

A Multi-parameter Generating Method for Colored Pencil Rendering

Shi Xiao Nan*

Department of Computer Science and Technology, Xi'an University of Science and Technology, Xi'an, China

Abstract

Due to the existing algorithm of pencil drawing mainly focuses on automatically generated the images, and rendering with parameters is researched little. In view of this, it developed a colored pencil drawing rendering algorithm with multi-parameter in this paper. It presented an improved pencil texture generation algorithm by setting 6 parameters for pencil rendering, and using wavelet methods to obtain low frequency image information, after adding white noise and calculated angle matrix, it used motion blur filter to get pencil texture, and formed a White-Black pencil drawing with superimposing of contour, finally it used the three color space conversion of color space integration to generate final colored pencil drawings. The experimental results demonstrate that the algorithm can obtain a satisfactory colored pencil drawing, and keep the color fidelity.

Keywords

Colored Pencil Drawing, Multi-parameter, Pencil Texture, Motion Blur, Color Space Conversion

Received: March 22, 2016 / Accepted: April 1, 2016 / Published online: September 3, 2016

© 2016 The Authors. Published by American Institute of Science. This Open Access article is under the CC BY license.

<http://creativecommons.org/licenses/by/4.0/>

1. Introduction

Early computer graphics are mainly studying how to really simulate of the real world, and in recent years, research on how to create an art style works with a picture received lots of attention. These Non-photorealistic computer graphic methods are widely used in image synthesis, effects and other commercial image application.

Pencil drawing in Non-photorealistic computer graphics simulation has been given more attention and developments. The principal method to render a pencil drawing is divided into two categories: manually (interactively) and automatically [1].

Many researchers have proposed different texture rendering method to generate a pencil picture automatically [2-6]. There are also some researches on fast generating algorithm by using hardware [7-8].

With real pencils, the artist's paintings will consider the

contrast, brightness, line length, angle of a line, and such other factors. Drawing with these arguments and with a blur filter, it presented a method in this paper, which is based on the gray segmentation to generate the pencil texture and through the fusion of three kinds of color space to get a colored pencil drawing. Compared with other algorithms for generating pencil drawings, the proposed method avoided complexity and difficulty to implement image segmentation algorithm based on linear convolution, and by using blur filter to produce a pencil drawing textures at the same time, taking into account the 6 drawing parameters, generated the painting texture, which is very similar to hand-drawn pencil sketches. In addition, colored pencil drawings are generated by using multiple color space conversion fusion. It can ensure that the processed image will not have a high color distortion.

* Corresponding author

E-mail address: shelly200607@126.com

2. Related Works

Since Vermeulen [9] developed a pencil Sketch system in 1989, which needed numerous drawing parameters in order to produce an image manually. More researchers focus on automatic algorithms. Sang Sang [10] conceived a method based on the texture and contour. Sun Shuo [11] designed a method using region generating. Ren Xiao Kang [12] proposed a method according to luminance layering values based on wavelet hierarchy, and applied motion blur to generate different direction of the texture. Nan Li [13] presented a method for generating a digital image using the feature geometric pencil drawing attributes obtained by analysis of image moment and texture of each region. Zhang RenZhi [14] used line integral convolution to create the image texture. Lu Cong Lin combined a LAB space, color difference image and a texture image as features. In 3d algorithm, Mohr [16] mimicked the multiple contours in a pencil drawing. Loviscach [17] put forward a multiple contour method based on graphics hardware support.

3. The Framework for Generating the Colored Pencil Drawings

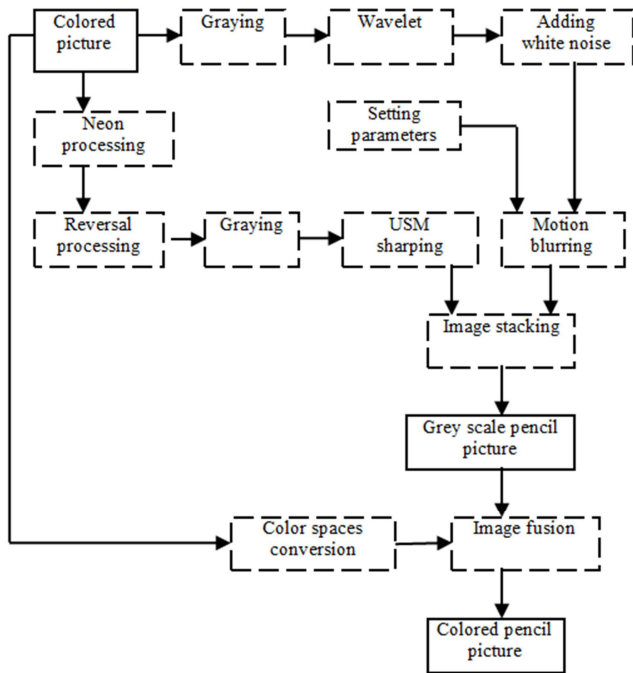


Fig. 1. Framework of the algorithm.

