

GENERATION OF POINT-LIGHT-STYLE IMAGES USING PERIPHERAL DIFFERENCE FILTERS WITH DIFFERENT WINDOW SIZES

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

Received March 2022; accepted May 2022

ABSTRACT. *In non-photorealistic rendering (NPR), which converts photographic images into non-photorealistic images such as paintings and illustrations, many studies on NPR that imitate patterns in human society and nature have been conducted. We proposed an NPR method for generating point-light-style (PLS) images from photographic images. The proposed method is executed by an iterative calculation using two types of peripheral difference filters with different window sizes and can automatically generate the PLS patterns according to changes in edges and shade of photographic images. To verify the effectiveness of the proposed method, an experiment was conducted using various photographic images. An additional experiment was conducted to visually confirm the PLS images generated by changing the iteration number and window size.*

Keywords: Non-photorealistic rendering, Point light, Peripheral difference filter, Window size, Automatic generation

1. Introduction. Non-photorealistic rendering (NPR) is a generic term for computer graphics techniques [1, 2] that produce non-photorealistic images. Photographs are mainly used in modern times for the purpose of recording and communicating scenes. However, non-photorealistic images [3, 4, 5, 6, 7] such as paintings and illustrations are more suitable for communicating only a part of the scene or for incorporating the intentions of the artist into the scene. This is because only the necessary information can be deformed and the extra information can be omitted. On the other hand, there are also non-photorealistic images [8, 9, 10, 11] that incorporate geometric patterns in human society and nature. These non-photorealistic images are automatically or semi-automatically generated from photographic images, videos, and three-dimensional data. For example, in [8], reaction-diffusion patterns similar to patterns of fish and animals are automatically generated in Photoshop, in [9], fingerprint patterns are automatically generated using cosine-wave-weight smoothing filter from photographic images, respectively, in [10], cell patterns are automatically generated using inverse iris filter from videos, and in [11], moire patterns are automatically generated using bilateral filter and unsharp mask from three-dimensional data.

In this study, we propose a method for automatically generating point-light-style (PLS) images from photographic images using image processing techniques [12]. The PLS images are non-photorealistic images with white dots like light points that are placed throughout and along the edges of the photographic images. The proposed method is executed by iterative calculation using peripheral difference filters with different window sizes. In the process, a smoothing filter is introduced to reduce the change from the photographic images in areas other than the white dots. To verify the effectiveness of the proposed

method, we conducted an experiment to visually confirm the PLS images generated by changing the iteration number and the window size. Furthermore, we experimented with applying the proposed method to various photographic images.

As a study on NPR using the filter similar to the peripheral difference filter, a method to generate labyrinth-like images [13] from photographic images has been proposed. The labyrinth-like images are non-photorealistic images composed of equally spaced lines like a maze, and the labyrinth-like patterns generated by process using an inverse filter [14] and the improved Laplacian filters with different window sizes. Although the proposed method and the conventional method [13] are similar, the difference is that the proposed method uses a formula that does not use the inverse filter, two types of the peripheral difference filter, and the smoothing filter.

This paper is organized as follows: the second section describes the proposed method to generate the PLS images, the third section shows experimental results and reveals the effectiveness of the proposed method, and the conclusion of this paper is given in the fourth section.

2. Proposed Method. The proposed method is implemented in two steps using grayscale photographic images as input: 1) calculate the absolute values of the difference between the peripheral difference filtered values with the different window sizes, where the absolute values of the difference are smoothed; 2) convert the images using the absolute values. By repeating the two steps, PLS images are generated. The flow chart of the proposed method is shown in Figure 1.

