Efficient Region-Based Pencil Drawing

Shuo Sun
Tianjin, 300160
China, School of Science
Tianjin Polytechnic University
Sunshuo_0@163.com

Dongwei Huang
Tianjin, 300160
China, School of Science
Tianjin Polytechnic University
tjhuangdw@163.com

Abstract

This paper proposes an extension to the existing automatic pencil drawing generation technique based on Line Integral Convolution (LIC). The original LIC pencil filter utilizes image segmentation and texture direction detection techniques for defining outlines and stroke directions, and the quality of a resulting image depends largely on the result of the white noises and the texture directions. It may fail to generate a reasonable result when the white noises and the texture directions are not consistent with the texture structure of the input image. To solve this problem, we propose in this paper to improve the existed LIC-based method. First, a more accurate and rapid graph-based image segmentation method is introduced to divide the image into different regions. Second, we present a new region-based way to produce white noises and texture directions. We also demonstrate the enhanced LIC pencil drawing is closer to the real artistic style.

KeyWord pencil drawing, line integral convolution, image segmentation, Non-photo-realistic rendering

1. Introduction

Recently, Non-Photo-Realistic rendering (NPR) has become one of the most important research topics of computer graphics. A number of techniques have been developed to simulate traditional artistic media and styles, such as pen and ink illustration[Win94a, Win96a, Sal94a, Sal97a], graphite and colored pencil drawing[Tak99a, Sou99a, Sou99b], impressionist styles[Lit97a], paintings of various materials including oil[Her98a], water color[Cur97a] and so on. The existing researches on painterly image generation mainly take two different approaches. The first approach is to provide physical simulation to the materials and skills, and has been mainly combined with interactive painting systems or 3D non-photo-realistic rendering systems for generating realistic painterly images. The second approach is the painterly filtering, which involves taking an image and applying some kind of image processing or filtering techniques to convert it into an image of a painterly look. While many excellent painterly filtering techniques have been developed for generating brushstroke based paintings [Goo01a], relative few publications can be found on converting a source image into line stroke based drawings. In case of drawing, geometric information such as the outline of regions, the direction and shape of strokes becomes more critical, while it is usually difficult to extract such information from 2D raster images automatically. Instead of modeling line strokes geometrically, Mao etc [Mao01a] have developed a pencil drawing filter using Line Integral Convolution (LIC), a texture based flow visualization technique [Cab93a]. The technique utilized the similarity between the appearance of LIC images and pencil strokes, and succeeded in generating line stroke like images with pixel-by-pixel image filtering. It employs image segmentation and texture analysis technique to automatically detect the outlines and decide the stroke

 ISBN 978-80-86943-98-5

orientations for the images. Then for each region, if it contains directional textures, the texture directions are used as the stroke directions, otherwise a randomly chosen stroke direction is assigned. It may fail to generate a reasonable result when the white noises and the texture directions are not consistent with the texture structure of the input image. To solve this problem, we propose in this paper to improve the existed LIC-based method. First, a more accurate and rapid graph-based image segmentation method is introduced to divide the image into different regions. Second, we present a new region-based way to produce white noises and texture directions.

We realize automatic generation pencil drawing from 2D images. There are several advantages of our method comparing to the existed. First, according to the characteristics of the pencil drawings produced by the artists, different areas have different texture expressions. General speaking, some areas might have very complicated textures and others might be faint in textures, and that we are desire to achieve this kinds of artistic effects. Through controlling the noises of the different areas, we can not only express the shade areas of the texture but also hold some subtle area and necessary blank area in the image. The result is more similar to the artistic style.

The remainder of this paper is organized as the follows: Section 2 gives a short survey on related work. Section 3.1 introduces the original LIC-Based pencil filter method. Section 3.2 describes the algorithm graph-based image segmentation and Section 3.3-3.4 introduces our region-based pencil filter method. Section 4 concludes the paper and gives some examples.

