A Method for Generating Pointillism Based on Seurat's Color Theory

Junichi Sugita†† and Tokiichiro Takahashi†† (member)

Abstract We propose a method for generating pointillistic style images from input image considering the features of Seurat's pointillism. Georges-Pierre Seurat is a pioneer and prime exponent of neo-impressionism. He established a technique called pointillism based on scientific color theory. There are three important features of Seurat's pointillism: optical mixture, complementary color contrast and halo effect. The most important feature is the optical mixture. To implement the optical mixture faithfully, we present a pointillistic halftoning method for color halftoning on random dots by utilizing a spatial data structure of boundary sampling algorithm. In addition, we implement complementary color contrast and halo effect according to actual Seurat's painting steps.

Keywords: Non-photorealistic rendering, error diffusion, pointillism, boundary sampling, optical mixture, Georges Seurat.

1. Introduction

Georges-Pierre Seurat (1859-1891) is a pioneer, and prime exponent of "neo-impressionism". He established a painting technique called "pointillism" in which small and distinct dots of pure colors are applied in order to form a picture, instead of long strokes. One of the most important features of pointillism in neo-impressionism is based on the color theory such as optical mixture, complementary color contrast, and halo effect. Seurat applied the color theory to color arrangements particularly.

Fig.1 (a) is the most famous work of Seurat, entitled "Sunday afternoon on the island of La Grande Jatte". According to the analysis on Seurat's work1,2, important characteristics of Seurat's pointillism are appeared in this work, and summarized as follows.

(1) **Pointillist painting**: As shown in Fig.1 (a), pointillism is formed by small and distinct dots of pure colors instead of long strokes.

(2) **Use of primitive colors**: 11 primitive color pigments from tubes are used without mixing them on his palette (Fig.2) for avoiding the reduction of saturation.

(3) **Control of luminance by white pigment**: Only white pigment is mixed to each color on his pallet in order to control the luminance as shown in Fig.2.

(4) **Optical mixture** is a visual effect of color fusion. Fig.1 (b) is an example of optical mixture. Juxtaposed vivid color dots cause similar spectra of colors...
because the additive color mixtures on a retina blend these dots into one color.

(5) **Complementary color contrast** is devised by Seurat. This visual effect generates very brilliant color contrast by placing two or more complementary colors side by side. It is used to express shading. Seurat selected complementary colors based on Chevreul’s color wheel (Fig.3 (a)) or RYB color wheel (Fig.3 (b)). Fig.1 (c) is an example of complementary color contrast, where we can see blue dots on a man in orange T-shirt.

(6) **Halo effect** was introduced to emphasize the contrast of the motif and background. Fig.1 (d) is an example of halo effect, where the contrast of background of black dress is enhanced.

(7) **Ebauche** is a kind of under-paint, and is drawn before painting pointillist strokes on a canvas. Ebauche makes sure no dots overlapped, and is visible between pointillist strokes.

Note that the important characteristics (4), (5), and (6) are derived from a scientific color theory.

In this paper, we propose a non-photorealistic rendering (NPR) method that generates pointillistic style images from input images taking into account of 7 important characteristics of Seurat’s pointillism. In our method, a given input image can be turned into a painterly style image with the presumably Seurat’s pointillism. To implement the optical mixture effectively, the authors propose a pointillistic halftoning method. The pointillistic halftoning method is available for color halftoning on random dots. Our method can express tonality of the input image by fewer colors efficiently compared with conventional methods.

Experimental results verify that our method can effectively generate pointillistic images with features of Seurat’s painting styles.

2. Related Work

Seurat learned the color theory through the books of Blanc⁴, Chevreul⁵ and Rood⁶. As already mentioned above, the characteristics of Seurat’s pointillism are summarized as follows: pointillist painting, use of primitive colors, control of luminance by white pigment, optical mixture, complementary color contrast, halo effect, and ebauche.

There have been proposed several methods to generate pointillistic images from given input images. However, there are fewer methods to simulate the whole characteristics of Seurat’s pointillistic painting styles.

2.1 Pointillistic Image Generation

Hertzmann⁷ proposed a method that generated various painterly images including pointillism incidentally by using curved brush strokes of multiple sizes.

Hays and Essa⁸ applied the same kinds of rendering techniques as Hertzmann to apply animation.

