Trans-Polygon Stroke Method for Frame Coherent Pastel Images

K. Murakami and R. Tsuruno

Graduate School of Kyushu Institute of Design, Fukuoka, Japan

Abstract

We propose Trans-Polygon Stroke Method (TPSM) for creating pastel-like animation that keeps frame-to-frame coherence. There are several variable factors in hand-drawn pastel such as paper roughness and pigments. When these factors are simulated in computer graphics animation, they cause flickers, which reduces visibility. To increase the visibility of pastel-like animation by reducing the flickers, it is need to fix the factors. The procedure of the TPSM is to (1) model objects with quadrilateral polygons, (2) generate particles on the polygons, (3) give a vector direction to each particle, and (4) draw a line from the particle to the n-th polygon along with the direction. Besides the TPSM, the amount of pigments used for one stroke is read from a given table to fix this variable factor. We demonstrate several types of drawn strokes, such as hatching, stipple, and blending, using the proposed method.

Categories and Subject Descriptors (according to ACM CCS):I. 3.3[Computer Graphics]: Display algorithms

1. Introduction

We have proposed a method for generating pastel like strokes that enables for the production of realistic images by rendering a 3D model[1]. However, the approach has not been applied to animations.

One of the major problems with the production of Non-Photorealistic Rendering (NPR) animations is that of frame-to-frame coherence. In the animation of real paintings, the locations of the individual strokes and support media change in each frame[2]. This results in incoherence, which can be observed as flickering. It also occurs in an animation of NPR, but that can be reduced by consider how to generate strokes. And flicker-free hand drawn like animation is generated only by CG.

The purpose of this research is to create a pastel drawing -like animation system that maintains frame-to-frame coherence. Related works are discussed in the next section. The generation of pastel-like strokes whilst maintaining frame-to-frame coherence is discussed in section 3 and the results of test animations are presented in section 4. This is followed by a set of conclusions and recommendations for future work.

2. Related works

There are many researches about frame coherent NPR[3-10]. Our previous research is one of the NPR, pastel-like CG strokes. The feature of pastel drawings are (1) irregularities on paper surface is appear to a picture strongly, (2) pigments that put on paper surface can remove easily by techniques such as blending. We realized computer generated pastel-like rendering for still images[1]. Frame-to-frame transition method of pastel strokes and pigments must be developed to expand it from still images to animation.

Meier proposed a method to keep frame-to-frame coherence by distributing and fixing particles on polygons and generating paint-like strokes from the positions of the particles[3]. This method requests users to define the stroke orientations, but we define them automatically by using quadrilateral polygons and make pastel-like strokes considering pigment and paper texture.

Mapped stroke texture methods are the methods that create pre-calculated strokes and texture strokes on an object.

Praun used a Tonal Art Map to achieve real-time pen strokes without flicker[4]. As this method divides object surfaces into mapping units, it is hard to draw wide strokes such as pastel-like strokes.

Kalnins proposed an interactive system that considered the effect of the paper type. User can select the drawing tool (for example pastel, crayon, pen and pencil), stroke type and hatching pattern to render silhouettes and surfaces of each object[5]. It is hard for this method to blend pigments between different objects because strokes are put on each object, while it is easy for our proposed method.

3. Pastel stroke generation without flicker

3.1. Paper modeling

Physical characteristics of paper must be considered in any study of frame-to-frame coherence because they concern the distribution of pigment significantly. Stroke locations and paper conditions of traditional pastel animations change each frame by frame. Animators often use a frame for continuous several frames to reduce flickers but keep the flickers a little for the pastel-like atmosphere. Same effect can be created in NPR if stroke locations are not considered.

Mapping hand-drawn strokes on an object is one of methods to fix both the stroke location and the amount of pigments on the paper. As this method does not cause any flickers, animation objects look like not drawing objects but paper craft objects (see Figure 1). Like Kalnins's method, our method does not change the physical characteristics of paper and the amount of pigment of a stroke but change stroke location to create drawing impression.

The technique for fixing the location of each stroke is described in the next section.

