芸術科学会論文誌 Vol. 13, No. 3, pp. 177-184

Shader Based Generated Ornamentation for Rendering Wayang Beber of Pacitan Character's Cloth Pattern

Banung GRAHITA **Toshihiro KOMMA** **Kumiko KUSHIYAMA**

Tokyo MetropolitanUniversity

banung-grahita@ed.tmu.ac.jp komma@tmu.ac.jp kushi@ea.mbn.or.jp

Abstract

Wayang Beber of Pacitan is one type of wayang beber, a traditional picture from Indonesia. It is usually performed to tell a story. Wayang beber of Pacitan has unique visual form, distinctive shape and complex ornamental detail. In past days, wayang beber had an important role within the local society, however it is barely known today. These conditions are our motivation to do a project to develop a CG animation of wayang beber of Pacitan. We want to re-introduce wayang beber of Pacitan and find a traditional artistic value from it that could be applied to modern media. This paper explains a research as a part of the project. This research is developing a program that could be used to generate and control patterns on wayang beber of Pacitan character's cloth. The algorithm of our program consists of two important parts: elements and rules. The elements define a basic shape used in pattern arrangement, and the rules are the procedures to arrange the elements. There are two rules in our algorithm, the first rule is a condition to do iteration of the coloration process, and the second rule is a condition to define the area of the coloration process. Our program allows artists to generate and control variation of the pattern.

Keywords: computer graphic, pattern generation, animation, non-photorealistic rendering, Indonesian cultural heritage

1. Introduction

This reseach is part of a project to produce a CG animation of "wayang beber of Pacitan". Wayang beber of Pacitan is an old Indonesian traditional picture made around the year 1692 [1]. It is usually performed to tell a story. One set of wayang beber of Pacitan contains six scrolls of picture; each scroll has 4 scenes of story (see fig.1).

Figure1. The sample of original wayang beber of Pacitan scrolls.

Wayang beber of Pacitan existence is endangered today. It is rarely performed and barely known; most of Indonesian people especially the young generation never seen it. In addition, the condition of its scrolls is very poor, some part of the picture already gone. Related to this condition, we have two motivations to do the project:

1. We want re-introduce the wayang beber Pacitan to the current generation using a new medium that close related to them such as CG animation.
2. We want to seek a new concept of CG animation that based on a local artistic tradition.

The project preceded by two researches, the research on the artistic issues and the research on technical issues (see Fig.2). The research on the artistic issues has important role; it is the foundation of the project. The research on artistic issues examines the typical visual features, artistic values and philosophy of wayang beber of Pacitan picture. This research found 3 important visual features in the character figures of wayang beber of Pacitan, those are:

1. The distinctive figure shape.
2. Thick outline.
3. Complex ornamental pattern on the clothes.

Complete explanation of this research could be seen in [2]. The result of the research on artistic issues is used as an art direction in subsequent research.

The subsequent research is a research on technical issues. This research tries to develop a technique to create CG animation of wayang beber of Pacitan, especially a technique to simulate the visual form. This research is divided into several phases. First phase is a research to generate outline in the 3DCG figure. The result of this research could be seen in [3], second is a research to generate patterns of wayang beber of Pacitan character's cloth, and third is a research to create the figure shape of wayang beber of pacitan. This paper explains the second phase, the research to generate pattern of the character's cloth.

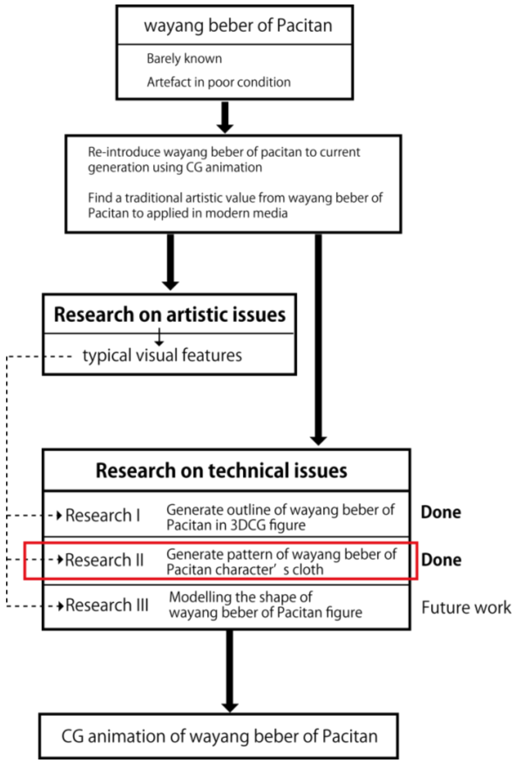


Figure2. The research plan

1.1 Problem Statement

In CG production, the common technique to make a pattern on 3DCG surface is using a texture map. However, this technique is unsuitable for our project. The texture map technique obtains the pattern from a bitmap file. Modifying the pattern can only be done by change the bitmap file. To simulate wayang beber of Pacitan picture in CG, the pattern must be modified several times. There are many characters in wayang beber of Pacitan that has different cloth pattern, sometimes they have similar pattern with a few variation. Using different bitmap file for each character will result a big file size. Also modifying a bitmap file one by one is time consuming.

