GENERATION OF MARBLING IMAGES USING CORRELATION COEFFICIENTS BETWEEN RGB OF PHOTOGRAPHIC IMAGES

Toru Hiraoka

Department of Information Systems
University of Nagasaki
1-1-1, Manabino, Nagayo-chou, Nishisonogi-gun, Nagasaki-ken 851-2195, Japan
hiraoka@sun.ac.jp

ABSTRACT. *We propose a non-photorealistic rendering method to generate marbling images from photographic images. Marbling is a drawing technique that paint or ink is floated in a heavy liquid and the floating patterns are dyed on paper. Marbling images combine the floating patterns with photographic images. Our method is implemented by an iterative processing using correlation coefficients between RGB of photographic images. By using our method, it is possible to automatically generate marbling patterns on photographic images. To verify the effectiveness of our method, we conducted an experiment with changing the values of parameters of our method using Lenna image and an experiment using various photographic images. Potential applications of our method range from the Internet, television and magazines.*
Keywords: Non-photorealistic rendering, Marbling, Correlation coefficient, RGB, Automatic generation

1. **Introduction.** Since the 1990s, non-photorealistic rendering (NPR) techniques for realizing pictorial expressions such as oil paintings and watercolors and expressions of creative art such as illustrations and cartoons by computer graphics [1] have been studied [2, 3, 4, 5, 6, 7, 8]. NPR techniques can reduce the cost of creating works of art manually. All of the necessary materials and tools for NPR techniques are implemented as software. With NPR techniques, processing such as color tone conversion and blurring can be easily performed, and it is possible to immediately cancel processing with a simple operation even if processing fails. For these reasons, NPR techniques are currently used in a variety of fields such as the Internet, television and magazines by professional and amateur users.

A marble pattern is a printed pattern that imitates marble and expresses flowing shapes in multiple colors. A drawing technique that patterns made by floating paint or ink in a heavy liquid are dyed on paper is called "marbling". The patterns created by marbling are similar to marble, and are colorful, streamlined and complicated.

In this paper, we develop an NPR method to automatically generate marbling images from photographic images. Our marbling images are represented by colorful, streamlined and complex marble patterns. Our method is implemented by an iterative processing using correlation coefficients between RGB of photographic images. To verify the effectiveness of our method, the changes of marbling images generated by changing the parameters of our method were visually confirmed using Lenna image. An experiment using various photographic images was also conducted.

A number of methods to generate marbling images have been proposed [9, 10, 11, 12, 13]. Jin et al. [9] generated marbling designs as the result of color advection in two-dimensional flow fields obtained by numerically solving Navier-Stokes equations on GPU with a multi-grid solver. Ando and Tsuruno [10] generated marbled textures that can be exported into

a vector graphics format based on an explicit surface tracking method. Lu et al. [11, 12] generated marbling animations using a mathematical approach with closed-form expressions, and marbling designs of three-dimensional solids that supports a compact random-access vector representation. Gazziro et al. [13] developed an interactive and intuitive application that simulates the marbling process digitally in real time. Although these conventional methods cannot generate marbling patterns from photographic images, the proposed method can generate marbling patterns from photographic images. Since it has become possible to easily take pictures with the widespread use of cameras embedded in mobile terminals and can be easily processed by the application installed in mobile terminals, many users such as professional and amateur users can be assumed by generating marbling images from photographic images as in our method.

There are some studies [14, 15, 16, 17] on NPR using correlation coefficients. NPR images generated using correlation coefficients were parallel-fine-curve-line images [14], concrete-wall-like images [15], ribbed-pattern-like images [16] and crepe-paper-like images [17]. Marbling images of our method and the conventional NPR images [14, 15, 16, 17] have different expressions.

The rest of this paper is organized as follows. Section 2 describes our method to generate marbling images from photographic images. Section 3 shows experimental results, and reveals the effectiveness of our method. Finally, Section 4 concludes this paper.