3.2. Fixing strokes with Trans-Polygon Stroke Method

We define a model consisting of quadrilateral polygons. It is because square means plane that is minimum part when we look an object. So people can draw hand drawn like strokes by put strokes along to quadrilateral polygons.

Suppose to put number 0 to 3 and set *u*-*v* coordinate system along the number to each polygon as in Figure 3. Both *u* and *v* axes assume the corner numbered 1 as the origin and the u or *v* value is [0,1].

Each polygon is assigned this *u*-*v* coordinate system, with a set of particles distributed randomly across the polygon. This is because a global coordinate system cannot be used to the polygons if the connections between the polygons are like the right image of Figure 2.

Distributed particles on each polygon serve as the origins of pastel strokes. This is based on Meier's method[5], except that the target is a quadrilateral. The number of particles per polygon is determined according to the ratio of the stroke density to the polygon area.

The parameters of each particle are:

Coordinates of pastel stroke origin on the polygon *S(su, sv)*,
Vector of the stroke across the polygon *V(a, b)*,

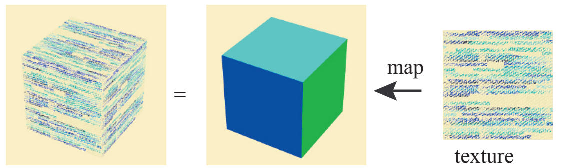


Figure 1. An example of textured box.

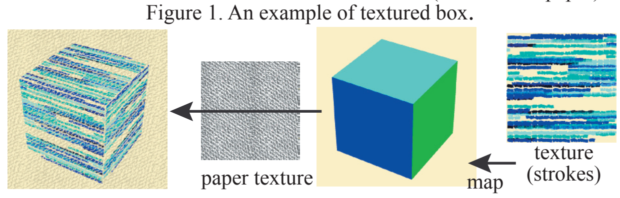


Figure 2. An example of pigment maintaining in strokes by texturing (OpenGL preview).

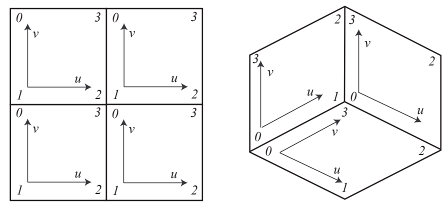


Figure 3. Connections of polygons.

Stroke length *l*,
Stroke color *(r, g, b)*, and
Effect switch (blending on/off).

After all of the particle have been prepared, the point at which the stroke meets the boundary of the polygon *en(enu, env)* is determined from the stroke vector *V*. Stroke length *l* means the numbers of polygons that a stroke is crossing. The stroke is then treated as a series of segments, with stroke properties continuous along the length of the stroke. The process is illustrated in Figure 4.

For strokes with *l*>=1, the stroke is projected backwards along vector -*V* to find the boundary point. We call this point the "projected origin" *B*. The normalized distance from *S* to *B*, *pdist*, is then defined as:

$$\text{if } |a| >= |b| \qquad pdist = |\, su - Bu|$$
$$\text{if } |a| < |b| \qquad pdist = |\, sv - Bv|$$

In this case, stroke length *l* becomes *l+pdist*.

3.3. Fixing pigment

After distribution of the points, pastel-like strokes are generated from the points using our previous method[1]. A stroke is consist of some number of basic strokes. In Figure 4, the stroke that is from *S* to *E*, is generated from strokes that are *S* to *e1*, *e1* to *e2*, *e2* to *e3*, and *e3* to *E*. Each basic stroke is generated according to the following sequence (see Figure 5).

1. Basic line: define basic stroke location from distributed points on polygons, using them as origin and end points of each stroke. The stroke is divided into certain length (a) and described as a sequence of points. Basic line is blurred