2. Related Works

Pencil drawing has been an important topic since the beginning of painterly image generation research history. In an early 2D painting system called Pencil Sketch [Ver89a], a mouse based virtual tablet is provided for allowing users to interactively specify a set of parameters, such as the hardness of pencil, the pressure applied to a pencil, and the orientations of strokes. Recently, Sousa and Buchanan developed several pencil drawing rendering techniques based on an observation model of pencil drawings [Sou99a, Sou99b]. They built the models of pencil, paper and how lead pencils interact with drawing paper through a careful investigation of the real pencil drawings using scanning electron microscope. When the parameters of those models and the strokes are specified, a 2D image can be converted into a pencil drawing. 3D polygon models can be automatically rendered into pencil drawings by referring to the tone value lookup table for the parameter values of the models. Takagi and Fujishiro proposed to model the paper microstructure and color pigment distribution as 3D volume data and use volume ray-tracing for rendering color pencil drawings [Tak99a]. Other existing painterly image generation techniques closely related with our work are probably those successful works on pen–and-ink illustrations [Win94a, Win96a, Sal94a, Sal97a]. In their interactive systems, pen-and-ink illustrations can be generated either from 3D models or 2D images by using a set of pre-stored stroke textures. The largest difference between our technique and all these existing techniques is that our technique can generate a pencil drawing from a source image in a completely automatic way while all these existing techniques rely, to certain extent, on user interventions, for specifying the attributes and directions of strokes. Several commercial packages provide some filters for creating pencil drawing effects. For example, Jasc Paint Shop Pro software supports a black pencil filter. However, to obtain a satisfactory result with those filters, a user usually needs to combine the effects of many other filters and explore the best generation process experimentally through trial and error for many times.

3. LIC Pencil Filter

3.1. LIC Algorithm

Line Integral Convolution (LIC) is a texture based vector field visualization technique [Cab93a]. As shown in Figure 1, it takes a 2D vector field and a white noise image as the input, and generates an image which has been smeared out in the direction of the vector field through the convolution of the white noise and the low-pass filter kernels defined on the local streamline of the vector filed.

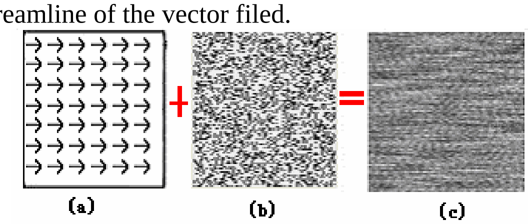


Figure 1 Line Integral Convolution (LIC) (a) Input vector field; (b) Input white noise; (c) Output result

The images in figure 2 show the basic algorithm of the LIC. The inputs are the vector fields and white noises.

 ISBN 978-80-86943-98-5

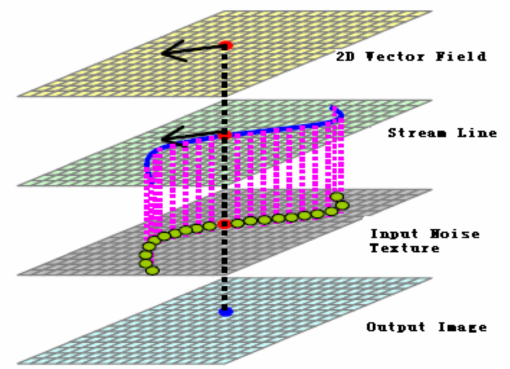


Figure 2 the basic algorithm of the LIC

P is the output pixel, $\rho(\tau)$ is stream line $(-L \le \tau \le L)$, L is the half length of the stream line。$T(\rho(\tau))$ is the noise texture value in the stream line, $K(\tau)$ is a convolution kernel. So the pixel value in P is $T'(\rho(0))$:

$$T'(\rho(0)) = \frac{\int_{-L}^{L} k(\tau) T(\rho(\tau)) d\tau}{\int_{-L}^{L} k(\tau) d\tau}$$