These two algorithms only perturb the color of the strokes randomly. Neither of these methods accurate pointillistic ways nor achieve the strict optical mixture effects.

2.2 Optical Mixture Only

Luong, et al.⁹ and Jin, et al.¹⁰ proposed methods for generating pointillism style images taking into account of optical mixture effect only. In their algorithms, juxtaposed colors have the same luminance determined by an additive mixing model to enhance human visual perception.

Hausner¹¹ also achieves the optical mixture effect by error diffusion approach efficiently, which relies on randomly placed dots. His algorithm builds a graph of pixels with links between neighbors on the image, and diffuse an error to neighbors by using the graph. However, they do not deal with other characteristics of the color theory of Seurat’s pointillism.

2.3 Pointillistic Image Generation Based on a Seurat’s Color Theory

There are a few studies to reproduce almost all the styles of the Seurat’s painting. Yang, et al.¹² proposed a method for simulating the Seurat’s pointillistic painting styles. Seo, et al.¹³ also proposed another generation method. These two methods simulated three important phenomena based on the color theory such as optical mixture, complementary color contrast and halo effect.

In order to implement optical mixture effect, both of their methods are used perturbs colors make near colors of input image color be chosen within a narrow region.

As a result, their methods required 72 hue values, although Seurat used only 11 hue values. This is
because their algorithms are too naïve to implement the optical mixture by using fewer colors. None of them adopt error diffusion techniques to reduce color differences.

Seo et al. took into account of shading expression by complementary colors. However, Yang et al. applied a simplistic manner on the whole canvas, without shading expression.

Both of Yang, et al. and Seo, et al. synthesized halo effects by applying different simple weighted filters, which yield very bright or very dark and contrasty images. Proposal method improves these problems mentioned above. Our improvements compared to previous methods are summarized at Table 1.

### 3. Pointillistic Image Generation Method

In this section, we describe the pointillistic image generation method from input image based on the color theory and steps. Overview of our method shows Fig.4.

First, we generate a halo filtered image (Fig.4 (b)) from input image (Fig.4 (a)). To calculate pure/complementary colors of rendered dots, the halo filtered image is used.

Second, we render pure color pointillist dots (Fig.4 (c)). These dots are rendered randomly but uniformly on a canvas.

Third, we have to render complementary color dots (Fig.4 (d)). Complementary color dots are placed more frequently in dark region for shading.

Fourth, ebauche is generated by using several image processing filters (Fig.4 (e)).

Finally, we synthesize three images in order mentioned above, which are ebauche, pure color pointillist painting and complementary color painting images for generating pointillistic image (Fig.4 (f)).

#### 3.1 Halo Effect

To generate halo, we use modified unsharp mask (modified UM), which introduces 2 types of weight function. Modified UM $U_i$ at pixel $i$ is expressed as follows:

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Optical mixture by fewer hue values</td>
<td>—</td>
<td>—</td>
<td>✓</td>
</tr>
<tr>
<td>primitive hue values</td>
<td>72</td>
<td>72</td>
<td>11</td>
</tr>
<tr>
<td>Shade expression by complementary color dots</td>
<td>Yang's method applied complementary color evenly on the whole canvas without shading expression</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Natural halo effect</td>
<td>These two methods can lose picture details because of extremely emphasized contrast of dark or bright region</td>
<td>Our method can generate halo with preserving details of input image</td>
<td></td>
</tr>
</tbody>
</table>

![Fig.4 Overview of proposal method.](image)
Where,

\[ U_i = I_i + w_{dist} \times w_{lum} \times (I_i - G_i) \]  \hspace{1cm} (1)

Where, \( I_i \) is the luminance at pixel \( i \) and \( G_i \) is a low frequency component, which is obtained using Gaussian filter with a kernel size \( s \). \( I_i - G_i \) is a high frequency component.

The \( w_{dist} \) is the weight factor for applying only surrounding edges defined as follows:

\[ w_{dist} = A \times \left( 1 - \frac{D_i}{s} \right) \]  \hspace{1cm} (2)

Where, \( A \) is a constant for tuning effect, \( D_i (0 \leq D_i \leq s) \) is a distance map obtained by distance transform from edges detected by Canny’s method\(^{14}\), \( s \) is the kernel size of Gaussian filter \( G_i \).