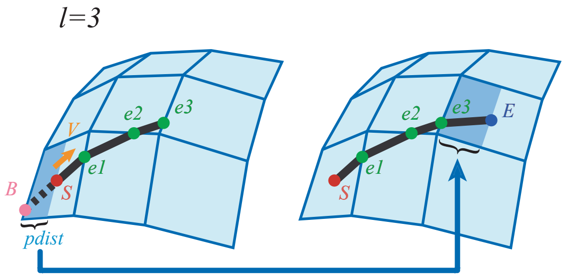
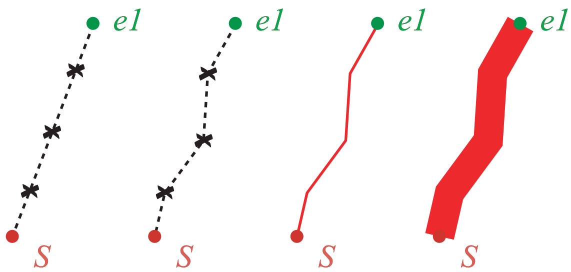


Figure 4. Trans-Polygon Stroke Method.



(a) (b) (c) (d)

Figure 5. Setting strokes.

randomly to represent hand drawn-like effect (b).
2. Interpolation stroke: set pigments along to basic line for interpolate basic line (c).
3. Adding width: make width to the sides of interpolation stroke (d).
4. Additional effect: blending.

Random numbers are used to represent hand drawn-like effects or width changes. Once the random numbers are generated, they are stored in a "random number table" for use in later frames. The random number in each random number table generate a unique and reproducible stroke. Polygon numbers are flagged to recall the correct random number table.

3.4. Simulation of hand-drawn strokes

Several different effects of strokes created by changing parameters described in section 3.2. are shown in Figure 6.

Hatching

Strokes can be made to overlap in order to produce a hatched effect by selecting random stroke vector parameters.
Cross-hatching or hatching by curved strokes is also can be generated by this method.

Stipple

A stippled effect is produced by wide, short pastel strokes. As shown in Figure 6(f), the particle density is higher than that for hatching. Stippling with pastels can be used to create a light effect by the use of mixed colors and a "fluffy" texture.

Blending

A blending effect can be seen in the lower set of images in Figure 6. This effect can be used in animations because stroke locations are fixed on the object. The effect is applied after translation to 2 dimensions, as proposed in [1].

4. Results

An animation was created using the proposed method and examples of the frames are shown in Figure 7. The animation was generated on a 2 GHz Pentium 4 CPU, Windows 2000 PC. The reference model is a glove consisting of 1314 polygons. The parameters were stroke vector $V(a,b)$=(1.0,0.0), stroke length l = 5.0, stroke width = 3.0. It cost about 1 minutes per an image.

Stroke color depends on the original color and shade. Four original colors were used in the animation. Using shade without blending is unusual in pastel drawing, but it enhances the appearance of solidity. Different materials can be represented by changing the stroke orientation. The glove was rotated through 360 degrees about the y-axis.

Another example is shown in Figure 8. These spheres

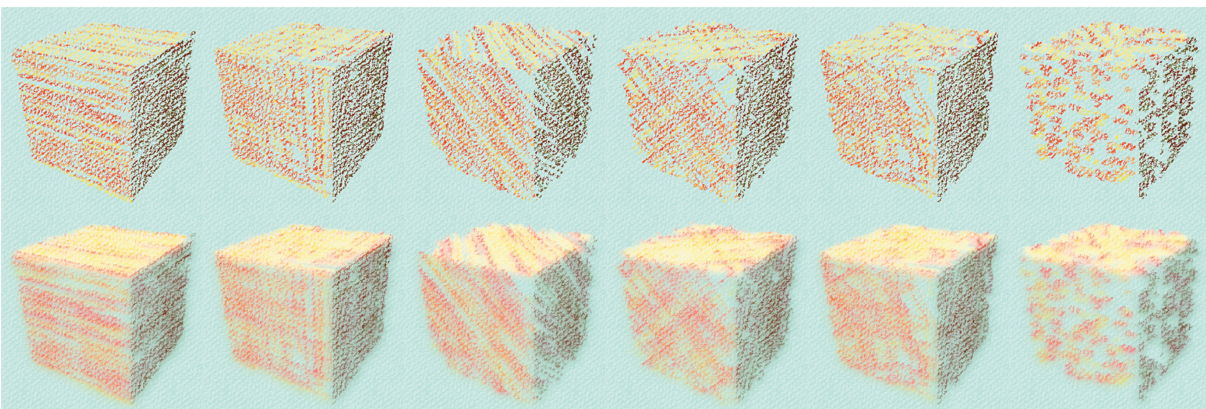

(a) (b) (c) (d) (e) (f)

Figure 6. Simulation of hand-drawn pastel strokes:
(a) Hatching, (b) Cross-hatching, (c) Hatching with diagonal strokes, (d) Cross-hatching with diagonal strokes, (e) Hatcing with curved strokes, (f) stipple, and blending is shown in each image below.