Figure3. Variation of wayang beber of Pacitan character's cloth pattern

For the project, we need different approach to create a pattern on 3DCG surface. We need an approach that could easily generate and modify pattern in each surface. Therefore, in this project we use a procedural pattern. The procedural pattern is a pattern that generated and could be controlled by a small program called shader [4]. This paper explains the research to develop a shader that could be used to simulate wayang beber of Pacitan cloth pattern.

1.2 Related works

The programing approach of generated pattern or ornamentation has been explored for many years. Geng [5] has reviewed several researches on this field. He categorized the techniques into several approaches; two of them are closely related to our research. Those are art pattern creation by shape grammar and lay out based creation of art patterns.

One of the popular methods is the layout based creation of art pattern. Many types of art pattern have obvious structural characteristic, using this method, artist can effectively create a design by making changes onto layout of these desirable art patterns [6]. One example of this method is a work by Hamekasi and Samavati [7].

Hamekasi and Samavati's work presented an analysis of Persian floral patterns based on circle packing. This approach separates the semantics and combinatorics of the design from its geometry. It could generate new designs that have high level characteristics such as visual balance and imperfect symmetry. However this works only suitable for generating a vines floral pattern.

Another approach is art pattern by shape grammars. Shape grammars specify a mechanism for selecting and performing rules for shape computation. This approach gives the designers freedom to specify, explore, develop a design language in terms of the shape grammar, and select alternatives for desirable art patterns [5]. This approach is applied in a work by Wong, Zongker, and Salesin. [8]. Their work presents a grammar based approach to generate floral patterns using an adaptive clip art which consist of a set of element and their growth rules. The pattern generation starts from user-specified seed points, and then the system will decide which element to grow. The Wong's system is automatic algorithmic system, in this system the artist has less control over the pattern. For the project, we need a semi-automatic system, which allow artist to control some variation of the pattern detail.

2. Features of Wayang Beber of Pacitan Character's Cloth Pattern.

The original scrolls of wayang beber of Pacitan condition is very poor, therefore we use its replica as a reference image. This replica made by a member of wayang beber metropolitan community in Jakarta. It is only depicts a part of fifth scroll of wayang beber of Pacitan but imitates the original picture very well (see fig.4).

Figure4. The comparison between the original picture of wayang beber of Pacitan and its replica

We studied the reference image and found some typical features of wayang beber of Pacitan character's cloth pattern. We categorized the pattern into three types (see Fig.5), those are:

1. line pattern
2. flower pattern
3. vines pattern

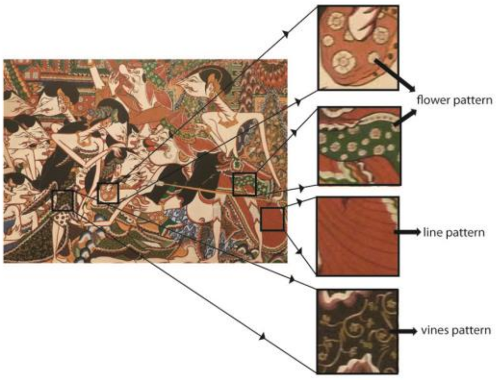


Figure 5. Three types of wayang beber of Pacitan character's cloth pattern

The line pattern is the simplest one. It is a thin vertical line drawn repeatedly across the cloth (see fig.6). Sukir [9] explain that this line pattern is called cawi. It is a typical visual feature appears not only on wayang beber, but also in another types of wayang called wayang kulit. This feature is used to fill up a wide blank area in the picture to maintain a good visual balance.

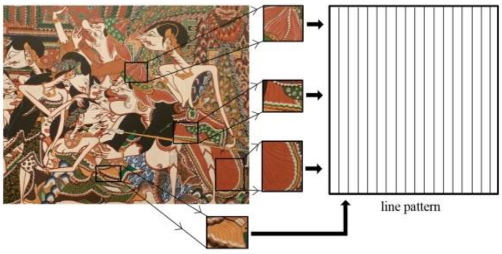


Figure 6. Line pattern on wayang beber of Pacitan's cloth

The other two patterns are floral patterns. There is no literature that explains the floral pattern on wayang beber of Pacitan character, so we use a literature of another traditional art to study the pattern. Nowadays we could easily find these similar floral patterns in a popular Indonesian traditional cloth called batik. In the past, batik cloth is usually worn by Javanese king or noble, we thought this is what is depicted in wayang beber of Pacitan.

