ICIC Express Letters
Volume **13**, Number **6**, June **2019**

ICIC International ©2019 ISSN 1881-803X
pp. 469–474

GENERATION OF BUBBLE IMAGES USING ADDITIVE AND MULTIPLICATIVE AVERAGES IN DIFFERENT WINDOW SIZES

Toru Hiraoka[1] and Kiichi Urahama[2]

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

[2]Department of Communication Design Science
Kyushu University
4-9-1, Shiobaru, Minami-ku, Fukuoka-shi, Fukuoka-ken 815-8540, Japan
urahama@design.kyushu-u.ac.jp

Received December 2018; accepted February 2019

Abstract. *This paper proposes a novel non-photorealistic rendering method for generating bubble images from gray-scale photographic images. Bubble images are non-photorealistic images embedded in many bubbles. These bubble patterns are automatically generated according to the density and contour of photographic images. The proposed method is executed by an iterative calculation using additive and multiplicative averages in different window sizes. As the number of the iterative calculation increases, bubble patterns gradually appear. In order to verify the effectiveness of the proposed method, experiments using Lenna image and other photographic images are conducted. In the experiments, it is clarified how to change bubble patterns generated by varying the values of the parameters in the proposed method.*

Keywords: Bubble image, Additive average, Multiplicative average, Different window sizes

1. **Introduction.** From the 1990s, researches on non-photorealistic rendering (NPR) have been actively conducted in academia and industry [1, 2, 3, 4, 5, 6, 7, 8]. NPR is a computer graphics technique for generating non-photorealistic images such as oil paintings and illustration drawings. Also, many NPRs using geometric patterns such as lines and curves have been proposed [9, 10, 11, 12, 13]. Such NPR is utilized in the fields of design, amusement, and entertainment. In recent years, many researches on NPR that synthesized patterns seen in nature are conducted [14, 15, 16]. Many things in nature are composed of visually distinct elements arranged in various distributions. For example, reaction-diffusion images [14], cell images [15], and interference-ripple images [16] have been proposed as such NPR in nature.

Focusing on NPR in nature, this paper proposes a novel method for generating bubble images from gray-scale photographic images. Bubble images are non-photorealistic images embedded in many bubbles. Bubble patterns consist of visually distinct separate bubbles arranged in various distributions. NPR that generates bubble images cannot be found in previous researches. If we give an image similar to bubble images, there are interference-ripple images. The conventional method generates continuous ripple patterns, and the proposed method generates partially divided bubble patterns. Therefore, these images give different impressions. In addition, the conventional method has regions where it is

DOI: 10.24507/icicel.13.06.469

difficult to generate interference-ripple patterns, but the proposed method can generate bubble patterns on the whole image.

The proposed method is executed by an iterative calculation using additive and multiplicative averages in different window sizes, and then is simple to process. Therefore, the proposed method is easy to implement in applications. By using the proposed method, bubble patterns are automatically generated according to the density and contour of photographic images, and the size and density of bubble patterns can be changed by varying the values of the parameters. In order to verify the effectiveness of the proposed method, experiment using Lenna image is conducted by varying the values of the parameters in the proposed method. The result of the experiment is an index when applying the proposed method to various photographic images of different sizes and contents. In addition, experiment using various photographic images other than Lenna image is done. As a result of the experiments, it is revealed that the proposed method can automatically generate bubble patterns according to the density and contour of photographic images.

The rest of this paper is organized as follows. Section 2 describes the proposed method for generating bubble images. Section 3 shows experimental results, and reveals the effectiveness of the proposed method. Finally, Section 4 concludes this paper.

2. **Proposed Method.** The proposed method generates bubble images by the iterative calculation using additive and multiplicative averages in different window sizes. Photographic images are converted using the values calculated from the additive and multiplicative averages. The flow chart of the proposed method is shown in Figure 1. The detailed procedure of the proposed method is shown as follows.

Step 0: The input pixel values for spatial coordinates (i, j) of a gray-scale photographic image are defined as $f_{i,j}$. Then, the pixel values of the image at the t-th iteration number are defined as $f_{i,j}^{(t)}$, where $f_{i,j}^{(1)} = f_{i,j}$. The pixel values $f_{i,j}^{(t)}$ have value of 256 gradation from 0 to 255.

Step 1: The additive averages $a_{1,i,j}^{(t)}$ are calculated using pixels within window size W around spatial coordinates (i, j) as the following equation.