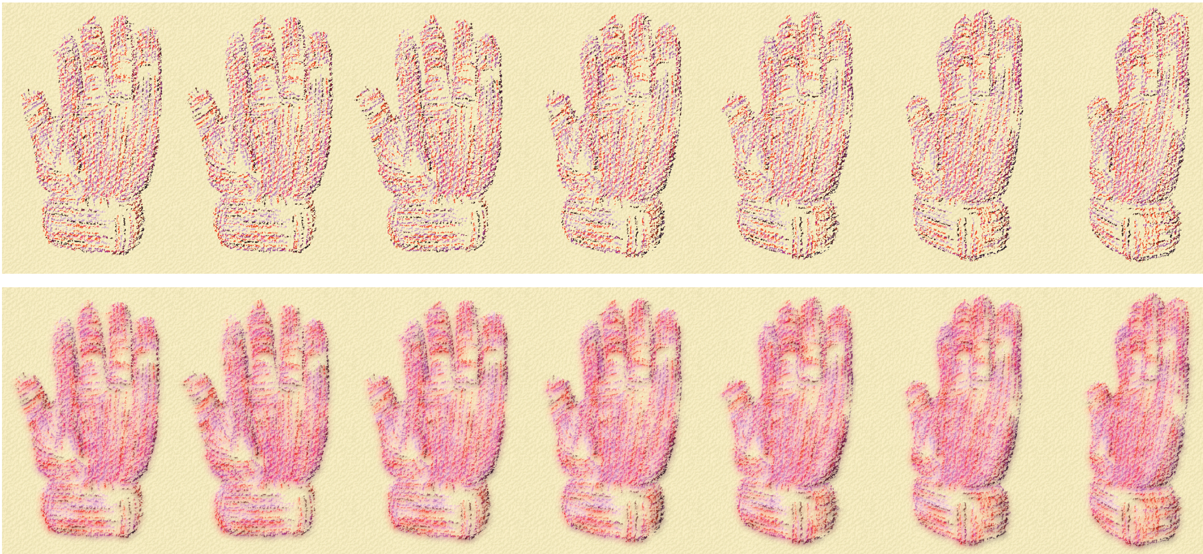


Figure 7. An animation with glove model. <animation1,2>

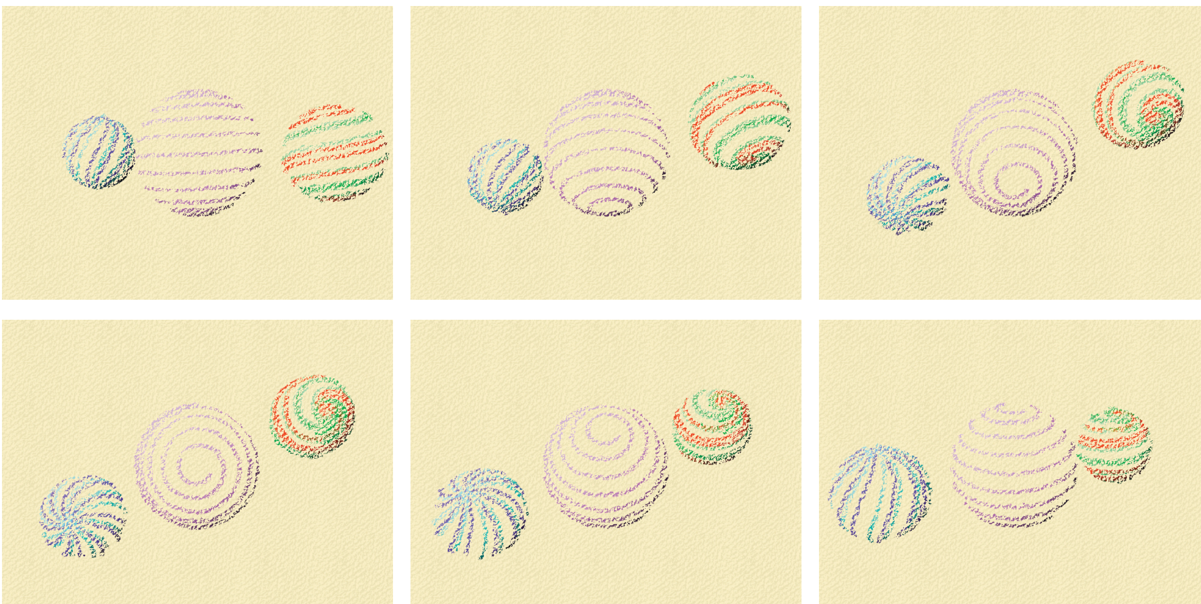


Figure 8. An animation with 3 balls using long strokes. <animation>

were drawn using long spiral strokes. The center example was drawn using a single stroke.

An example of an animation of a more complex model can be seen in Figure 9. This model is consisted of 11692 polygons. Diagonal strokes make these images artistic and frame-to-frame coherence was maintained for all strokes and image planes. It cost 10 minutes per an image.

5. Conclusions

The purpose of this paper was to present a new technique for generating pastel-like animations that considers coherence between frames by maintaining the loacations of the individual strokes. This work builds upon previous research into the generation of realistic pastel-like 2D images. The animation procedure is described in detail and examples are provided to illustrate the success of the approach.

In future researches, the authors will consider applying the technique to textured objects or triangular meshes, the orientation of the stroke vector at the start points, effective positioning and control of quadrilateral polygons, enhancing the method for pigment application and producing further animations.

Figure 9. An animation with Taj Mahal model. From left to right, narrow blend area, zoom, and rotate. <animation>

References

[1] Murakami K. and Tsuruno R.: Pastel-like Rendering Considering the Properties of Pigments and Support Medium, Conference abstracts and applications SIGGRAPH2002, p.227, 2002. 1, 2, 3