We did a comparison between the floral pattern on wayang beber of Pacitan and the floral pattern on batik cloth, so we could get a better understanding about these pattern's visual structures. There are two floral patterns on wayang beber of Pacitan, the flower pattern, and the vines pattern.

The pattern on batik cloth that similar to flower pattern is called ceplok, and the one that similar to vines pattern is called sulur-suluran (see fig.7). We studied the ceplok and sulur pattern in batik cloth. The ceplok pattern has philosophical meaning of giving hope, wisdom, or guidance [10]. The ceplok pattern is one of the oldest traditional patterns in Java, Indonesia. It could be seen in many sculptures from the ancient Buddha and Hindu [11]. Then it is very likely if this pattern also found in wayang beber of pacitan.

We apply the visual structure of ceplok and sulur pattern composition to create the flower and vines pattern of wayang beber of Pacitan's cloth.

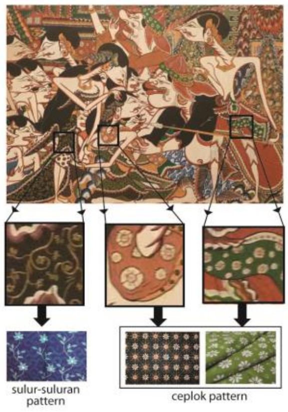
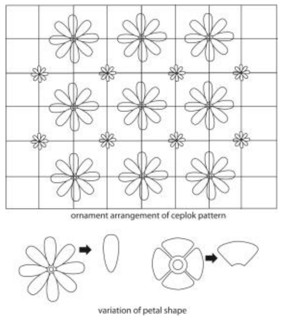


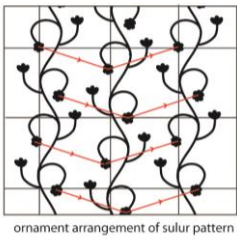
Figure 7. Comparison between the pattern of wayang beber of Pacitan's cloth and the pattern of batik clothes

Ceplok pattern is classified as a geometrical pattern. In this type of batik the ornaments are arranged geometrically, and particularly in ceplok pattern, the ornaments are arranged in a square shape (see fig.8a). Batik ceplok has various types of ornaments or motifs; one of them is called truntum. Truntum is a motif of small flower, such as jasmine. This ornament often depicted with eight petals, but sometimes depicted with four or six petals, with various shape.

Sulur (or sulur-suluran) pattern is classified as a free form pattern. This type of pattern has no specified rules but usually arranged vertically (see fig.8b). This pattern consists of three part, those are: main trunk, or vine, leaf, and flower.



(a)



(b)

Figure8. (a).The composition of ceplok pattern. (b) sulur pattern on batik cloth

3. Rendering Wayang Beber of Pacitan's Cloth

For the experiment, the 3D CG model of wayang beber of Pacitan's cloth is made in Autodesk MAYA and rendered by PRman, a renderman compliant renderer developed by Pixar. The 3D model is converted to RIB (Renderman Interface Bytestream) file as the input for renderer. The renderer applies two kinds of shader to render the image through the RIB file. The first shader is used to generate the outline; it has developed in our previous research [3]. The second shader used to render the color, line, and the shape patterns. This paper explains development of this second shader. The shaders are developed using RSL (Renderman Shading Language).

3.1 Developing the Pattern Shader

Our pattern shader algorithm basically works in texture space of the 3D model. Texture space is 2 dimensional space, defined by two axis, *x* and *y*, or in this case we called *s* and *t*. The coordinate of s and t both lie between 0.0 and 1.0. They represent a point location inside dimensionless image of unit size

The shader generates the shape of an element on the 3D CG surface by defining color position based on its coordinate in texture space; this process is called "s t coloration".

We adopt the Wong's [8] approach in our shader algorithm. We use the concept that similar to adaptive clip art, which consists of two parts: elements and rules. The elements correspond to the 2D geometric primitives that appear on the ornament, and the rules are the procedures to arrange the elements.

3.1.1 The Elements

We define 7 basic elements that used to arrange wayang beber of Pacitan's cloth pattern (see fig.9).

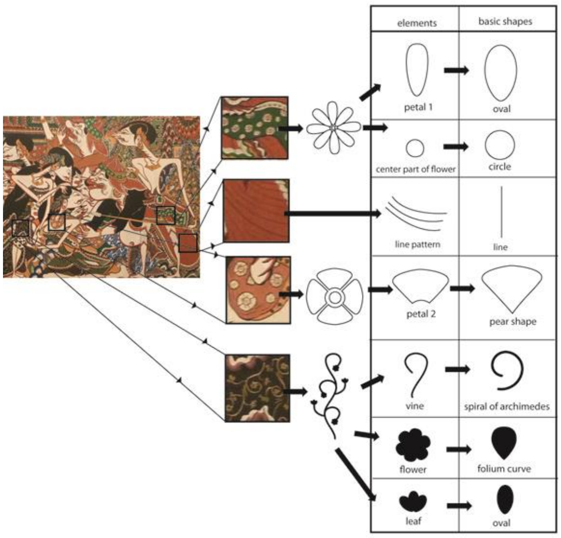


Figure9. The elements of wayang beber of Pacitan's cloth pattern.