2. **Our Method.** Our method generates marbling images from photographic images. Our method is executed in three steps. The first step smooths photographic images. The second step calculates correlation coefficients between RGB of the smoothed photographic images. The third step transforms photographic images using the correlation coefficients. A flow chart of our method is shown in Figure 1.
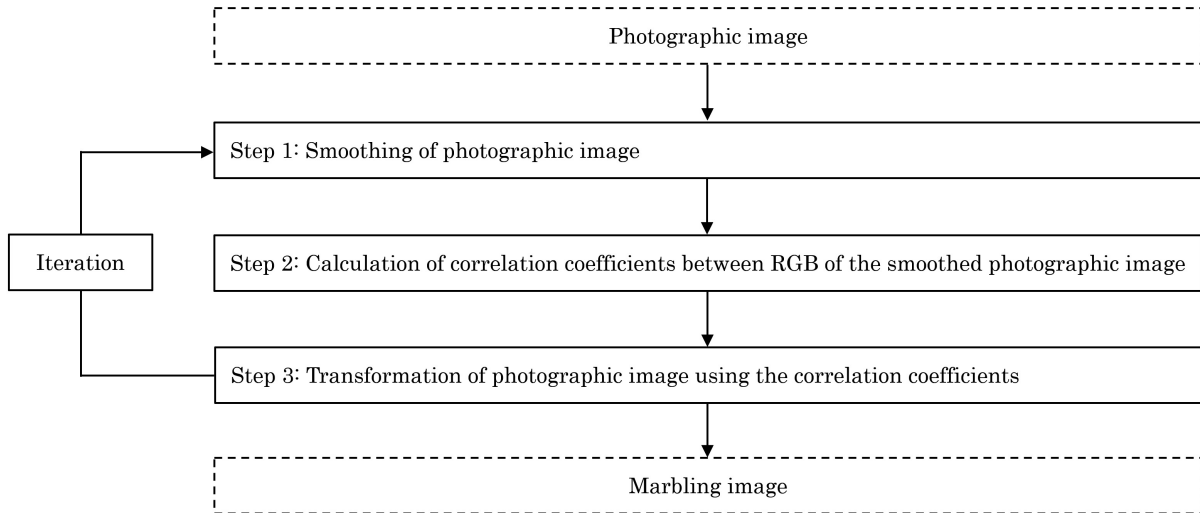

FIGURE 1. Flow chart of our method

Details of the procedure in Figure 1 are explained below.

Step 0: The input pixel values of RGB for spatial coordinates (i, j) ($i = 1, 2, \ldots, I$; $j = 1, 2, \ldots, J$) of a photographic image are defined as $f_{R,i,j}$, $f_{G,i,j}$ and $f_{B,i,j}$. The pixel values of the image at the t-th iteration number are defined as $f_{R,i,j}^{(t)}$, $f_{G,i,j}^{(t)}$ and $f_{B,i,j}^{(t)}$, where $f_{R,i,j}^{(1)} = f_{R,i,j}$, $f_{G,i,j}^{(1)} = f_{G,i,j}$ and $f_{B,i,j}^{(1)} = f_{B,i,j}$. The pixel values $f_{R,i,j}^{(t)}$, $f_{G,i,j}^{(t)}$ and $f_{B,i,j}^{(t)}$ have value of U gradation from 0 to $U - 1$.

Step 1: The pixel values $s_{R,i,j}^{(t)}$, $s_{G,i,j}^{(t)}$ and $s_{B,i,j}^{(t)}$ are computed using smoothing filter from $f_{R,i,j}^{(t)}$, $f_{G,i,j}^{(t)}$ and $f_{B,i,j}^{(t)}$ as the following equations, respectively.

$$s_{R,i,j}^{(t)} = \frac{\sum_{k=-W}^{W} \sum_{l=-W}^{W} f_{R,i+k,j+l}^{(t)}}{(2W+1)^2} \tag{1}$$

$$s_{G,i,j}^{(t)} = \frac{\sum_{k=-W}^{W} \sum_{l=-W}^{W} f_{G,i+k,j+l}^{(t)}}{(2W+1)^2} \tag{2}$$

$$s_{B,i,j}^{(t)} = \frac{\sum_{k=-W}^{W} \sum_{l=-W}^{W} f_{B,i+k,j+l}^{(t)}}{(2W+1)^2} \tag{3}$$

where W is the window size, and k and l are the positions in the window.