[2] Dianne Jackson (director) , The Snowman, Columbia/Tristar Studios, VHS, 1982. 2

[3] Meier B. J.: Painterly Rendering for Animation, Proc. SIGGRAPH 96, pp.477-484, 1996. 2

[4] E. Praun, H. Hoppe and M. Webb: A.Finkelstein: Real-Time Hating, Proc. SIGGRAPH2001, pp.581-586, 2001. 2

[5] R. D. Kalnins, K. L. Markosian, B. Meier, M. A. Kowalski, J. C. Lee, P. L. Davidson, and M. Webb: WYSIWYG NPR: Drawing Strokes Directly on 3D Models, Proc. SIGGRAPH2002, pp.755-762, 2002. 2

[6] Klein A. W., Li W. W., Kazhdan M. M., Correa W. T., Finkelstein A. and Funkhouser T. A.: Non-Photorealistic Virtual Environments, Proc. SIGGRAPH 2000, pp. 527-534, 2000. 2

[7] Oscar E. Meruvia Pastor and Thomas Strothotte.Frame-Coherent Stippling", Eurographics 2002, Volume 21, Number 3, 2002. 2

[8] Girshick A., Interrante V., Haker S. and Lemone T.: Line Direction Matters: An Argument for the Use of Principal Directions in 3D Line Drawings, Proc. the First International Symposium on Non Photorealistic Animation and Rendering (NPAR) for Art and Entertainment, Annecy, France, June 2000. 2

[9] Northrup J. D. and Markosian L., Artistic Silhouettes: A Hybrid Approach, Proc. the First International Symposium on Non Photorealistic Animation and Rendering (NPAR) for Art and Entertainment, Annecy, France, June 2000. 2

[10] Markosian L., Meier B. J., Kowalski M. A., Holden L. S., Northrup J. D., and Hughes J. F., Art-based Rendering with Continuous Levels of Detail, Proc. the First International Symposium on Non Photorealistic Animation and Rendering (NPAR) for Art and Entertainment, Annecy, France, June 2000. 2