In bright or dark area of input image, the UM lead to marked loss of detailed information. For this reason, we introduced another weight factor \( w_{lum} \) as follows:

\[ w_{lum} = B \times \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left( -\frac{(L_i - \mu)^2}{2\sigma^2} \right) \]  \hspace{1cm} (3)

Where, \( L_i \) is a luminance at pixel \( i \). Here, constant \( B = 100 \), expectation \( \mu = 127 \) and variance \( \sigma = 40 \). This is a Gaussian distribution (peak is the intermediate value of luminance). In this weight function, the modified UM will not be applied to extremely bright or dark region.

**Fig.5** shows examples of halos generated by our method. **Fig.5** (a) is input image, (b) is edges detected from (a), and (c) is the distance map generated from (b). (d) is the result without weight function \( w_{lum} \), which is same as Seo’s result of halo generation\(^{13}\). (e) is our result of halo generation by using Equation 1. As shown in (d), we can observe lack of details, for example, the women’s hand is extremely darker. As shown in (e), we can observe improvement of them.

### 3.2 Pointillist Painting

We adopt a halftorning approach to implement the optical mixture well by fewer colors. Error diffusion method\(^{15}\) is a commonly used halftoning algorithm. Error diffusion relies on ordered grid of pixels. However, in pointillism, the dots are placed uniformly but randomly. The authors propose the pointillistic halftoning method, which is an algorithm were available for color halftoning on random dots by the using spatial data structure of boundary sampling algorithm\(^{16}\).

#### 3.2.1 Boundary Sampling Algorithm

This section describes the method to determine the disk locations by boundary sampling algorithm. Boundary sampling is a variation of Poisson disk sampling algorithm\(^{17}\) that runs in linear time and space. Boundary sampling algorithm samples a halo filtered image, and places disks sequentially to appropriate locations on the canvas randomly but uniformly. The procedure is as follows.

**Step1:** A disk, radius \( r \), is sampled at random.

**Step2:** Push the disk sampled at Step1 into the candidate list \( \{ S_k \} \).

**Step3:** Pop the disk \( S_i \in \{ S_k \} \) out of candidate list \( \{ S_k \} \) randomly.

**Step4:** The disk \( S_j \) is sampled on the circumference of the circle at the center of the disk \( S_i \) from Step3 with radius \( 2r \). A collection of circular arcs centered at \( S_j \), which enable sampling \( S_j \) called available boundary. An example of available boundary shows **Fig.6** (a). As shown in **Fig.6** (a), solid line portion is available boundary.

**Step5:** Step4 is recurred until available boundary is undetectable. Then push the disk sampled on available boundary into the candidate list \( \{ S_k \} \).

![Fig.5 Generation of Halos](image)

![Fig.6 A spatial data structure of boundary sampling](image)
Step 6: Step 3~5 is recurred until the candidate list $\{S_k\}$ is empty.

### 3.2.2 Error Diffusion

Pointillist dots are rendered by colors of each sampled point which determined by boundary sampling mentioned previous section. However, Seurat had been used 11 primitive color pigments. Thus we require quantization of halo filtered image colors. We adopt error diffusion method to express tonality of input image. In this section, we will show a method to diffuse error to sampled disks.

We diffuse error to disks sampled by boundary sampling algorithm. In HSL color space, we quantize each channel, hue, saturation, lightness. Then the difference between input image value and quantized value is error. We diffuse the error to the collection of disks $\{S_m\}$ on the available boundary at the center of the arbitrary disk $S_i$, which is sampled by the boundary sampling algorithm. The $m$ is the number of disks on available boundary of $S_i$.

First, the color, which is each channel of hue, saturation, lightness in HSL color space, at disk $S_i$ is quantized. Then, define the set of quantized hue values as $\{H_n\}$, saturation as $\{S_a\}$, lightness as $\{L_n\}$, the hue values of input image at point $S_i$ as $H_{\text{input}}$, saturation as $S_{a\text{input}}$, lightness as $L_{\text{input}}$. Each set of quantized values divided each HSL value into each $n_H$, $n_S$, $n_L$ equal part. Where the number of $n_H$, $n_S$, $n_L$ is 24, $n_H$ is as a parameter. Categorize the HSL values of input image as one of the each $\{H_n\}$, $\{S_a\}$, $\{L_n\}$. After that, calculate Euclidean distance $d$ between each HSL value of input images and each set of quantized HSL values, and replace the HSL values of point $S_i$ with the minimum value of distance $d$. Here, the error of each channel after replacement is defined as following formula:

$$
\begin{align*}
e_H &= H_{\text{input}} - H_n \\
e_S &= S_{a\text{input}} - S_a \\
e_L &= L_{\text{input}} - L_n
\end{align*}
$$