Step 2: The correlation coefficients $c_{RG,i,j}^{(t)}$, $c_{GB,i,j}^{(t)}$ and $c_{BR,i,j}^{(t)}$ between RGB of the smoothed images are calculated as the following equations, respectively.

$$c_{RG,i,j}^{(t)} = \frac{\sum_{k=-W}^{W} \sum_{l=-W}^{W} \left(s_{R,i,j}^{(t)} - a_R^{(t)} \right) \left(s_{G,i,j}^{(t)} - a_G^{(t)} \right)}{\sqrt{\sum_{k=-W}^{W} \sum_{l=-W}^{W} \left(s_{R,i,j}^{(t)} - a_R^{(t)} \right)^2} \sqrt{\sum_{k=-W}^{W} \sum_{l=-W}^{W} \left(s_{G,i,j}^{(t)} - a_G^{(t)} \right)^2}} \tag{4}$$

$$c_{GB,i,j}^{(t)} = \frac{\sum_{k=-W}^{W} \sum_{l=-W}^{W} \left(s_{G,i,j}^{(t)} - a_G^{(t)} \right) \left(s_{B,i,j}^{(t)} - a_B^{(t)} \right)}{\sqrt{\sum_{k=-W}^{W} \sum_{l=-W}^{W} \left(s_{G,i,j}^{(t)} - a_G^{(t)} \right)^2} \sqrt{\sum_{k=-W}^{W} \sum_{l=-W}^{W} \left(s_{B,i,j}^{(t)} - a_B^{(t)} \right)^2}} \tag{5}$$

$$c_{BR,i,j}^{(t)} = \frac{\sum_{k=-W}^{W} \sum_{l=-W}^{W} \left(s_{B,i,j}^{(t)} - a_B^{(t)} \right) \left(s_{R,i,j}^{(t)} - a_R^{(t)} \right)}{\sqrt{\sum_{k=-W}^{W} \sum_{l=-W}^{W} \left(s_{B,i,j}^{(t)} - a_B^{(t)} \right)^2} \sqrt{\sum_{k=-W}^{W} \sum_{l=-W}^{W} \left(s_{R,i,j}^{(t)} - a_R^{(t)} \right)^2}} \tag{6}$$

where

$$a_R^{(t)} = \frac{\sum_{i=1}^{I} \sum_{j=1}^{J} s_{R,i,j}^{(t)}}{IJ} \tag{7}$$

$$a_G^{(t)} = \frac{\sum_{i=1}^{I} \sum_{j=1}^{J} s_{G,i,j}^{(t)}}{IJ} \tag{8}$$

$$a_B^{(t)} = \frac{\sum_{i=1}^{I} \sum_{j=1}^{J} s_{B,i,j}^{(t)}}{IJ} \tag{9}$$

Step 3: The pixel values $f_{R,i,j}^{(t+1)}$, $f_{G,i,j}^{(t+1)}$ and $f_{B,i,j}^{(t+1)}$ are updated using the correlation coefficients as the following equations, respectively.

$$f_{R,i,j}^{(t+1)} = \begin{cases} f_{R,i,j} - \alpha c_{RG,i,j}^{(t)} \left(s_{R,i,j}^{(t)} - s_{G,i,j}^{(t)} \right) & (t\%2 = 1) \\ f_{R,i,j} + \alpha c_{RG,i,j}^{(t)} \left(s_{R,i,j}^{(t)} - s_{G,i,j}^{(t)} \right) & (t\%2 = 0) \end{cases} \tag{10}$$

$$f_{G,i,j}^{(t+1)} = \begin{cases} f_{G,i,j} - \alpha c_{GB,i,j}^{(t)} \left(s_{G,i,j}^{(t)} - s_{B,i,j}^{(t)} \right) & (t\%2 = 1) \\ f_{G,i,j} + \alpha c_{GB,i,j}^{(t)} \left(s_{G,i,j}^{(t)} - s_{B,i,j}^{(t)} \right) & (t\%2 = 0) \end{cases} \tag{11}$$

$$f_{B,i,j}^{(t+1)} = \begin{cases} f_{B,i,j} - \alpha c_{BR,i,j}^{(t)} \left(s_{B,i,j}^{(t)} - s_{R,i,j}^{(t)} \right) & (t\%2 = 1) \\ f_{B,i,j} + \alpha c_{BR,i,j}^{(t)} \left(s_{B,i,j}^{(t)} - s_{R,i,j}^{(t)} \right) & (t\%2 = 0) \end{cases} \tag{12}$$