Pencil drawing processing in this paper can be summarized into the following steps:

STEP 1 to set the pencil drawing parameters.

STEP 2 to obtain the contour by Neon, reversed phase, gray

and USM sharpening.

STEP 3 to get the pencil texture after the continuous process of graying, adding white noise, wavelet transforming and motion blurring.

STEP 4 to overlay the contour and texture for obtaining a black and white pencil drawing.

STEP 5 to utilize different color space conversion and generate the final colored pencil drawing with image fusion of the multiple color space conversion results.

Figure 1 shows that the detailed framework of the proposed algorithm for getting a colored pencil drawing.

4. Key Algorithms

4.1. Obtaining the Contour

Contour is essential for generating color pencil picture. We firstly used neon approach to process the original image, and after the phase and the subsequent. It can get the contour of the image. See neon processing in Equation (1). Where R_1 denotes the red component of the pixel in the original picture, and R_2 represents t of which in the same row, and R_3 is the average value of the red component of the same column, and R is the final result. It's same to the components of G , B .

$$\begin{aligned} R &= 2 * \sqrt{(R_1 - R_2)^2 - (R_1 - R_3)^2} \\ G &= 2 * \sqrt{(G_1 - G_2)^2 - (G_1 - G_3)^2} \\ B &= 2 * \sqrt{(B_1 - B_2)^2 - (B_1 - B_3)^2} \end{aligned} \quad (1)$$

In order to enhance the contrast of the edges, we also use the unsharp mask sharpening method to deal with the contour information. Unsharp Mask sharpening is realized by a smooth filter, which subtracts with unprecedented image smoothing results. This method can increase the contrast of image edges. See specific expression in Equation (2)

$$I_{output} = I_{input} + \lambda * I_{filter} \quad (2)$$

Where I_{input} represents the pixel value of the input image, and I_{output} denotes the pixel value of the output image, besides, I_{filter} is the correcting signal, which is obtained by High-pass filter for x , and λ is a scaling factor, which is used to control the enhancement effect.

4.2. Wavelet Layered

Based on the wavelet layered thought, firstly, it uses wavelet basis function to process gray images, and achieves four channel information (a low frequency and three high frequency), secondly, it reconstructs the information to obtain data of 4 layers.

4.3. Adding White Noise

After the wavelet layered, due to there is little high frequency information in the low frequency image, thus we add some white noise to affect the final shading of pencil picture. Suppose that I_{input} is the grey value of the original image and P is a random number, we use the threshold gray level to control threshold [13], the specific algorithm is shown in equation (3).

$$I_{noise} = \begin{cases} I_{noise1} = \begin{cases} I_{min1}, P > T_1 \\ I_{max}, otherwise \end{cases} & I_{input} \leq L_1 \\ I_{noise2} = \begin{cases} I_{min2}, P > T_2 \\ I_{max}, otherwise \end{cases} & L_1 < I_{input} \leq L_2 \\ I_{noise3} = \begin{cases} I_{min3}, P > T_3 \\ I_{max}, otherwise \end{cases} & I_{input} \leq L_2 \end{cases} \quad (3)$$

In the equation(3), $T_1 = k_1 \times (I_{input} / 255)$, $T_2 = k_2 \times (I_{input} / 255)$, $T_3 = k_3 \times (I_{input} / 255)$, and k_1 , k_2 and k_3 were set by 0.9, 0.7, 0.7. These values can also be customized, but the value is needed among [0, 1].

According to the above method, the contrast result of adding a white noise picture and the original image is presented in figure 2.