The line pattern only consist of one element, it is a vertical thin line.

The flower pattern has two kinds of petal. The first petal could be created using an oval shape and the second petal could be created using pear shape quartic. The center part of the flower could be created using a circle.

The vines pattern arranged by three shapes; A spiral of Archimedes used for the vine, a folium used for the flower, and an oval used for the leaf.

Each element has attributes that controlled by rules.

3.1.2 The Rules

The rules are conditional statement that should be met for doing the coloration process. The rules divided into two:

1. The first rule is a condition for the iteration process. While this condition met, the iteration of coloration process will continued.
2. The second rule is a condition that describes the coordinate of coloration area. The coloration process will be performed in the areas that meet this condition. The second rule will determine the position of the element on the texture space.

3.1.2.1 Generating Line Pattern

We develop our rules or procedures to generate vertical line pattern based on Kendall Sor's ruled paper algorithm [12].

The first rule for line pattern is a procedure to repeat the coloration process in the texture coordinate. A new parameter is needed to define the color positions; it should be a periodically repeated numbers. The modulo operation is applied to the *s* and *t* value to obtain this parameter. The modulo operation finds the remainder of division of one number by another (see the following equation).

$$r_s = s - \left\lfloor \frac{s}{n} \right\rfloor \times n \quad (1)$$

$$r_t = t - \left\lfloor \frac{t}{n} \right\rfloor \times n \quad (2)$$

Where r_s is the remainder of *s* after the modulo operation is applied, $\left\lfloor \frac{s}{n} \right\rfloor$ is the largest integer less than or equal to $\frac{s}{n}$, n is the divisor, r_s is the remainder of *t* after the modulo operation is applied $\left\lfloor \frac{t}{n} \right\rfloor$ is the largest integer less than or equal to $\frac{t}{n}$.

To control the line width (thickness) we add a new variable *w*. We define it using the following equation.

$$w = \frac{n \times l}{10} \quad (3)$$

Where *n* is gap between the lines, and *l* is a value given by user to control the line width.

The second rule is a conditional statement for coloration process. The coloration will be done if a following condition is met.

$$r_s < w \quad (4)$$

Where r_s is a remainder of *s*, used to repeat the coloration process, and *w* is a parameter to control the line width.

The result can be seen in fig. 10. A variation of line width obtained by applying *w* value can be seen in fig.11

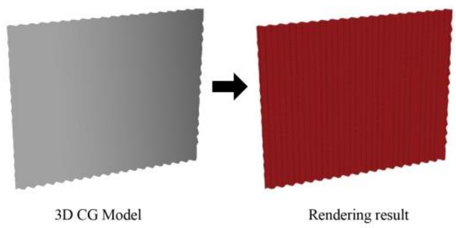


Figure10. Generated line pattern

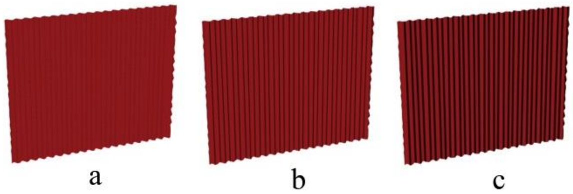


Figure11. Variation of line width

3.1.2.2 Generating Flower Pattern

To generate a flower pattern, r_s and r_t are used as coordinate of coloration area, r_s is result of modulo operatio on s value, while rt is result of modulo operation in t value. Both rs and r_t are repetitive numbers on vertical and horizontal axis. They divides the surface into number of square modules (see Fig.12). The coloration is done in every module.

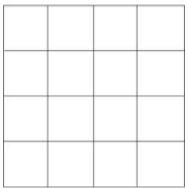


Figure12. Flower pattern module

The coloration begins by applying the first rule. The first rule of flower pattern is a conditional statement for iteration of the petal coloration process. To create flower petal configuration, the coordinate of the coloration area rotates gradually in every iteration. The rotation is done by applying following equation:

$$s_r = r_s \cos\alpha - r_t \sin\alpha \quad (5)$$

$$t_r = r_s \sin\alpha + r_t \cos\alpha \quad (6)$$

Where s_r is a new s coordinate of coloration position after rotation, t_r is a new t coordinate of coloration position after rotation, and α is the pattern rotation angle.

In every iteration α is recalculated, it is added by a value that given by user. The iteration performed until the pattern rotation angle surpasses 360 degree.