where α is a positive constant, and $\%$ represents a remainder operation. In case $f_{R,i,j}^{(t+1)}$, $f_{G,i,j}^{(t+1)}$ are less than 0, then $f_{R,i,j}^{(t+1)}$, $f_{G,i,j}^{(t+1)}$ must be set to 0, respectively. In

case $f_{R,i,j}^{(t+1)}$, $f_{G,i,j}^{(t+1)}$ are greater than $U - 1$, then $f_{R,i,j}^{(t+1)}$, $f_{G,i,j}^{(t+1)}$ must be set to $U - 1$, respectively. A marbling image is obtained after the processing of Step 1 to Step 3 of T times iteration, where T is an even number.

3. **Experiments.** We mainly conducted two experiments. The first experiment with changing the values of the parameters in our method was conducted using Lenna image shown in Figure 2. The second experiment was conducted to verify visually that marbling images can be generated using four photographic images shown in Figure 3. All photographic images used in the experiments were $512 * 512$ pixels and 256 gradation. Since marbling patterns could be generated entirely in all photographic images, the value of the iteration number T was set to 10 in the experiments.

FIGURE 2. Lenna image
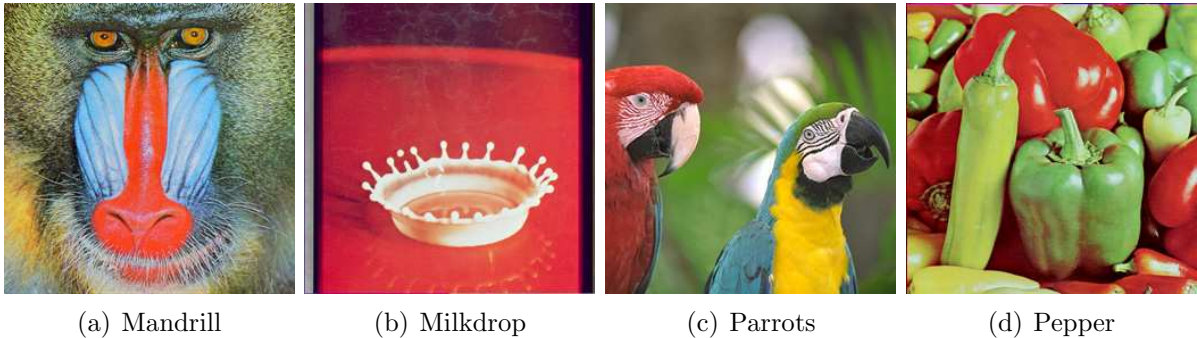
(a) Mandrill   (b) Milkdrop   (c) Parrots   (d) Pepper

FIGURE 3. Various photographic images

We visually confirmed marbling images changed the value of the window size W using Lenna image. We set the value of W to 1, 2, 3 and 4, and set the value of α to 3.0. The results of the experiment with changing the value of W are shown in Figure 4. As the value of W was larger, marbling patterns became wider and it became difficult to visually recognize Lenna image. The value of W may be set according to the application of the user.

We visually confirmed marbling images changed the value of the parameter α using Lenna image. We set the value of α to 1.0, 2.0, 3.0 and 4.0, and set the value of W to 2. The results of the experiment with changing the value of α are shown in Figure 5. As the value of α was larger, marbling patterns became clear and it became difficult to visually recognize Lenna image. The value of α may be set according to the application of the user.

We visually confirmed marbling images using four photographic images. Since marbling patterns and Lenna image were visually recognized well in the previous experiment, we set the values of the parameters W and α to 2 and 3.0, respectively. The results of the experiment using four photographic images are shown in Figure 6. All marbling images
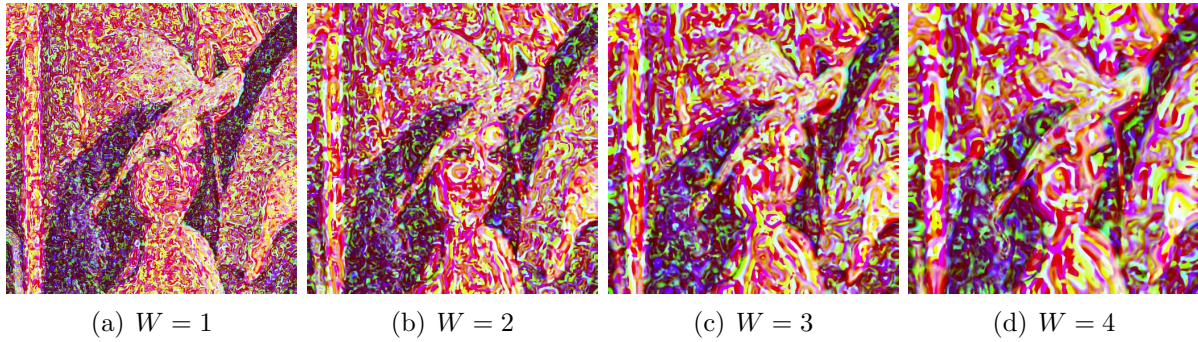
(a) $W = 1$ (b) $W = 2$ (c) $W = 3$ (d) $W = 4$