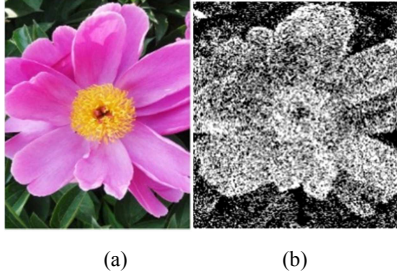


Fig. 2. The original image and the effect after adding white noise.

- (a) Original picture
- (b) Result of adding white noise

4.4. Setting Parameters

Before generating a pencil texture, we need to set these parameters: contrast, brightness, length of lines, angles, strength and width of lines. Given the value of contrast, it can change the image contrast. Brightness is used according to multiply the original pixel value of the setting parameter. It can enhance the brightness of the pencil pictures. The unit of length and Angle is the unit of pixel; they are the parameters, which are used in motion blurring. The length of the lines affects the density of lines in pencil texture, angle can vary the direction of lines; Strength and width are used to manipulate the gray scale information of the picture, thus, the greater value of strength is set, the deeper color is.

4.5. Motion Blur

Common measures for generating pencil line include linear convolution and the motion blur filter method. The difficulty of the former approach requires to calculate the linear convolution, which is complex and with higher time-consuming. However, effective motion blur is a filter to blur by setting the direction and intensity of the image, and the key parameters are the angle and intensity, in addition, the algorithm is easy to implement. This article uses the latter method to generate pencil lines.

By solving gradient of the pixel to get the blur angle, while fuzzy strength will be set according to the value experience. For large size images, solving the gradient is obviously time-consuming, so this article will gray scale the images, and use gray scale values to get blur angle, as shown in Equation (4). According to the test, 180 degree will be divided into 12 intervals, and it is advisable to make 15 degrees for each interval. The current pixel point within the range of 0 to 15 degrees will use the fuzzy filters with 7 degree angles, and within the range of 15 to 30 degrees to use fuzzy filters with 22 degree angle, and so on. As one of wavelet layered result, low-frequency image generation will be processed to generate 12 different direction graphs. The final pencil texture is the result of overlaying all the 12 graphs.

$$\alpha = \frac{I_{input}}{I_{max} - I_{min}} \times 180 \quad (4)$$

I_{max} is the max gray value of the input image, and I_{min} is the minimum value, the gray value of a pixel is I_{input} .

4.6. Overlap

We overlay the results after motion blurring with contour to get the grayscale pencil image. We used the equation (5) to achieve black and white picture. Set the contour of the original image in the M_1 , the final black and white pencil drawing as a result. The value of r ranged from 0 to 1.

$$final(i, j) = r * M_1(i, j) + (1 - r) * I_2(i, j) \quad (5)$$

4.7. Fusion of Multiple Color Space Image Conversion

There are three parameters in the HSV color model (H for Hue, S for saturation, V for value of brightness), and V is not related to the color of pixels. Brightness parameter Y in YUV model is also independent with other two parameters (U, V). When using a certain color space conversion to RGB color model, it will cause some color distortion, so in this paper, we used a fusion of multiple color space conversion, which can be obtained a lower distortion. The fusion formula is shown below, in which $a + b + c = 1$.

$$I_{output} = a * I_{HSV} + b * I_{YUV} + c * I_{Lab} \quad (6)$$

Based on many experiments, the general experience value is: $a=0.7$, $b=0.15$, $c=0.15$.

5. Experimental Result

With platforms of VC6.0 and Matlab, we realized the algorithm in a Windows 7 OS environment. The hardware configuration is: CPU is Intel dual-core 2.30 Hz and the RAM is 6 GB.

In this paper, we chose three different pictures to generate colored pencil drawings, and the test results are shown in table 1.

Table 1. Time consuming of different picture processing with color space conversion.

	Picture1	Picture2	Picture3
Picture specification	99*105	316*426	566*800
Picture size	30K	394K	1.29M
Time consuming (Hsv)	0.141s	1.872 s	9.812s
Time consuming (Yuv)	0.156 s	2.069 s	9.843s
Time consuming (Lab)	0.250 s	2.824 s	2.824 s
Time consuming (multi-color)	0.286 s	3.463 s	14.504s

Figure 3 displays the result of automatic rendering colored pencil drawing. As shown in figure 3 (c), with fusion of multiple color space conversion, the result has lower color distortion correction, contrast with single color space conversion result in Figure 3 (b). The result in Figure 3 (c) is processed by setting parameters as follows: contrast is 18, brightness is 16, length of the line is 28, angle is 143, strength is 23 and width is 0.1.