Next, diffuse the error to the points $\{S_m\}$ on available boundary (See Fig. 6 (b)) as following formula:

$$
\begin{align*}
H_n' &= H_n + D_L \times \frac{e_H}{m} \\
S_a' &= S_a + \frac{e_S}{m} \\
L_n' &= L_n + \frac{e_L}{m}
\end{align*}
$$

Where, $m$ is the number of disks on available boundary at $S_i$, and $D_L$ is inverse function of probability density function of Laplace distribution and defined as following formula:

$$
D_L = \begin{cases} 
\phi \ln 2U + \mu & \text{if } (U < 0.5) \\
-\left( \phi \ln 2(1-U) + \mu \right) & \text{otherwise}
\end{cases}
$$

Where, the parameter $\mu$ is the mean or expectation (location of the peak), $2\phi$ is the variance and $U$ is uniform random numbers in the range [0, 1]. In this paper, $\mu = 0$, $\phi = 6$. See$^{13}$ reported his analysis that Laplace distribution is the best fit for Seurat's color arrangement. We diffuse the amplified error $e_H$, $-\pi/2 \leq e_H \leq \pi/2$, based on Laplace distribution.

### 3.2.3 Effectiveness of Pointillistic Halftoning

To the verifying effect of the optical mixture, the authors performed preliminary experiments for generating a halftoned image by our pointillistic halftoning method. Fig. 7 shows the results of generating pointillistic halftoning method by altering the number of quantize hue values from 11 (Fig. 7 (a)) to 72 (Fig. 7 (e)). As shown in Fig. 7, we can observe the gradation of the color wheel by optical mixture. Especially, we could perceive the tonality of the color wheel with less number of primitive colors.

### 3.3 Complementary Color Contrast

Seurat uses complementary colors more frequently in dark areas for shading. In order to shade expression by using complementary color, we modify the pointillistic halftoning method for generating stippling$^{18}$. Here, it is necessary to distinguish between stippling and pointillism.

![Fig.7 Results of the pointillistic halftoning method by using different colors.](image-url)
In stippling, the focus is mainly on the placement of points, where are used to produce shading. Complementary color dots are used for expression of shading. Stippling is used for deciding the location of dots rendered with complementary colors. We enable both optical mixture and complementary color contrast by error-diffusion-based approach with the spatial data structure of the boundary sampling algorithm.

### 3.3.1 Stippling for Complementary Color

After replacing the color to monochrome, diffuse the error to a neighboring point by using spatial data structure of the boundary sampling algorithm. For adapting the stippling to the error diffusion algorithm, we are placing black dots rather than black (luminance \(= 0\)) and white (luminance \(= 255\)) pixels. We could place stipple rendered with complementary color for every black point.

Diffuse the error to the set of points \(\{S_m\}\) on available boundary at arbitrarily disk \(S_i\). Here, \(m\) is a number of disks on available boundary at \(S_i\). Then, luminance \(f_i \in [0, 255]\) of input image at \(S_i\) is updated as following formula:

\[
g_i = \begin{cases} 
255 & \text{if } (f_i > 127) \\
0 & \text{otherwise} 
\end{cases}
\]  

(7)

Here, the error after replacement as \(e = f_i - g_i\). Diffuse the error \(e\) to the points \(\{S_m\}\) on available boundary:

\[
I'_n = I_n + w_s \times e 
\]  

(8)

Where \(I'_n\) is the new luminance at point \(\{S_m\}\) after diffusing error, \(I_n\) is the luminance of input image at point \(\{S_m\}\) and the density of point is controlled by spatially-related adjustment weight factor\(^{19}\) \(w_s\), which is gamma collection. Gamma collection is the effectively technique for tuning contrast. Here, weight factor \(w_s\) is defined as following formula:

\[
w_s = \begin{cases} 
\left(\frac{1}{r_s}\right)^G & \text{if } (e < 0) \\
\left(\frac{r_s}{G}\right)^G & \text{otherwise} 
\end{cases}
\]  

(9)

Where, \(r_s\) is the radius of the disks at point \(S_i\) calculated by Equation 10 described in Section 4.5. \(G_+\) and \(G_-\) are able to control error: increasing positive error and decreasing negative error.