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

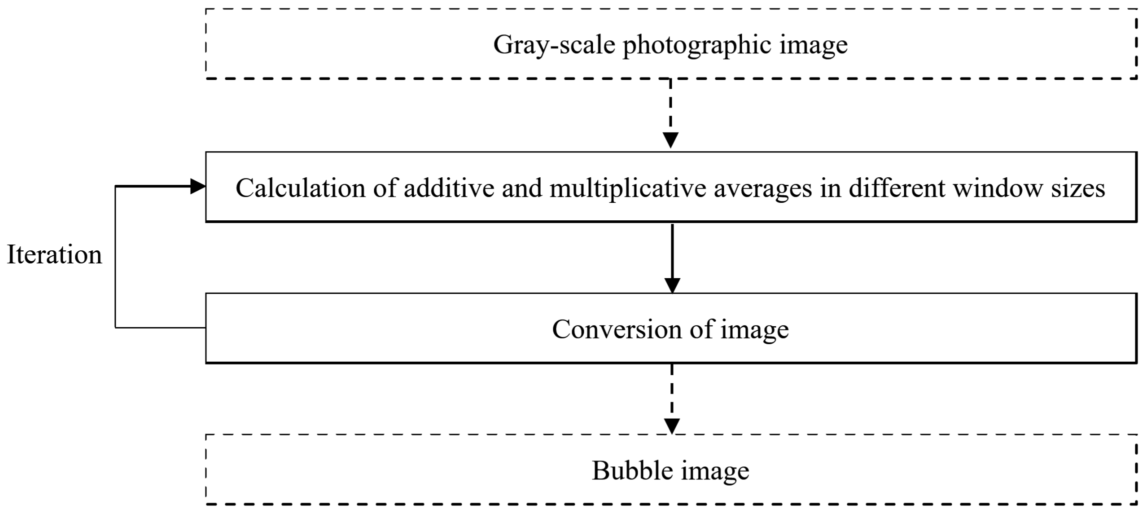


FIGURE 1. Flow chart of the proposed method

Similarly, the additive averages $a^{(t)}_{2,i,j}$ are calculated by the window size $W+1$ as the following equation.

$$a^{(t)}_{2,i,j} = \frac{\sum_{k=-W-1}^{W+1}\sum_{l=-W-1}^{W+1} f^{(t)}_{i+k,j+l}}{(2W+3)^2} \tag{2}$$

The multiplicative averages $m^{(t)}_{1,i,j}$ are calculated using pixels within window size W around spatial coordinates (i,j) as the following equation.

$$m^{(t)}_{1,i,j} = \left(\prod_{k=-W}^{W}\prod_{l=-W}^{W}\left(f^{(t)}_{i+k,j+l} - f^{(t)}_{\min,1,i,j} + 1\right)\right)^{\frac{1}{(2W+1)^2}} + f^{(t)}_{\min,1,i,j} - 1 \tag{3}$$

where $f^{(t)}_{\min,1,i,j}$ is the minimum pixel value in the window of size W. Similarly, the multiplicative averages $m^{(t)}_{2,i,j}$ are calculated by window size $W+1$ as the following equation.

$$m^{(t)}_{2,i,j} = \left(\prod_{k=-W-1}^{W+1}\prod_{l=-W-1}^{W+1}\left(f^{(t)}_{i+k,j+l} - f^{(t)}_{\min,2,i,j} + 1\right)\right)^{\frac{1}{(2W+3)^2}} + f^{(t)}_{\min,2,i,j} - 1 \tag{4}$$

where $f^{(t)}_{\min,2,i,j}$ is the minimum pixel value in the window of size $W+1$. The values $g^{(t)}_{i,j}$ are calculated from $a^{(t)}_{1,i,j}$, $a^{(t)}_{2,i,j}$, $m^{(t)}_{1,i,j}$, and $m^{(t)}_{2,i,j}$ as the following equation.

$$g^{(t)}_{i,j} = a^{(t)}_{1,i,j} + a^{(t)}_{2,i,j} - m^{(t)}_{1,i,j} - m^{(t)}_{2,i,j} \tag{5}$$

The larger the scatter of the pixel values in the window is, the larger the difference between the additive and multiplicative averages becomes. On the other hand, the smaller the scatter is, the closer the difference approaches zero. Therefore, as the pixel values in the window are scattered, the values $g^{(t)}_{i,j}$ become large. Furthermore, as the scatter of the pixel values in the small and large windows is large, the values $g^{(t)}_{i,j}$ become large.

Step 2: The pixel values $f^{(t+1)}_{i,j}$ are updated using the values $g^{(t)}_{i,j}$ and the input pixel values $f_{i,j}$ as the following equation.