Fig. 3. Result of the proposed algorithm.

- (a)Original picture
 (b)Rendering with single color space
 (c)Rendering with multi-color space

With the pencil drawing parameters, the colored pencil drawing is generated. Table 2 shows the value ranges of the parameters and empirical values.

Table 2. The value ranges of the parameters and empirical values.

	Value range	Empiricalvalue
Contrast	0-49	12-20
Brightness	0-50	12-16
Length of lines	0-50	20-40
Angle	-360-360	-170-170
Strength	0-60	20-40
Width of lines	0-1	0.1

6. Conclusion

The proposed algorithm in this paper takes into consideration the parameters when rendering the pencil picture, and utilizes angle matrix to form pencil lines, the resulting pencil lines are similar to hand-drawn. The method is efficient for its least distortion in color, meanwhile, there is little difference between the time consuming of the multiple color space conversion and Lab to RGB conversion. In order to avoid the line intersection operation, setting parameter is not according to the pixel width, but with the fixed value, so the thickness of line setting method will be researched in future work.

Acknowledgment

The author gratefully acknowledges the support of this research by Xi'an University of Science and Technology Youth Team Foundation Project.

References

- [1] Shi Xiao Nan. The Research Summary of pencil rendering Science & Technology Information, 2010. 35(12): 94-95.
- [2] Paiva, A., Brazil, E., Petronetto, F., Sousa, M.: Fluid-based hatching for tone mapping in line illustrations. The Visual Computer 25, 2009: 519-527.
- [3] Sun DanDan, Tang Di. Method for simulation of pencil drawing based on motion blur. Computer Engineering and Design, 2009, 30(24): 689-5691.
- [4] Li Long Sheng, Zhou Jing Ye.. New generation method of pencil drawing using USM sharpening. Journal of Jilin University (Engineering and Technology Edition), 2007, 37(2): 442-447.
- [5] XieDang'en, Zhang Zhili, Xu Dan. AN IMPROVED 2D AUTO-RENDERING ALGORITHM FOR COLOURED PENCIL DRAWING. Computer Applications and Software. 2013.30(8): 8-32.
- [6] Yan Li Wei. Research and Implementation based on Gray with color pencil drawing [D]. Northwest Normal University, 2012.
- [7] Xie Dang. En, Zhao Yang, Xu Dan. A Method for Generation of Pencil Filter and its Implementation on GPU [J]. Journal of Computer-Aided Design &Computer raphics, 2008, 20(1): 26-30.
- [8] Zhang Jiang Hong, Zhao Yang. Architecture Design and Optimization of Pencil Sketching Algorithm Based on FPGA. Journal of Engineering Graphics, 2006.4(4): 78-91.
- [9] Vermeulen A H, Tanner P. Pencil sketch a pencil-based paint system [C]. Graphics Interface'89 Conference Proceedings, London, 1989: 138-143.
- [10] Sang Sang, Hao Peng Yi, Ding You Dong, et al. Pencil Drawing Generation Based on Texture and Profile [J]. Journal of Shanghai University (Natural Science Edition), 2010, 16(3): 312-317.

- [11] SunShuo, Huang Dong - wei. Efficient region - based pencil drawing. *Computer Engineering and Applications*, 2007, 43(14): 34-37.
- [12] Ren Xiao Kang, YAN Li Wei, GAO Hongyu. Method for simulation of pencil drawing texture based on wavelet layers. *Computer Engineering and Applications*, 2012, 48(11): 210-213.
- [13] Nan Li, Zhong Huang. A Feature-Based Pencil Drawing Method [C]. *Computer graphics and interactive technique, Australasia and South East Asia*, 2003: 135-143.
- [14] Zhang RenZhi, Zhou Yuan. Research and Painting of Pencil Sketching Based on Convolution Threshold [J]. *Laser & Optoelectronics Progress*, 2015, 52: 1-6.
- [15] Liu Conglin, Shang Yuan Yuan. Pencil drawing method based on arctangent hue differentiation model [J]. *Journal of image and graphics*, 2015, 20: 1253-1261.
- [16] Mohr, A. Non-invasive interactive stylized rendering. In *Proc. 2001 ACM Symposium on Interactive 3D Graphics*. 2001: 175-178.
- [17] Loviscach. J. Stylized haloed outlines on the GPU. *ACM Computer Graphics*. 2004.