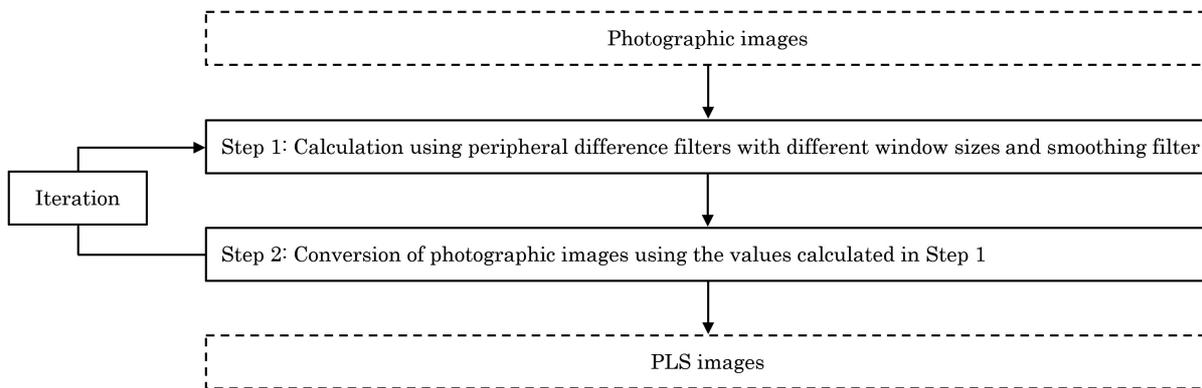


FIGURE 1. Flow chart of the proposed method

The detailed procedure of the proposed method is as follows.

Step 0: The input pixel values for the spatial coordinates (i, j) of a grayscale photographic image are denoted by $f_{i,j}$. Subsequently, the pixel values of the image at the t th iteration number are denoted by $f_{i,j}^{(t)}$, where $f_{i,j}^{(0)} = f_{i,j}$. The pixel values $f_{i,j}^{(t)}$ have values of U gradation from 0 to $U - 1$.

Step 1: The values $g_{1,i,j}^{(t)}$ and $g_{2,i,j}^{(t)}$ are respectively calculated using the two types of the peripheral difference filters with the window sizes W_1 and W_2 ($W_1 < W_2$) in the following equations:

$$g_{1,i,j}^{(t)} = \left| \frac{\sum_{k=-W_1}^{W_1} \sum_{l=-W_1}^{W_1} \substack{i \neq k \vee j \neq l \\ i+k, j+l} f_{i+k, j+l}^{(t-1)}}{(2W_1 + 1)^2 - 1} - f_{i,j}^{(t-1)} \right| \quad (1)$$

$$g_{2,i,j}^{(t)} = \frac{\sum_{k=-W_2}^{W_2} \sum_{l=-W_2}^{W_2} \substack{i \neq k \vee j \neq l \\ i+k, j+l} f_{i+k, j+l}^{(t-1)}}{(2W_2 + 1)^2 - 1} - f_{i,j}^{(t-1)} \quad (2)$$

where k and l are the positions in the window. Note that Equation (1) has an absolute value and Equation (2) has no absolute value. Equation (1) obtains the absolute value of the difference between the target pixel value and the average of the pixel values in the window (W_1) excluding the target pixel. Equation (2) obtains the difference between the target pixel value and the average of the pixel values in the window (W_2) excluding the target pixel.

The values $h_{1,i,j}^{(t)}$ and $h_{2,i,j}^{(t)}$ are respectively calculated from the values $g_{1,i,j}^{(t)}$ and $g_{2,i,j}^{(t)}$ using the smoothing filters with the window size W_1 in the following equations:

$$h_{1,i,j}^{(t)} = \frac{\sum_{k=-W_1}^{W_1} \sum_{l=-W_1}^{W_1} g_{1,i+k,j+l}^{(t)}}{(2W_1 + 1)^2} \quad (3)$$

$$h_{2,i,j}^{(t)} = \frac{\sum_{k=-W_1}^{W_1} \sum_{l=-W_1}^{W_1} g_{2,i+k,j+l}^{(t)}}{(2W_1 + 1)^2} \quad (4)$$

Equations (3) and (4) obtain the averages of the pixel values $g_{1,k,l}^{(t)}$ and $g_{2,k,l}^{(t)}$ in the window (W_1 and W_2), respectively.

The values $s_{i,j}^{(t)}$ are calculated using the values $h_{1,i,j}^{(t)}$ and $h_{2,i,j}^{(t)}$ in the following equations:

$$s_{i,j}'^{(t)} = \left| h_{1,i,j}^{(t)} - h_{2,i,j}^{(t)} \right| \quad (5)$$

$$s_{i,j}^{(t)} = (U - 1) \frac{s_{i,j}'^{(t)} - s_{\min,i,j}'^{(t)}}{s_{\max,i,j}'^{(t)} - s_{\min,i,j}'^{(t)}} \quad (6)$$

where $s_{\min,i,j}'^{(t)}$ and $s_{\max,i,j}'^{(t)}$ are the minimum and maximum values of $s_{i,j}'^{(t)}$, respectively. Equation (5) obtains the absolute value of the difference between the values $h_{1,i,j}^{(t)}$ and $h_{2,i,j}^{(t)}$. Equation (6) converts the value $s_{i,j}'^{(t)}$ from 0 and $U - 1$.