FIGURE 4. Marbling images generated by changing the value of the window size W

(a) $\alpha = 1.0$ (b) $\alpha = 2.0$ (c) $\alpha = 3.0$ (d) $\alpha = 4.0$
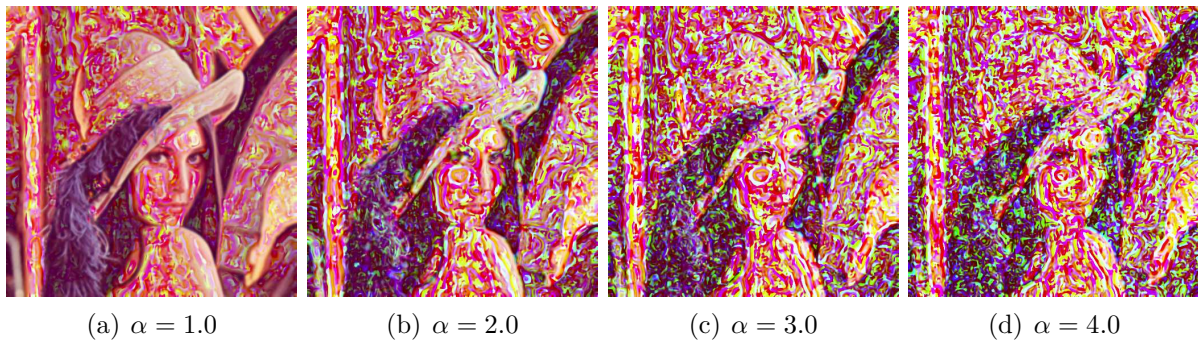
FIGURE 5. Marbling images generated by changing the value of the parameter α

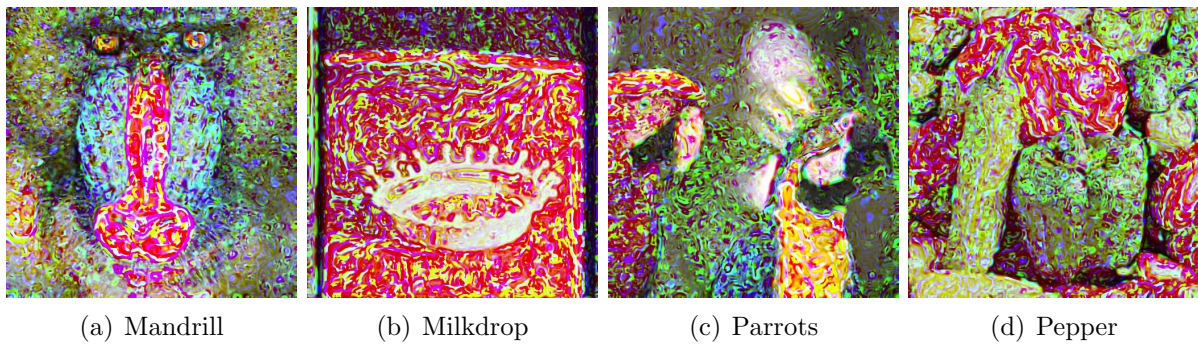(a) Mandrill   (b) Milkdrop   (c) Parrots   (d) Pepper

FIGURE 6. Marbling images

were represented by colorful, streamlined and complex marble patterns. The conventional methods [9, 10, 11, 12, 13] could not generate marbling images from photographic images, but our method enables the generation of marbling images from photographic images. And, comparing NPR images of the conventional methods [14, 15, 16, 17] generated using correlation coefficient with marbling images of our method, the expressions were clearly different. However, marbling patterns were hard to occur in the fine texture area of the hairs in Figure 6(a), in the black area on the right side of Figure 6(b) and in the white area in Figure 6(c).