Then the second rule or procedure is applied. The second rule for the coloration is a conditional statement to controls the coloration area shape in texture space. The coloration will be done if the condition is met. To create the flower petal user could choose two shapes, those are oval, and pear shape quartic. The condition for oval shape is:

$$sr^2 + 0.3^{s_r} \times 2 \times t_t^2 <= z \quad (7)$$

Where s_r is a new s coordinate of coloration position after rotation, t_r is a new t coordinate of coloration position after rotation, and z is the petal size.

The condition for pear shape quartic is:

$$b \times (t_r \times \frac{c}{z})^2 < (s_r \times \frac{c}{z})^3 \times (a - (s_r \times \frac{c}{z})) \quad (8)$$

Where a,b, and c are arbitrary real numbers.

Center of the flower is made by a circle. The condition for the circle is:

$$s_r^2 + t_r^2 < z \quad (9)$$

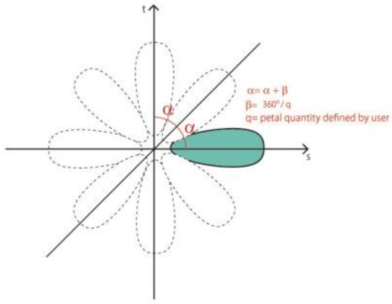


Figure13. Iteration of flower pattern coloration

Some cloth has two types of flower pattern, big flower and the small one. The size of the small flower is half of the big one. It defined by:

$$z_s = \frac{z}{2} \quad (10)$$

The small flower petal could made by an oval shape, the condition for the oval shape is:

$$sr^2 + 0.3^{s_r} \times 2 \times t_t^2 <= z_s \quad (11)$$

It also could made by pear shape quartic, the condition for pear shape quartic is:

$$b \times (t_r \times \frac{c}{z_s})^2 < (s_r \times \frac{c}{z_s})^3 \times (a - (s_r \times \frac{c}{z_s})) \quad (12)$$

Center of the small flower is made by a circle. The condition for the circle is:

$$s_r^2 + t_r^2 < z_s \quad (13)$$

The result can be seen in fig.14.

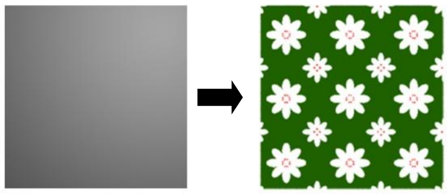


Figure14. Generated flower pattern

3.1.2.3 Generating Vines Pattern

To generate a vines pattern r_s and t value are used as coordinate of coloration area. The r_s is the result of modulo operation on s value, it is repetitive numbers on horizontal axis. It divides the surface vertically into several modules (see fig.15). The coloration is done in every module.

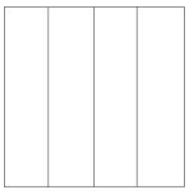


Figure15. Vines pattern module

The vines pattern has two types of element that are subjected by different rules. The first type is vine, and the second type is flower and leaf.

The first rule for the vines is a condition for iteration of coloration process. This rule allow t value be added by 0.4 in every iteration. The iteration performed until the t value surpasses 1, this value is a limit of the surface's border.

The first rule for the flower and the leaf allow its coloration area to rotate gradually in every iteration. The rules are similar to the first rule of flower pattern, the iteration performed until the pattern rotation angle surpasses 360 degree.

The second rule is a condition for the coloration process. Shape of the vine is created using spiral of Archimedes. The condition for this shape is defined by:

$$s_{pos}1 < (R \times \theta) < s_{pos}2 \tag{14}$$

Where s_{pos1} is a first s coordinate of the coloration area in the texture space, and s_{pos2} is a second s coordinate of the coloration area in the texture space. The s_{pos2} must bigger than s_{pos1}, R is radial coordinate of the spiral, and θ is angular coordinate of the spiral.

The spiral of Archimedes is defined by a polar equation, therefore R and θ values must be calculated by converting the texture coordinates (s,t) to their polar equivalents. The conversion can be done by following equations:

$$R = \sqrt{(r_s - 0.1)^2 + (t - 0.2)^2} \tag{15}$$

$$\theta = arctan\left(\frac{t-0.2}{r_s-0.1}\right) \tag{16}$$

Besides the vine, vines pattern also has flowers and leafs. The coloration for the flower will be done in a condition if:

$$((r_s \times z - s_{pos})^2 + (t \times z - t_{pos})^2) \times ((t \times z - t_{pos})^2 + (r_s \times z - s_{pos}) \times ((r_s \times z - s_{pos}) + b)) < 4a \times (r_s \times z - s_{pos}) \times (t \times z - t_{pos})^2 \tag{17}$$

Where r_s is the result of modulo operation on s value, s_{pos} is s coordinate of the coloration position, t_{pos} is t coordinate of the coloration position, z is size, a, and b are arbitrary real numbers.