Step 2: The output pixel values $f_{i,j}^{(t)}$ at the t th iteration number are calculated using the values $s_{i,j}^{(t)}$ in the following equation:

$$f_{i,j}^{(t)} = f_{i,j} + s_{i,j}^{(t)} \quad (7)$$

where $f_{i,j}^{(t)}$ must be set to $U - 1$ if $f_{i,j}^{(t)}$ is greater than $U - 1$.

A PLS image is obtained after Steps 1 and 2, which involve T iterations.

3. Experiments. This paper mainly conducted two experiments. First, the experiment with changing the values of the parameters in the proposed method was conducted using the woman's image shown in Figure 2, and the way of generation of PLS patterns was confirmed visually. Next, the experiment was conducted using various photographic



FIGURE 2. Woman image

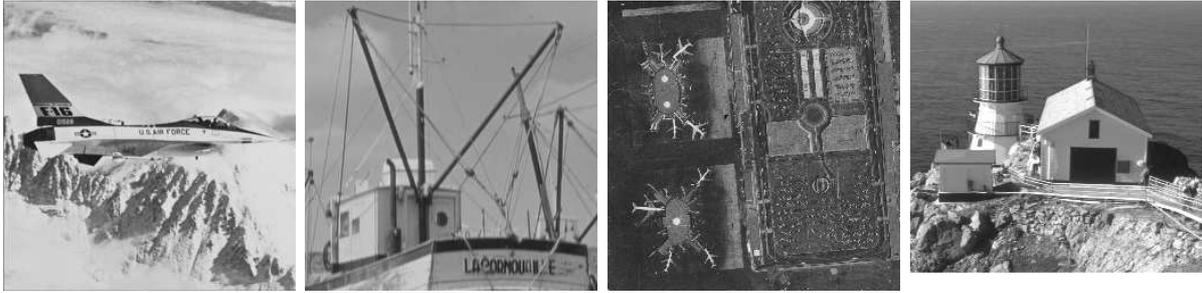


FIGURE 3. Various photographic images

images shown in Figure 3, and PLS images were confirmed visually. All photographic images used in the experiments comprised $512 * 512$ pixels and 256 gradations.

3.1. Experiment with changing parameters. PLS images generated by changing the iteration number T were visually confirmed using the woman's image. The iteration number T was set to 5, 10, 20, and 40. The other parameters W_1 and W_2 were set to 1 and 3, respectively. The PLS images generated under these conditions are shown in Figure 4. As the iteration number T increased, the PLS patterns became clearer.

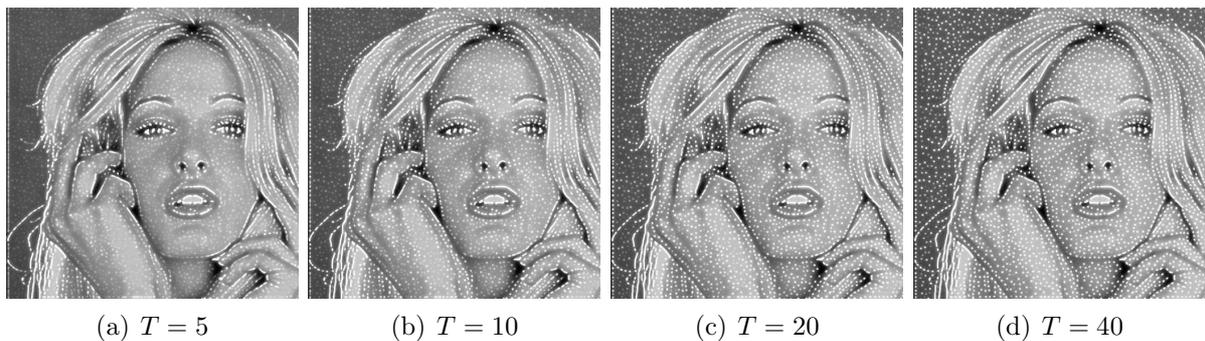


FIGURE 4. PLS images generated by changing the iteration number T

PLS images generated by changing the window size W_1 were visually confirmed using the woman's image. The window size W_1 was set to 1, 2, 3, and 4. The other parameters T and W_2 were set to 40 and 5, respectively. At this time, the window size W_2 needs to be set larger than the window size W_1 , so the window size W_2 is set to 5. The PLS images generated under these conditions are shown in Figure 5. The larger the window size W_1 , the larger the size of the PLS patterns and the darker the PLS patterns in the region far from the edges of the woman's image.