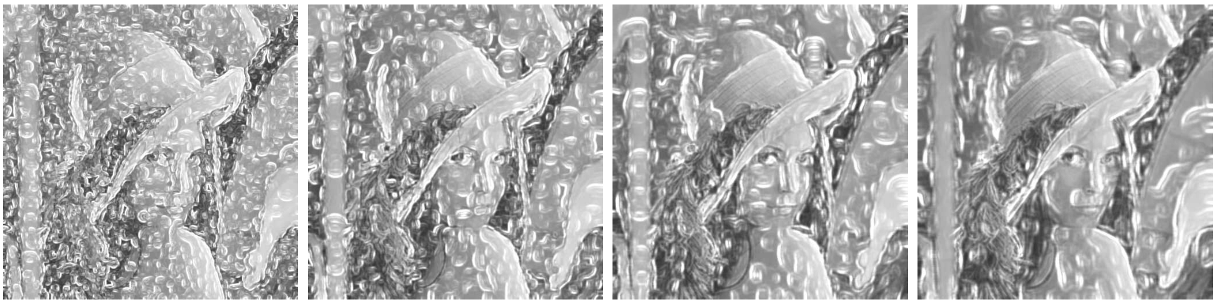
$$f^{(t+1)}_{i,j} = f_{i,j} + bg^{(t)}_{i,j} \tag{6}$$

where b is a positive constant. If $f^{(t+1)}_{i,j}$ is less than 0, then $f^{(t+1)}_{i,j}$ must be set to 0. If $f^{(t+1)}_{i,j}$ is greater than 255, then $f^{(t+1)}_{i,j}$ must be set to 255. The processing of Steps 1 and 2 is repeated T times, and then an image composed of the pixel values $f^{(T)}_{i,j}$ is the bubble image.

3. **Experiments.** Two experiments are conducted. The first experiment is visually confirmed bubble patterns by varying the values of the parameters in the proposed method using Lenna image shown in Figure 2. The second experiment is visually confirmed bubble images generated from other photographic images. All images used in the experiments are $512 * 512$ pixels and 256 gradation. And, the iteration number T is set to 100 at which bubble patterns were generated on the whole images.

First, bubble images by varying the window size W are confirmed visually using Lenna image. The window size W is set to 1, 2, 3, and 4. The parameter b is set to 3.0. The results of the experiment are shown in Figure 3. As the value of W is larger, the size of bubble patterns becomes bigger. Then, as the value of W is smaller, the density of bubble patterns increases and it is difficult to see the original image.

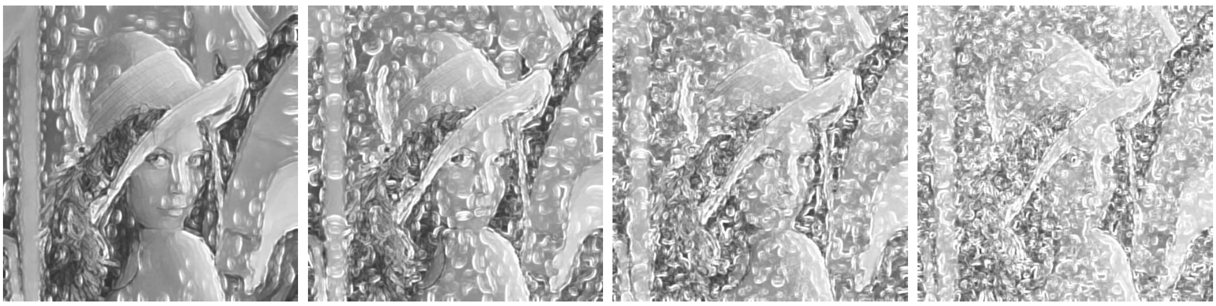
FIGURE 2. Lenna image



(a) $W = 1$ (b) $W = 2$ (c) $W = 3$ (d) $W = 4$

FIGURE 3. Bubble images in the case of the window size W = 1, 2, 3, and 4

Next, bubble images by varying the parameter b are confirmed visually using Lenna image. The parameter b is set to 2.5, 3.0, 3.5, and 4.0. The window size W is set to 2. The results of the experiment are shown in Figure 4. As the value of b is larger, the density of bubble patterns increases and it is difficult to see the original image. When using the proposed method, the user will adjust the size and density of bubble patterns by varying the values of W and b.



(a) $b = 2.5$ (b) $b = 3.0$ (c) $b = 3.5$ (d) $b = 4.0$

FIGURE 4. Bubble images in the case of the window size b = 2.5, 3.0, 3.5, and 4.0

Finally, the proposed method is applied to four gray-scale photographic images shown in Figure 5. With reference to the above experiments, to generate bubble patterns on the whole image and make the original image as easy to see as possible, the parameters W and b are set to 2 and 3.0, respectively. The results of the experiment are shown in Figure 6. In all bubble images, it can be seen that bubble patterns consist of visually distinct separate bubbles arranged in various distributions, and are automatically generated according to the density and contour of photographic images. There is also the method [16] for generating interference-ripple patterns similar to bubble patterns. In the