#### 3.3.2 Calculation of Complementary Color

Dots are rendered by complementary color on the set of points \(\{S_{stat}\}\) of generated stippling in the previous section. Fig.8 shows a method for determining complementary color in HSL color space. First, perturb the hue value of input image at \(\{S_{stat}\}\) based on a Laplace distribution using Equation 6. Next, Complementary color is calculated using RYB color wheel\(^{20}\) (Fig.3 (b)), which is similar to Chevreul’s color wheel (Fig.3 (a)). Calculate complementary color by adding \(\Pi\) to the hue value in HSL color space, which convert from RYB color space. Finally, each channel of hue, saturation, lightness are quantized as is the case with the previous section.

#### 3.3.3 Effectiveness of Complementary Color

Let us demonstrate the shade expression by complementary color dots. Fig.9 (a) shows the result of pointillist painting without complementary color dots. Fig.9 (b) shows the result of pointillist painting with complementary color dots. As shown in Fig.9 (b), we can observe many complementary color dots in the dark areas, but the brighter areas contain fewer complementary colors.

### 3.4 Ebauche

As mentioned in Section 1, under-paint or outline called ebauche is created before pointillist stroke paintings. To simulate ebauche, covert input image to smoothed image with a faint color. First, colors of input image are reduced by popularity algorithm\(^{21}\). Next, the image is smoothed by median filter.

#### 3.5 Stroke Rendering

Pointillistic image is composited by many textures that imitate short brush stroke as shown in Fig.10. The
strokes have the following properties:

**Position, Color:** The way to determine the position and the color of strokes is different in pure color and complementary color. In pure color, position and color are determined as mentioned in Section 3.2. In complementary color, they are also determined as mentioned in Section 3.3.

**Size:** In pointillism, short strokes are used.

**Pure color:** The size of strokes is basically determined by the radius of the disk of the boundary sampling algorithm. In actual pointillism, each stroke has multiple sizes by hand. Thus radius $r$ of disks is modified based on the luminance values of input image as following formula:

$$r' = r_{\text{min}} + \left( \frac{r_{\text{max}} - r_{\text{min}}}{255} \right) \times (255 - I_{\text{original}})$$  (10)

Where, $r_{\text{max}} = r \times \alpha_p$, $r_{\text{min}} = r \times \beta_p$, $\alpha_p$ and $\beta_p$ is parameters. $I_{\text{original}}$ is a luminance of input image at sampled point. Radius $r$ varies linearly with the original intensity $I_{\text{original}}$ in the range $r_{\text{min}} \leq r' \leq r_{\text{max}}$.

The width and height of the texture are specified by $\text{width} = 2r'$, $\text{height} = 1.5 \times \text{width}$ as shown in Fig. 10.

**Complementary color:** Strokes painted with complementary colors become less dense as compared to the case of pure color pointillist strokes. Therefore, large size of the radius of the boundary sampling algorithm as compared pure color strokes is used. As a parameter, the radius $r_c$ for rendering complementary color strokes in distinction from radius $r_p$ of boundary sampling for complementary color stippling is defined. The $r_c$ is modified by Equation 10 in range $r_{\text{min}} \leq r_c \leq r_{\text{max}}$.

Where, $r_{\text{max}} = r_c \times \alpha_c$, $r_{\text{min}} = r_c \times \beta_c$, $\alpha_c$ and $\beta_c$ is parameters.

**Direction:** Seurat paint strokes aligned with the contours of the motif. Thus the strokes should flow along the edges of input image. To achieve this, the direction of the texture is determined by applying an edge tangent flow (ETF). The ETF is extracted by computing the gradient magnitude and tangent orientation. Then the ETF is smoothed by them of the neighboring pixels.

**Order:** The order of rendering textures of stroke is same as the order of the boundary sampling algorithm. First, pointillist strokes of pure colors are rendered on ebauche. Next, previously rendered image added with strokes of complementary colors.

### 4. Experimental Results

To verify that proposed method is able to generate pointillistic image effectively from input image, the authors performed several experiments.