4. **Conclusions.** We developed an NPR method to automatically generate marbling images from photographic images. Our method was executed by an iterative processing using correlation coefficients between RGB of photographic images. The effectiveness of

our method was visually verified through experiments using various images. The experimental results showed that our method can generate marbling images that have colorful, streamlined and complex marble patterns.

A subject for future study is to generate marbling patterns in the fine texture, black and white areas. Another future study is to expand our method for application to videos and three-dimensional models.

REFERENCES

[1] L. Xue, X. Yi, Y. C. Lin and J. W. Drukker, An approach of the product form design based on GRA-fuzzy logic model: A case study of train seats, *International Journal of Innovative Computing, Information and Control*, vol.15, no.1, pp.261-274, 2019.

[2] P. Haeberli, Paint by numbers: Abstract image representations, *ACM SIGGRAPH Computer Graphics*, vol.24, no.4, pp.207-214, 1990.

[3] J. Lansdown and S. Schofield, Expressive rendering: A review of nonphotorealistic techniques, *IEEE Computer Graphics and Applications*, vol.15, no.3, pp.29-37, 1995.

[4] J. Daniel, S. Erik, Y. Anders and R. Timo, A survey of volumetric illumination techniques for interactive volume rendering, *Computer Graphics Forum*, vol.33, no.1, pp.27-51, 2014.

[5] B. Dalstein, R. Ronfard and M. V. D. Panne, Vector graphics animation with time-varying topology, *ACM Trans. Graphics*, vol.34, no.4, 2015.

[6] D. Martin, G. Arroyo, A. Rodriguez and T. Isenberg, A survey of digital stippling, *Computers & Graphics*, vol.67, pp.24-44, 2017.

[7] K. Lawonn, I. Viola, B. Preim and T. Isenberg, A survey of surface-based illustrative rendering for visualization, *Computer Graphics Forum*, vol.37, no.6, pp.205-234, 2018.

[8] T. Hiraoka, Generation of pop art-like images using binomial distribution, *ICIC Express Letters*, vol.14, no.3, pp.227-233, 2020.

[9] X. Jin, S. Chen and X. Mao, Computer-generated marbling textures: A GPU-based design system, *IEEE Computer Graphics and Applications*, vol.27, no.2, pp.78-84, 2007.

[10] R. Ando and R. Tsuruno, Vector graphics depicting marbling flow, *Computers & Graphics*, vol.35, no.1, pp.148-159, 2010.

[11] S. Lu, A. Jaffer, X. Jin, H. Zhao and X. Mao, Mathematical marbling, *IEEE Computer Graphics and Applications*, vol.32, no.6, pp.26-35, 2012.

[12] S. Lu, X. Jin, A. Jaffer, F. Gao and X. Mao, Solid mathematical marbling, *IEEE Computer Graphics and Applications*, vol.37, no.2, pp.90-98, 2017.

[13] M. Gazziro, J. P. Gois, C. Gonzales and J. F. Rodrigues, A computational method for interactive design of marbling patterns, *The 17th Brazilian Symposium on Computer Games and Digital Entertainment*, pp.352-361, 2018.

[14] T. Hiraoka, T. Katayama and K. Urahama, Generation of parallel-fine-curve-line images by iterative calculation using correlation coefficient, *ICIC Express Letters*, vol.12, no.11, pp.1131-1136, 2018.

[15] T. Hiraoka, T. Katayama and K. Urahama, Generation of concrete-wall-like images by autocorrelation coefficient and inverse filtering, *ICIC Express Letters*, vol.13, no.2, pp.127-132, 2019.

[16] T. Hiraoka, Generation of arbitrarily-oriented ribbed-pattern images using autocorrelation coefficient, *Journal of the Institute of Industrial Applications Engineers*, vol.7, no.3, pp.79-82, 2019.

[17] T. Hiraoka, Generation of edge-preserving crepe-paper-like images using smoothing filter and correlation coefficient, *ICIC Express Letters*, vol.14, no.5, pp.443-448, 2020.