And the the coloration for the leaf will be done in a condition if:

$$(r_s - s_{pos})^2 + (0.3^{(r_s - s_{pos})} \times 2 \times (t - t_{pos})^2) <= z \tag{18}$$

The result can be seen in fig.16.

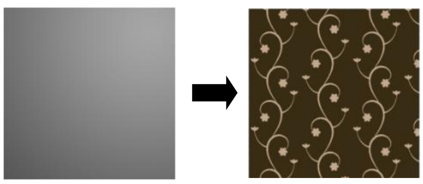


Figure16. Generated vines pattern

Our algorithm also allows artists to control certain attributes of the pattern, such as pattern types, line width, petal types, petal size, petal quantity, vines sizes, and pattern color. The overall algorithm summarized by the following pseudocode.

```
Input the type of pattern.
Apply modulo operation to s value using equation (1).
Apply modulo operation to t value using equation (2).
If type of pattern = "line",
    Then, input color, input line width.
    Calculate line width (w) using equation (3).
    Calculate line coloration area using equation (4).
    Output color (result can be seen fig.10).
Else if type of pattern =" flower pattern".
    Then input how many petal types will use, 1 or 2.
        Input petal shape, size, quantity (q), and input color.
        Input second petal shape, quantity (q), and color if petal types = 2.
        Declare the petal angle (α), α = 0
        Calculate the rotation angle (β), β= 360 / q
        Calculate second petal size using equation (10)
        Apply first rule of flower pattern's coloration.
        While α ≤ 360
            Calculate s value rotation using equation (5).
            Calculate t value rotation using equation (6).
            Apply second rule of flower pattern's coloration.
            If petal shape = oval, then
                Calculate coloration area using equation (7).
            Else if petal shape = oval, then
                Calculate coloration area using equation (8).
            If petal types= 2, then
                If second petal shape = oval, then
                    Calculate coloration area using equation (11).
                Else if second petal shape = pear, then
                    Calculate coloration area using equation (12).
            End if.
            Output color.
            If α > 360, then break loop.
            Recalculate α, α=α+β
        Calculate center of first flower coloration area using equation (9).
        Calculate center of second flower coloration area using equation (13) if
        petal types = 2.
        Output color
         (If petal types =1, result can be seen fig.18).
         (If petal types =2, result can be seen fig.19).
Else if type of pattern =" vines pattern",
    Then, input vine size, and color.
    Apply first rule of vines coloration.
    While t ≤ 1
        Calculate radial coordinate (R) using equation (15).
        Calculate angular coordinate (θ) using equation (16).
        Apply second rule of vines coloration.
        Calculate vine coloration area using equation (17).
        Output color.
        If t > 1, then break loop.
        Recalculate t, t=t +0.4
    Apply first rule of flower coloration.
    While α ≤ 360
        Calculate flower coloration area using equation (18).
        Output color.
        If α > 360, then break loop.
        Recalculate α, α=α+β
    Apply first rule of flower coloration.
    While α ≤ 180
        Calculate flower coloration area using equation (19).
        Output color (result can be seen fig.20).
        If α > 360, then break loop.
        Recalculate α, α=α+β
```

4. Result

Our shader could produce three types of pattern; line pattern, flower pattern, and vines pattern. The flower pattern is arranged geometrically in square modules, it resemble a typical arrangement of ceplok pattern composition which is also similar with the composition of wayang beber of Pacitan's flower pattern. The vines pattern is arranged vertically, it resemble the arrangement of sulur pattern.

Our shader allow artist to control certain parameters of the pattern. Using these parameters, the detail of the pattern, such as petal shape, petal size, petals quantity, color, and arrangement could be easily modified. This shader's feature solved our problem to render many character's cloth with certain variation of pattern. The comparison of rendering result using different pattern types can be seen in fig.17.

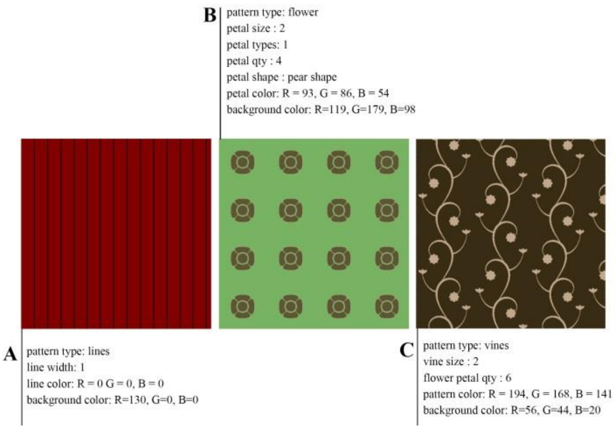


Figure17. Comparison of rendering result using various pattern types and color

Fig. 17A is rendered using line pattern type, fig. 17B is flower pattern type, and fig. 17C is vines pattern types.