#### 4.1 Examples of Generated Pointillistic Images

Fig. 11 shows generated results by the proposed method. Table 2 lists the parameters used in our method. Where the parameters of pointillistic halftoning describe $P(0.003, 1.5, 1.2, 11)$. The notation $P(0.003, 1.5, 1.2)$ means $r = 0.003$, $\alpha_p = 1.5$, $\beta_p = 1.2$, $n_H = 11$. The parameters of stippling for complementary color described as following: the notation $P_c(0.012, 10, 5, 1.2, 1.0, 0.003)$ means $r_s = 0.012$, $G_- = 10$, $G_+ = 5$, $\alpha = 1.2$, $\beta = 1.0$, $r_p = 0.003$. (c) is generated from (a) as $P(0.003, 1.7, 1.3, 11)$, $P_c(0.008, 10, 10, 1.5, 1.2, 0.003)$. (d) is generated from (b) as $P(0.003, 1.7, 1.4, 11)$, $P_c(0.008, 5, 5, 1.5, 1.2, 0.0025)$.

#### 4.2 Validation for Features of Seurat's Pointillism

Fig. 12 shows how our algorithm simulates some of the features of Seurat's pointillism. (a) is generated as $P(0.002, 1.7, 1.5, 11)$, $P_c(0.007, 10, 10, 1.6, 1.4, 0.002)$. (b), (c), (d) show detail of (a). We can observe the juxtaposition of different color dots from (b), more complementary colors at the shaded region from (c), and that halo are expressed from (d).

#### 4.3 Comparison with Actual Seurat's Work

Fig. 13 demonstrates our claimed resemblance by placing Seurat’s paintings and our generated results side by side for comparison. (a) and (b) compare halo

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>$r$</td>
<td>Radius of sampled disks for pointillist painting in the range [0, 1].</td>
</tr>
<tr>
<td>$\alpha_p$</td>
<td>Scale factor of $r_p$ (equation 10) for pointillist painting in the range [0, 1].</td>
</tr>
<tr>
<td>$\beta_p$</td>
<td>Scale factor of $r_p$ (equation 10) for pointillist painting in the range [0, 1].</td>
</tr>
<tr>
<td>$\alpha_c$</td>
<td>Number of hue for quantization.</td>
</tr>
<tr>
<td>$r_c$</td>
<td>Radius of sampled disks for stippling by complementary colors in the range [0, 1].</td>
</tr>
<tr>
<td>$G_- G_+$</td>
<td>The parameters for adjusting the density of stippling by complementary colors for light and dark region respectively (equation 9).</td>
</tr>
<tr>
<td>$\alpha_s$</td>
<td>Scale factor of $r_s$ (equation 10) for stippling by complementary colors in the range [0, 1].</td>
</tr>
<tr>
<td>$\beta_s$</td>
<td>Scale factor of $r_s$ (equation 10) for stippling by complementary colors in the range [0, 1].</td>
</tr>
<tr>
<td>$r_p$</td>
<td>The size of radius for rendering complementary color strokes in the range [0, 1].</td>
</tr>
</tbody>
</table>
effect, while (c) and (d) color juxtaposition, which is similar in coloration. Through these comparisons, we believe that our method can express characteristics of Seurat’s pointillism faithful.

4.4 Comparison with Various Parameters

It is possible to generate various images by altering the parameters. Several examples are shown in Fig.14. (b), (c), (e) shows examples generated by altering the size and density of dots while keeping the other parameters. Control of size and density of dots is important, because the size and density of dots have a relationship to the optical mixture closely. We tried some cases altering the

Fig.11 Results of our pointillistic image generation method.

Fig.12 Characteristics of Seurat’s pointillism mimicked by the proposed method.
number of hue values $n_H = 11$ in (d), $n_H = 24$ in (e), $n_H = 72$ in (f). We could observe color reproducibility by the optical mixture even with less number of colors.

Most potentially influential parameter in regard to the appearance of generated image is a radius $r$ of sampled disks. We might well be able to get an effective outcome as
pegging \( r \) at range \([0.002, 0.003]\) approximately. Seurat had been drawing complementary color strokes smaller than pure color pointillist strokes. Thus \( r_c \) should be set smaller values than \( r \). \( \alpha_p = 1.7, \beta_p = 1.4, \) are considered reasonable and proper. In the same manner as \( r_c \), parameters \( \alpha_c \) and \( \beta_c \) should be set smaller than \( \alpha_p \) and \( \beta_p \) respectively. The parameter \( r_s \) at range \([0.007, 0.008]\) are considered proper.