Figure 5. Photographic images

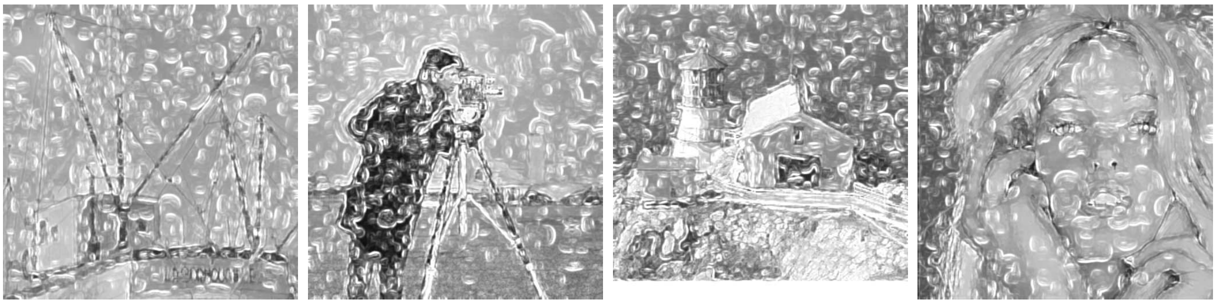


Figure 6. Bubble images

conventional method, there are regions where interference-ripple patterns hardly occur, but the proposed method can generate bubble patterns on the whole image.

4. **Conclusions.** This paper focused on NPR in nature, and then proposed a novel NPR method for generating bubble images from gray-scale photographic images. The proposed method used an iterative calculation using additive and multiplicative averages in different window sizes. As a result of experiments using various images, the three features of the proposed method became clear. The first feature is that the proposed method can generate impressive bubble images that bubble patterns consist of visually distinct separate bubbles arranged in various distributions. The second feature is that the proposed method can automatically generate according to the density and contour of photographic images. The third feature is that the proposed method can change the size and density of bubble patterns by varying the values of the parameters.

We hope that the proposed method can contribute to the development of the field of N-PR, and bubble images are utilized in the fields of design, amusement, and entertainment. A future study is to expand the proposed method for application to color photographic images and videos.

REFERENCES

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

[2] D. D. Seligmann and S. Feiner, Automated generation of intent-based 3D illustrations, *ACM SIGGRAPH Computer Graphics*, vol.25, no.4, pp.123-132, 1991.

[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] L. A. Gatys, A. S. Ecker and M. Bethge, Image style transfer using convolutional neural networks, *The IEEE Conference on Computer Vision and Pattern Recognition*, pp.2414-2423, 2016.

[6] W. Qian, D. Xu, K. Yue, Z. Guan, Y. Pu and Y. Shi, Gourd pyrography art simulating based on non-photorealistic rendering, *Multimedia Tools and Applications*, vol.76, no.13, pp.14559-14579, 2017.
[7] D. Martin, G. Arroyo, A. Rodriguez and T. Isenberg, A survey of digital stippling, *Computers & Graphics*, vol.67, pp.24-44, 2017.
[8] T. Wu, Saliency-aware generative art, *Proc. of the 10th International Conference on Machine Learning and Computing*, pp.198-202, 2018.
[9] H. Pedersen and K. Singh, Organic labyrinths and mazes, *Proc. of the 4th International Symposium on Non-Photorealistic Animation and Rendering*, pp.76-86, 2006.
[10] A. Sparavigna and B. Montrucchio, Non-photorealistic image rendering with a labyrinthine tiling, *Cornell University Library*, arXiv:cs/0609084, 2006.
[11] K. Inoue and K. Urahama, Halftoning with minimum spanning trees and its application to maze-like images, *ELSEVIER Computers & Graphics*, vol.33, no.5, pp.638-647, 2009.
[12] T. C. Inglis, S. Inglis and C. S. Kaplan, Op art rendering with lines and curves, *ELSEVIER Computers & Graphics*, vol.36, no.6, pp.607-621, 2012.
[13] A. G. M. Ahmed, Line-based rendering with truchet-like tiles, *CAe'14 Proc. of the Workshop on Computational Aesthetics*, pp.41-51, 2014.
[14] M. T. Chi, W. C. Liu and S. H. Hsu, Image stylization using anisotropic reaction diffusion, *The Visual Computer*, vol.32, no.12, pp.1549-1561, 2016.
[15] T. Hiraoka, M. Hirota, K. Inoue and K. Urahama, Generating cell-like color images by inverse iris filter, *ICIC Express Letters*, vol.11, no.2, pp.399-404, 2017.
[16] T. Hiraoka, Generation of interference-ripple images by inverse Sobel filter, *ICIC Express Letters*, vol.12, no.5, pp.409-415, 2018.