty of Plants,* Springer Verlag, New York, 1990.

[16] d'Arcy Thomson, *On Growth and Form,* Universty Press, Cambridge, 1952.

[17] A. P. Witkin and P. S. Heckbert, "Using Particles to Sample and Control Implicit Surfaces", *Computer Graphics,* vol 27, no. 3, pp. 269-277, 1994.

[18] G. Wyvill and A. Trotman, "Ray Tracing Soft Objects", *CG International'90: Computer Graphics Around the World,* Editors: T. S. Chua and T. L. Kunii, Springer Verlag, pp. 467-476, 1990.