### 4.5 Comparison with Conventional Methods

Fig.15 compares our generated result with conventional methods. (a) is input image, (b) is the result of Yang’s method\(^{16}\), (c) is Seo’s one\(^{17}\) and (d) is proposed method. (d) is generated as \( P(0.0025, 1.8, 1.4, 11), P_c(0.008, 5, 5, 1.7, 1.3, 0.0025) \). By contrast with our method obtained using 11 hue values, conventional method requires 72 hue values for color reproduction.

Table 3 shows PSNR values of pointillistic images generated by our method (Fig.15 (d)) and the conventional methods (Fig.15 (b) and (c)). PSNR values of Table 3 are calculated by following formula:

\[
PSNR = 10 \log_{10} \frac{255^2}{\sigma}
\]

(11)

Where, \( \sigma \) is represented by:

\[
\sigma = \frac{1}{3mm} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} \left[ R(i,j) - \overline{R(i,j)} + G(i,j) - \overline{G(i,j)} + B(i,j) - \overline{B(i,j)} \right]^2
\]

(12)

Here, \( m \) and \( n \) is width and height of an image. \( [R(i,j), G(i,j), B(i,j)] \) is RGB values at position \((i, j)\) of input image. \( [\overline{R(i,j)}, \overline{G(i,j)}, \overline{B(i,j)}] \) is RGB values at position \((i, j)\) of pointillistic image.

In our method, we calculate PSNR values with altering hue values as 11, 24 and 72. As shown in Table 3, we can observe that the more the number of hue, the better reproduction rate of the input image tonality is achieved. However, we can also see that PSNR value of our method with 11 hue values is better than PSNR values of conventional methods with 72 values. This can be considered as an effect of the error diffusion method. Thus our method can express tonality of input images by fewer colors efficiently. It is said that Seurat was using 11 colors as the original colors, and regarding this point, we believe that our proposed method is superior to other conventional methods.

As mentioned in Section 4.1, conventional method for generating halo has the possibilities of loss of detailed information. Our method improved this problem.

### 5. Conclusion

The method of generating pointillistic image was proposed in this paper corresponding to the features of Seurat’s pointillism, which including optical mixture, complementary color contrast and halo effect. Especially, the pointillistic halftoning method using the spatial data structure of the boundary sampling algorithm for
applying error diffusion to uniform but random sampled points was proposed. Proposed method enabled the optical mixture faithful with fewer hue values. The authors demonstrated the effect of the proposed method by generating several experimental results. The authors believe that it is possible to get closer to Seurat’s style.

In future work, the authors would like to reveal the relationship of distance between viewer and generated image to size of dots for best optical mixture. In any case, starting with a 2D image imposes limitations, which is a lack of depth information and a lot of noise. The authors would like to consider generating the pointillistic image from 3D scene.

References

6) O.N. Rood: "Modern Chromatics, with Applications to Art and Industry", Appleton, New York (1879)
8) J. Hays and I. Essa: "Image and Video Based Painterly Animation" 2004 Non-Photorealistic Animation and Rendering, pp.113-120 (June 2004)
13) S.H. Seo and K.H. Yoon: "Color Juxtaposition for Pointillism Based on an Artistic Color Model and a Statistical Analysis", The Visual Computer, 26, 6-8, pp.421-431 (June 2010)

Junichi Sugita received his B.E. and M.E. degrees from Tokyo Denki University, Japan, in 2006 and 2008, respectively. He had been working at Toppan Printing Co., Ltd. from 2008 to 2012. Since 2012, he has been an assistant in Faculty of Healthcare, Tokyo Healthcare University. He received IIEJ Nishita award in 2008. His research interests include computer graphics, especially non-photorealistic rendering.

Tokiichiro Takahashi received his B. E. degrees from Niigata University in 1977, and Ph. D. from the University of Tokyo in 2005, respectively. He had been working at Nippon Telegraph and Telephone Corporation since 1977. Since 2003, he has been a professor at Tokyo Denki University. His research interests include computer graphics, image processing, and visual computing.