The discrete form describes as follows: p_i is the discrete point in the stream line; W_i is the contribution of *pi* to P, namely the area that $K(\tau)$ cover between p_{i-1} with pi.

$$T'(P) = \frac{\sum_{i=0}^{N} T(p_i) W_i}{\sum_{i=0}^{N} W_i}$$

The idea of using LIC for pencil drawing generation was inspired by the visual similarity of LIC images and pencil drawings. As an LIC image is obtained by low-pass filtering a white noise along the local streamlines of a vector field, we can observe the traces of streamlines along which intensity varies randomly. Such traces have a similar appearance of pencil strokes where the variance of intensity is caused by the interaction of lead material and the roughness of paper surface. Figure 3(a) is a digitized sample of a typical imitative tone used in real pencil drawings. Figure 3(b), presents the very similar features as the tone image by LIC processing.

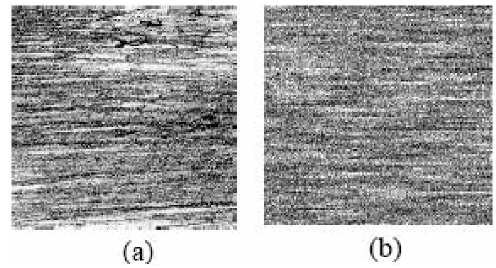


Fig.3: Comparison of a real pencil drawing (a) and an LIC texture (b).

3.2. The existed LIC pencil drawing method

In general, for producing a pencils drawing from a 2D source image, several steps are done. First we generate a white noise from the source image, then the original image is divided into different region and the boundary is extracted. Next we generate the vector field representing the orientation of strokes, and produce pencil drawing by applying LIC to the white noise and the vector field. Figure 4 depicts the algorithm of original LIC pencil filter. It converts a 2D source image into a pencil drawing in the following seven steps:

1. Generate a white noise (Figure 4(b)) from the source image (Figure 4(a)).
2. Segment the input image (Figure 4(a)) into different regions (Figure 4(c)).
3. Extract region boundary (Figure 4(d)).
4. Generate the vector field (Figure 4(e)) representing the orientation of strokes.
5. Generate stroke image (Figure 4(f)) by applying LIC to the white noise (Figure 4(b)) and the vector field (Figure 4(e)).
6. Add the boundary (Figure 4(d)) to obtain the drawing with outlines (Figure 4(g)).
7. Composite the resulting image (Figure 4(g)) with the paper sample (Figure 4(h)) to obtain the finished pencil drawing (Figure 4(i)).

 ISBN 978-80-86943-98-5

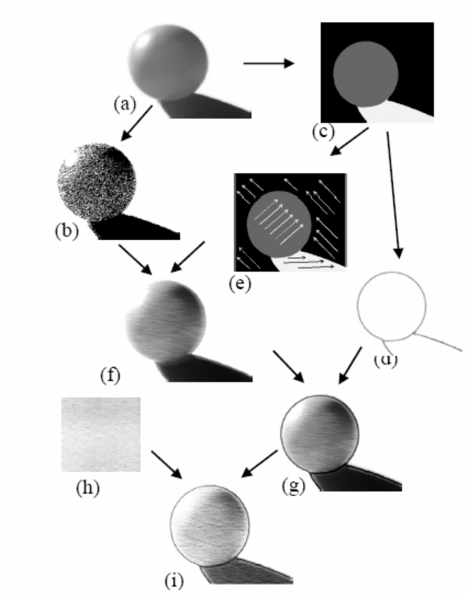


Figure4. The existed LIC algorithm (the image comes from [Mao01a])

4. Enhanced Region-based LIC Pencil Filter

4.1. Graph-Based Image Segmentation

A well used technique in pencil drawing for conveying the 3D shapes of objects and spatial relationship among different objects in a scene is to emphasize the boundary between two different regions by drawing outlines or changing the appearance of strokes in the two regions. To create such effect, we propose to divide the input image into different regions using existing image segmentation technique. In our current implementation, a Graph-Based image segmentation technique [Fel04a] is used for the region extraction. Contrasting to the method in [Mao01a], this method can dramatically promote the performance of our pencil filter. As shown in figure 5, the left image is segmented into many areas which represent the respective color distributions for each region.