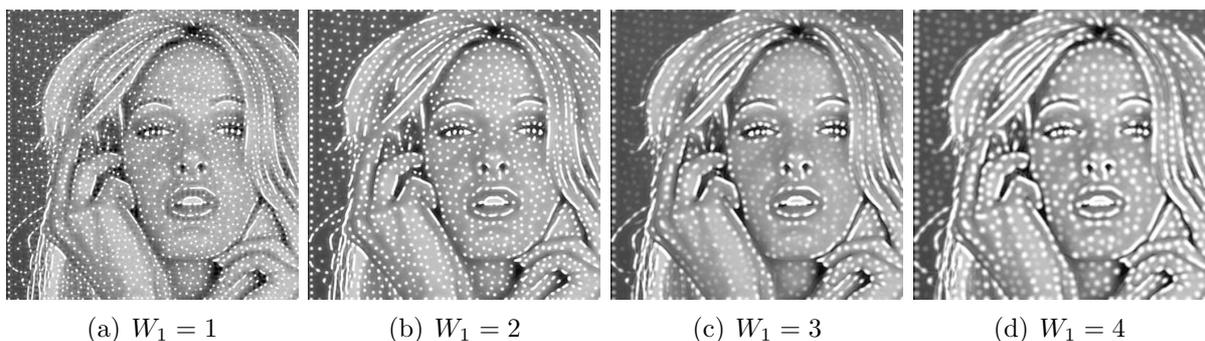


FIGURE 5. PLS images generated by changing the window size W_1

PLS images generated by changing the window size W_2 were visually confirmed using the woman's image. The window size W_2 was set to 2, 3, 4, and 5. The other parameters T and W_1 were set to 40 and 1, respectively. At this time, the window size W_1 needs to be set smaller than the window size W_2 , so the window size W_1 is set to 1. The PLS images generated under these conditions are shown in Figure 6. The larger the window size W_2 , the larger the size of the PLS patterns and the clearer the PLS patterns in the region far from the edges of the woman's image.

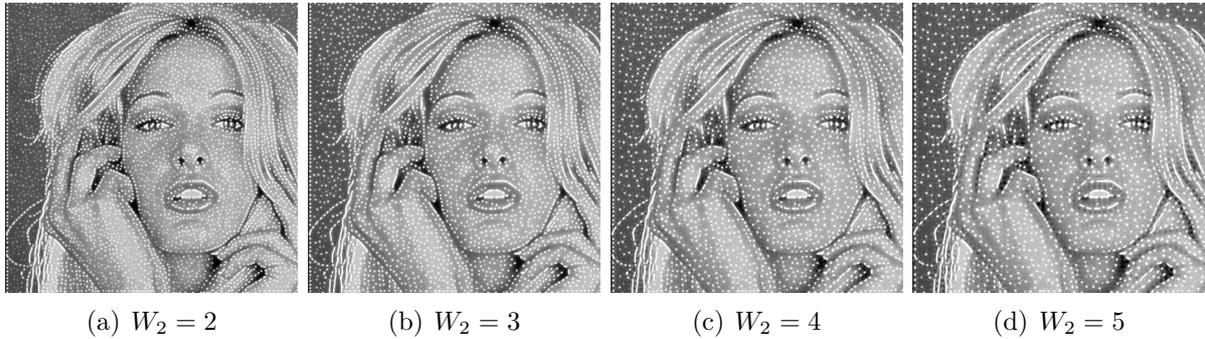


FIGURE 6. PLS images generated by changing the window size W_2

From the above, to clearly express the PLS patterns, it is necessary to set the iteration number T to 40 times or more in an image of $512 * 512$ pixels. To express the size of the PLS patterns larger, it is necessary to set the window sizes W_1 and W_2 to be larger. To clearly express the PLS patterns in the region far from the edges, it is necessary to set the window size W_1 to be smaller and the window size W_2 to be larger. The window size W_1 needs to be set smaller than the window size W_2 .

3.2. Experiment using various photographic images. The proposed method was applied to four photographic images shown in Figure 3. Since PLS patterns were visually recognized well in the previous experiments, the parameters T , W_1 , and W_2 were set to 40, 1, and 3, respectively. The PLS images generated under these conditions are shown in Figure 7. In all cases, the PLS patterns could be automatically generated according to changes in the edges and shade of photographic images. In addition, the PLS patterns could be placed throughout and along the edges of the photographic images.