The comparison of flower pattern rendering result using different petal quantity, shape, and color can be seen in fig.18.

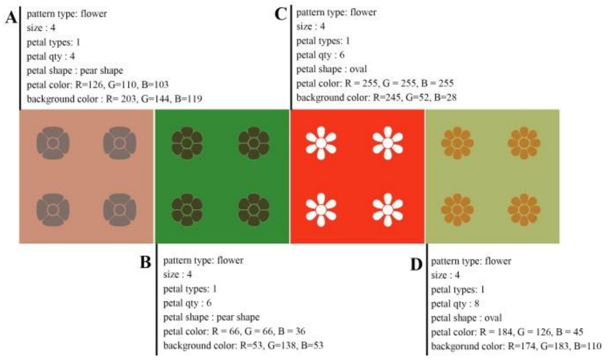


Figure18. Comparison of flower pattern rendering result using various petal shape, quantity, and color

Fig.18A and 18B is rendered using pear shape, fig.18C and 18A is rendered using oval shape. Fig.18A is using 4 petals, fig 18B, and 18C is using 6 petals, and fig.18D is using 8 petals.

Using our shader user could make variation flower arrangement by define how many petal types that used in the arrangement. User could choose whether 1 or 2 pattern types used in the arrangement. The comparison of flower pattern rendering result using different petal arrangement, and color can be seen in fig.19.

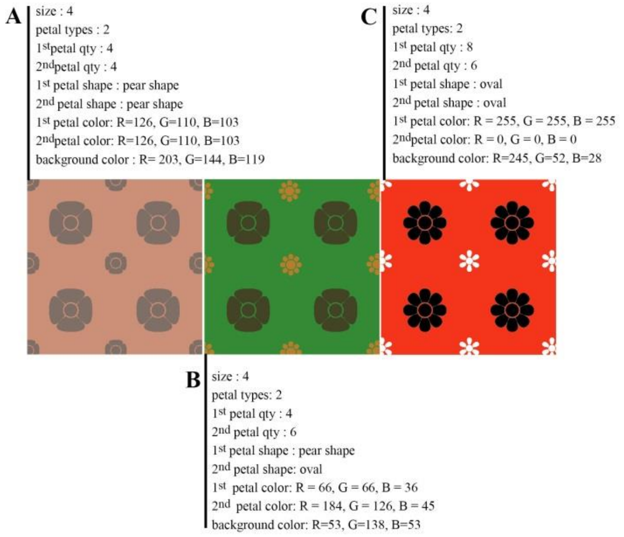


Figure19. Comparison of rendering result using various pattern arrangements

The figure 19A shows a flower pattern arrangement using two similar petal types, and quantity. While fig.19B and 19C shows a flower arrangement using two different petal types, and quantity.

The comparison of rendering result using different pattern size can be seen in fig.20.

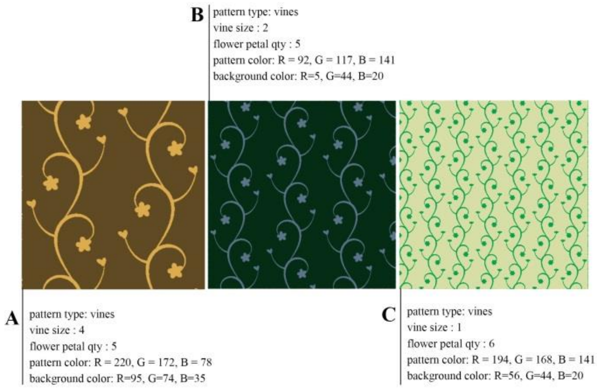


Figure20. Comparison of vines pattern rendering result using various pattern sizes, and color.

The figure 20A shows a pattern with a size =4, the figure 20B show a pattern with a size = 2, while fig. 18C shows a pattern with a size = 1.

This shader will be used in a CG animation production. It could apply to 3D CG model surface easily. The Implementation of our shader in 3D CG models can be seen in figure 21and 22.

Figure21. Implementation of our shader in 3D CG models

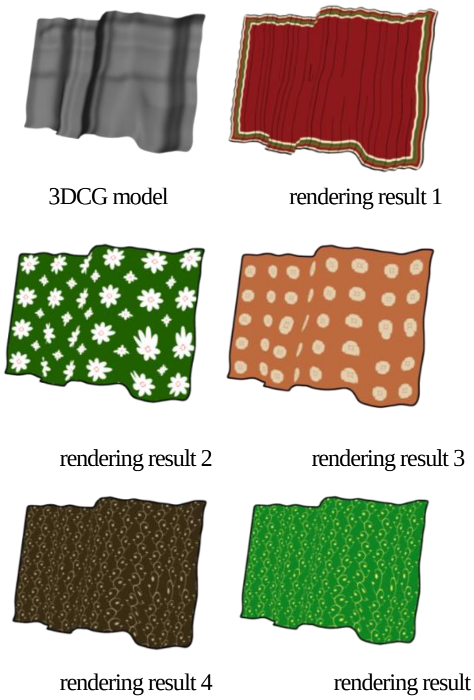


Figure22. Implementation of our shader in 3DCG models

5. Conclusion

We develop a shader with a semi-automatic system to generate the flower pattern of wayang beber of Pacitan character's cloth. Using this system, artist can control several parameters and modify the detail of the pattern such as petal size, petal quantity, color, and arrangement.