Figure 5. Graph-Based image segmentation

Graph-based image segmentation techniques generally represent the problem in terms of a graph G = (V;E) where each node $v_i \in V$ corresponds to a pixel in the image, the edge set E is constructed by connecting pairs of pixels that are neighbors in an 8-connected sense (any other local neighborhood could be used). A weight is associated with each edge based on some property of the pixels that it connects, such as their image intensities. Neighbor edges are clustered into a forest, and each tree in the forest is related to a minimum spanning tree (MST). Finally each MST is a sub-area. To judge whether two trees can be merged into one tree, a predicate is defined. The predicate expression is showed as follow.

$$D(C_1,C_2)=\begin{cases} True.\ if(Dif(C_1,C_2)>MinT(C_1,C_2)) \\ False.\ otherwise \end{cases}$$

$D(C_1,C_2)$ means the merging predicate of the areas of C_1 and C_2, $Dif\ (C_1,\ C_2)$ means the difference between the area C1 and the area C2. $MinT\ (C_1,\ C_2)$ is the minimum internal difference.

The advantage of the method is that the accuracy of the image segmentation can be adjusted by users and some details of the certain regions can be ignored. This will improve on the effect of the pencil drawing. In addition, this results in a graph with $O(n)$ edges for n image pixels, and an overall running time of the segmentation method of $O(n \log n)$ time.

4.2 Region-based Noise Production

The white noise image is generated in a way that the probability a white value is set for a pixel is proportional to the intensity level of the corresponding pixel in the input image. The gray-scale tone of a resulting pencil drawing is mainly decided by the white noise image. To match the tone between the input image and the resulting pencil drawing, we use the tone of the input image to guide the distribution of noise.

An important characteristic of the pencil drawing is its ability to preserve detail in low-variability image regions while ignoring detail in high-variability regions. The input image is then divided into many small regions which have corresponding meanings. The method mentioned in [Mao01a] dealt with the noise according to uniform criterion. The result noises would be failure to distinguish the important elements and the unimportant elements, when the range of the intensity of the image is small. Our region-based method solves the question through dynamic adjusting the threshold value of different areas. It is very important to pencil drawing. We simply introduce the algorithm.

 ISBN 978-80-86943-98-5

Because the processing of LIC is global, so some local regions with exquisite texture will be destroyed. In addition to, as a kind of sketch drawing, pencil drawing should keep some blank areas from the point of aesthetics. So we need to segment the original image into different region through above graph-based method, then we deal with the each area through giving certain threshold value. Commonly we select the average gray intensity of the certain regions as the threshold values.

Our method improves the producing way of white noise. We deal with the original gray image according to different gray intensity range in the respective regions. So the contrasts between light and shade in the results are more eminent and more similar to the artistic style. The formulation of white noise is showed as follows.

Let I_{input} be the intensity of a pixel in the input image, and P is a floating-point number generated with a pseudo-random function; R_i is the average intensity in the ith regions, and it is selected as the threshold to control the distribution of the noise. Then the intensity I_{noise} of the corresponding pixel in the noise image is decided in the following way:

$$I_{noise} = \begin{cases} I_{noise1} = \begin{cases} \max 1 : (if(P > T_1)) \\ \min 1 : (else) \end{cases} (if(Input \le R_i)) \\ I_{noise2} = \begin{cases} \max 2 : (if(P > T_2)) \\ \min 2 : (else) \end{cases} (if(Input > R_i)) \end{cases} \quad (P \in [0,1])$$