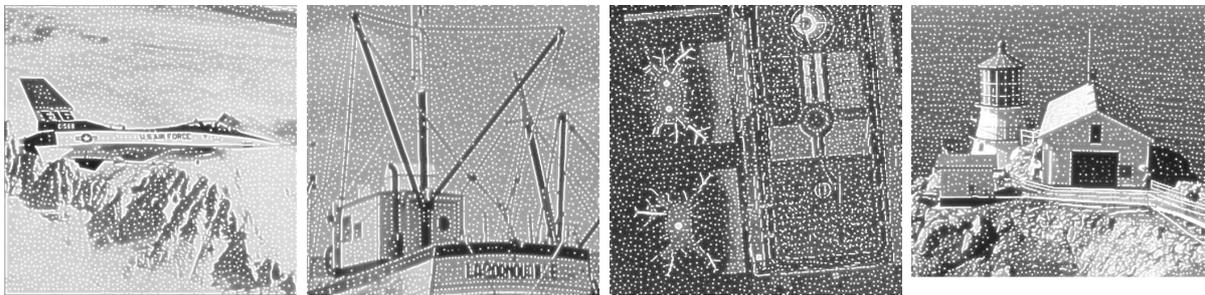


FIGURE 7. PLS images

4. Conclusions. We proposed an NPR method for automatically generating PLS images from photographic images. The proposed method was executed by an iterative calculation using two types of peripheral difference filters with different window sizes. To verify the effectiveness of the proposed method, we conducted experiments using various photographic images. Using the proposed method, the PLS patterns could be placed throughout and

along the edges of the photographic images. In addition, the size of the PLS patterns could be altered by changing the window size.

Future studies include extending the proposed method to applications involving color photographic images, videos, and three-dimensional data.

Acknowledgment. This work was supported by JSPS KAKENHI Grant Number JP19-K12664.

REFERENCES

- [1] A. Adil, S. Hadi and B. K. Triwijoyo, An interactive mobile augmented reality system for learning human vision structure, *ICIC Express Letters*, vol.15, no.12, pp.1337-1343, 2021.
- [2] T. M. Li, Differentiable visual computing: Challenges and opportunities, *IEEE Computer Graphics and Applications*, vol.42, no.2, pp.101-109, 2022.
- [3] P. Haeberli, Paint by numbers: Abstract image representations, *ACM SIGGRAPH Computer Graphics*, vol.24, no.4, pp.207-214, 1990.
- [4] 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.
- [5] D. Martin, G. Arroyo, A. Rodriguez and T. Isenberg, A survey of digital stippling, *Computers & Graphics*, vol.67, pp.24-44, 2017.
- [6] P. L. Rosin, Y. K. Lai, D. Mould, R. Yi, I. Berger, L. Doyle, S. Lee, C. Li, Y. J. Liu, A. Semo, A. Shamir, M. Son and H. Winnemoller, NPRportrait 1.0: A three-level benchmark for non-photorealistic rendering of portraits, *Computational Visual Media*, vol.8, no.3, pp.445-465, 2022.
- [7] Q. Sun, Y. Chen, W. Tao, H. Jiang, M. Zhang, K. Chen and M. Erdt, A GAN-based approach toward architectural line drawing colorization prototyping, *The Visual Computer*, vol.38, no.4, pp.1283-1300, 2022.
- [8] A. Werth, Turing patterns in Photoshop, *Proc. of Bridges 2015: Mathematics, Music, Art, Architecture, Culture*, pp.459-462, 2015.
- [9] T. Hiraoka, Generation of fingerprint-pattern images using cosine-wave-weight smoothing filter, *ICIC Express Letters*, vol.16, no.2, pp.153-158, 2022.
- [10] T. Hiraoka and M. Hirota, Generation of cell-like color animation by inverse iris filter, *ICIC Express Letters*, vol.12, no.1, pp.23-28, 2018.
- [11] R. Takaki and T. Hiraoka, Generation of moire-like images from RGB-D images, *ICIC Express Letters*, vol.15, no.1, pp.37-42, 2021.
- [12] Y. Zhao and Q. Huang, Image enhancement of robot welding seam based on wavelet transform and contrast guidance, *International Journal of Innovative Computing, Information and Control*, vol.18, no.1, pp.149-159, 2022.
- [13] T. Hiraoka, Generation of edge-enhancing labyrinth images using inverse filter and two improved Laplacian filters, *ICIC Express Letters*, vol.14, no.2, pp.121-127, 2020.
- [14] Z. Yu and K. Urahama, Iterative method for inverse nonlinear image processing, *IEICE Trans. Fundamentals*, vol.E97-A, no.2, pp.719-721, 2014.