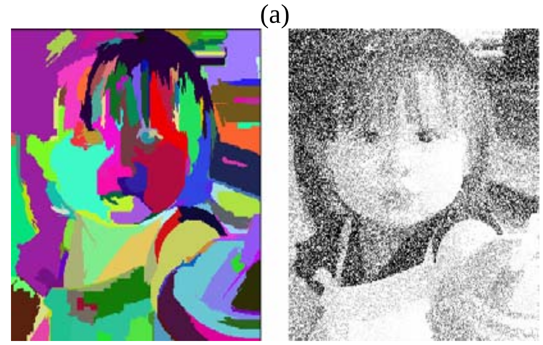
Here max1 and max2 are the maximum gray values of the output pixels, which usually are 255; and min1 and min2 are the minimum gray values of the output pixels, which usually are R_i. However, we can adapt these values to fit for the whole tone of the pencil drawing.

$$T_1 = k_{i,1}(1 - \frac{I_{input}}{255}), T_2 = k_{i,2}(1 - \frac{I_{input}}{255})$$
$$k_{i,1} = \lambda_1(1 - \frac{I_{input}}{255}), k_{i,2} = \lambda_2(1 - \frac{I_{input}}{255})$$
$$\lambda_1, \lambda_2 \in [0,1]$$

According to our method, λ_1 and λ_2 are two experiential values. Our initial experiment result suggests that a default value of 0.7 and 0.3 produce a visually acceptable result for most scenes. We also allow users to interactively adjust the values of λ_1 and λ_2.

Through above method, the noise of each area can be controlled to match the features of the pencil drawing to a great extent. As shown in figure 6, the picture (a) show the noise produced by old way, and the picture (b) is produced by our method. Obviously, the result noise dealt with by our method can reflect the nature of the pencil drawing. Some areas need to be emphasized corresponding to more dense noises, and some areas need to be fade out corresponding to sparse noises.

(a)



(b)

Figure 6 The existed noise algorithm (a), and Region-Based noise method (b).

4.3.Stroke Orientation

When using imitative tone, the direction of strokes is an important factor contributing to the impression of a pencil drawing. The stroke direction is also important for conveying the shapes and textures of objects. If a texture presents a directional feature, then strokes should be oriented in a direction matching that of the original texture. For example, when we paint human hair, we would have our pencil follow the direction of hair strands. To produce such effect, Mao etc [Mao01a] implemented a Fourier texture analysis technique for extracting the local texture directions. The basic principle of the technique is that if a texture presents a specific direction, then after transform the texture into frequency domain, the power spectrum should have large values in the direction orthogonal to the direction of the texture. They deal with each pixel

 ISBN 978-80-86943-98-5

in the input image and compute respective texture direction by the analysis of the FFT.

The method mentioned above is very time-consuming, and they don't consider the local structure features of the image. Usually, some regions have similar directions of the strokes, and these areas can be dealt with the uniform direction according to artistic style. On the other hand, the directions of the strokes vary dramatically in some regions, so we keep the results of the respective directions, and then the details features in the area should be represented. As shown in figure 7, the direction of hair strands is important in (a) and the details of the cloud should be depicted in (b).

(a)

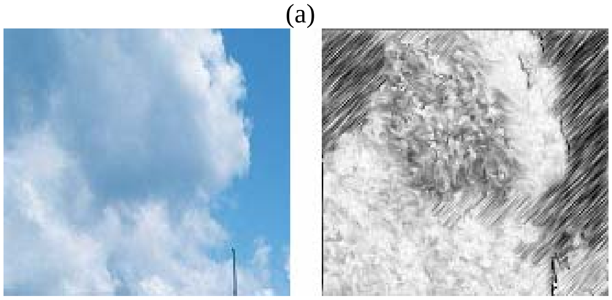


Figure 7 the direction of hair strands (a), and the details of the cloud (b).

Here we propose a method to produce the direction of the stroke according to the features in the regions. First, the orientation vectors of the pixels are calculated by sobel operator. Then a criterion is presented to decide the direction of the stroke.

Let $V_{i,j}$ ($j = 0,1,2,... m_i$) be the field vector of the jth pixel in the ith area. It is computed by sobel operator. Here m_j is the numbers of the pixels in the jth area.

Then the mean of the field vectors in each area is calculated. At the same time, the CV (Coefficient of Variation) is computed to reflect the orientation's distribution.

$$V_i = (\sum_{j=0}^{m_i} V_{i,j}) / m_i$$

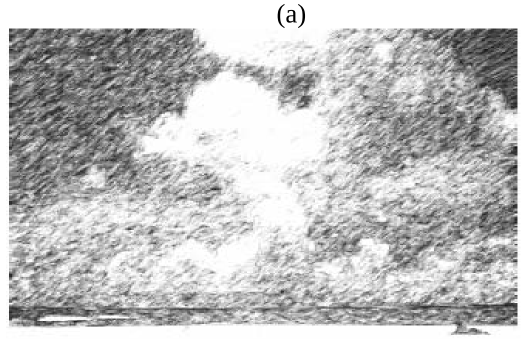
$$D_i = \sum_{j=0}^{m_i} (V_{i,j} - V_i)^2$$

The final field vector of the each pixel is determined through the following criterions:

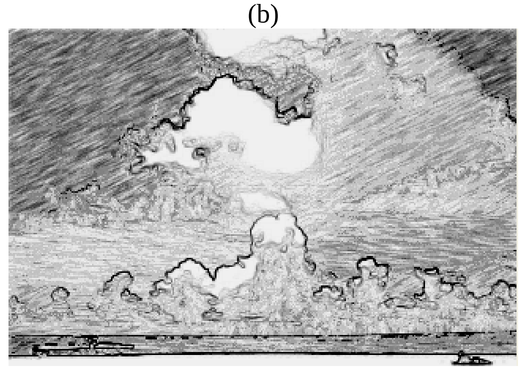
$$FV_{i,j} = \begin{cases} V_{i,j} : if\,(D_i / m_i > T) \\ V_i : (else) \end{cases}$$

Here *T* is a given threshold. Experiments suggest that a default value of 0.7 produce a visually acceptable result for most scenes.

(a)



(b)



(c)

Figure 8. The contrasts of the stroke directions

Figure 8 (a) is original image, and (b) is the result in [Mao01a]. The picture (c) is our result. Obviously,

 ISBN 978-80-86943-98-5

our result not only reflects the details in some areas but also embody the style of the pencil drawing.

5. Experiment results

We have implemented all the techniques described above and built an automatic pencil drawing generating system on Windows environment. Basically, after the input image is specified, a pencil drawing style image will be generated automatically with a set of default value and other information derived by the system from the input image. Users are also allowed to interactively specify some parameters, such as the parameters for controlling the region tone of the output image, the threshold used for graph-based image segmentation and the length of convolution.

Comparing to the existing method in [Mao01a], the new algorithm is much faster than the existing one and more similar with the artistic style of the pencil drawing. The existing LIC pencil filter took about 60 seconds to generate an image of size 350*280 on a Pentium IV PC. Our method only require about 5 seconds. For an image of size 1024*768, our method takes about 20 second, and the more important feature contrasting to the method in [Mao01a] is that our method dynamically analyzes the texture structure of the regions. So it is effective and lifelike. Several results of the pencil drawing as shown in figure 9.

Figure 9. Several results of the pencil drawing

6. Conclusion

We propose in this paper to improve the existed LIC-based method. First, a more accurate and rapid graph-based image segmentation method is introduced to divide the image into different regions. Second, we present a new region-based way to produce white noises and texture directions. The new algorithm is much faster than the existing one and more similar with the artistic style of the pencil drawing. We believe the purpose of NPR systems is not to replace artists, but rather to provide a tool for users with no training in a particular medium. Potential application fields of our technique include producing posters from photo graphs, processing videos into pencil drawing style animations, obtaining the preparatory sketches for creating paintings of other styles, and so on.

 ISBN 978-80-86943-98-5

The major future research directions include the realization of abstraction and focusing, establishment of the curved strokes, extension to colored pencil drawing and application to non-photorealistic rendering of 3D models and scenes.

7. Acknowledgements

The work reported in this article is supported by the National Natural Science Foundation of China (No.10472077, No.50375107).

8. References

[Cab93a] B. Cabral and C. Leedom, “Imaging Vector Field Using Line Integral Convolution”, *SIGGRAPH93 conference Proceeding*, pages 263-270, 1993.

[Cur97a] C. J. Curtis, S. E. Anderson, J. E. Seims, Kurt W. Fleischer, and David H. Salesin, “Computer-Generated Watercolor”, *SIGGRAPH 97 conference proceeding*s, pages 421–430, 1997.

[Fel04a] P.F. Felzenszwalb, D.P. Huttenlocher “Efficient Graph-Based Image Degmentation” *International Journal of Computer Vision[J]*, 2004

[Goo01a] Gooch, *Non-Photorealistic Rendering,* AK Peters Ltd, 2001.

[Her98a] A. Hertzmann, “Painterly Rendering with Curved Brush Strokes of Multiple Sizes”, *SIGGRAPH 98 conference proceeding*s, pages 453–460, 1998.

[Lit97a] P. Litwinowicz, “Processing Images and Video for An Impressionist Effect”, *SIGGRAPH97 conference proceedings*, pp.407-414, 1997.

[Mao01a] X. Mao, Y. Nagasaka and A. Imamiya", Automatic Generation of Pencil Drawing from 2D Images Using Line Integral Convolution", Proceedings of the Seventh International Conference on Computer Aided Design and Computer Graphics CAD/GRAPHICS2001, PP. 240-248, 2001. *conference proceedings*, pages 138-143 , 1989.

[Sal94a] M.P. Salisbury, S. E. Anderson, R. Barzel, and D. H. Salesin, “Interactive Pen–And–Ink Illustration”, *SIGGRAPH 94 conference proceedings,* pages 101–108, 1994.

[Sal97a] M. P. Salisbury, M. T. Wong, J. F. Hughes, and D. H. Salesin, “Orientable Textures for Image-Based Pen-and-Ink Illustration”, *SIGGRAPH 97 Conference Proceeding*s, pages 401–406, 1997.

[Sou99a] M.C.Sousa and J.W.Buchanan, “Observational Model of Blenders and Erasers in Computer-Generated Pencil Rendering”, *Graphics Interface '99 conference proceedings,* pages 157-166, 1999.

[Sou99b] M.C.Sousa and J.W.Buchanan, “Computer-Generated Graphite Pencil Rendering of 3D Polygonal Models”,*EUROGRAPHICS '99 conference proceedings*, pages 195-207, 1999.

[Tak99a] S. Takagi, I. Fujishiro and M. Nakajima, “Volumetric modeling of colored pencil drawing,” *Pacific Graphics'99 conference proceedings,* page 250-258, 1999.

[Tam84] H. Tamura Edited, “Introduction to Computer Image Processing (In Japanese)”, *Soken Shuppan Publisher Inc*. 1984, ISBN4-7952-6304-3.

[Ver89a] A. H. Vermeulen and P. P. Tanner, “Pencil sketch—A Pencil-Based Paint System”, *Graphics Interface '8.*,1989

[Win94a] G. Winkerbach, D. H. Salesin, “Computer-Generated Pen-and-Ink Illustration”, *SIGGRAPH94 conference proceedings*, pages 91-100, 1994.

[Win96a] G. Winkenbach and D. H, Salesin, “Rendering Parametric Surfaces in Pen and Ink”, *SIGGRAPH96 conference proceeding*s, pages 469–476, 1996.