The algorithm of our shader adopts the adaptive clip art concept. This concept consist two important things, the elements and the rules. The element is basic shape that used in a pattern arrangement. At this time our shader has 7 elements, these element is sufficient to arrange the pattern of wayang beber of Pacitan's cloth. In the future, it is possible to add more elements into our shader to create more various patterns.

The rules are certain procedures that used to arrange the element. Our algorithm consists of two rules, the first rule is a condition to do iteration of the coloration process, and the second rule is a condition to define the area of the coloration process.

6. Future Works

Our final goal is to adopt the wayang beber of pacitan visual form into an animated film using 3D CG technology. The wayang beber of Pacitan visual form has are many features. There are several technical issues regarding this matter, we have solved two of them. As planned, we have one more issue to solve about modelling wayang beber of Pacitan figure shape in 3D CG.

In addition to those already planned, there are more research could be done regarding our animation project. For instance, research on modelling the wayang beber of Pacitan character. Wayang beber of Pacitan character has a distinctive figure. We need a specific research to simulate and animate it in CG.

Acknowledgement

Author gratefully acknowledges Japan International Cooperation Agency (JICA) and Bandung Institute of Technology (ITB) for the financial support through JICA-ITB scholarship.

References

[1] Tabrani, P. *Bahasa Rupa (third edition)*, Kelir, pp.37, 2102

[2] Grahita, B., Komma,T., Kushiyama, K., *"Visual Analysis of Wayang Beber Pacitan Character Figure"*, JSGS 2012 conference proceeding, pp.71-74, 2012.

[3] Grahita, B, Komma, T., Kushiyama, K.*," Generating Wayang Beber of Pacitan Character's Outline Using Renderman Interface"*, International Journal of ADADA, Vol.17, No.2, pp. 68-75, 2013.

[4] R. Cortes, S. Raghavachary, *"The Renderman Shading Language Guide*", Thomson, pp 361, 2008.

[5] Geng, W., *The Algorithm and Principles of Non-photorealistic Graphic*s, Zeihan University Press – Springer, pp. 91, 2009.

[6] Geng, W., *The Algorithm and Principles of Non-photorealistic Graphics*, Zeihan University Press – Springer, pp. 100, 2009.

[7] Hamekasi, N., Samavati, F., *Designing Persian Floral Patterns using Circle Packing*, GRAPP/IVAPP, pp. 135-142, 2012.

[8] Wong, Michael T., Zongker, D. E., Salesin, D. H., *Computer-Generated Floral Ornament*. Proceedings of the 25th annual conference on Computer graphics and interactive techniques. ACM, pp. 423-434, 1998.

[9] Sukir, *Bab Natah Sarta Nyungging Ringgit Wacucal*, Departemen Pendidikan dan Kebudayaan. Jakarta, p64-65, 1980.

[10] Moertini, V. S., Sitohang, B., *Algorithms of Clustering and Classifying Batik Images Based on Color, Contrast and Motif*. Department of Informatics Engineering–Bandung Institute of Technology, 2005.

[11] Rosanto, A., *Kajian Batik Motif Kawung Dan Parang Dengan Pendekatan Estetika Seni Nusantara,*. Brikolase, Vol.1, No.2, 2012.

[12] R. Cortes, S. Raghavachary, *"The Renderman Shading Lamguage Guide"*, Thomson, pp 373-375, 2008.

Authors:

Banung Grahita

He received bachelor and master degree in design from Bandung institute of Technology, Indonesia in 2006 and 2009. Currently studies as a doctoral student in Industrial Art Department, Graduate School of System Design, Tokyo Metropolitan University, specializing in media design.

Toshihiro Komma

He is a Professor in Industrial Art Department, Faculty of System Design, Tokyo Metropolitan University. He graduated from Tokai University in 1979, and received doctoral degree from Tokyo University of Agriculture and Technology in 2014. Until 2005 he worked as Professor in Shobi University. He also has worked for IMAGICA Corporation, NKK Corporation, Japan Computer Graphics Lab, and Japan Patent Information Organization.

Kumiko Kushiyama

She is a Professor in Industrial Art Department, Faculty of System Design, Tokyo Metropolitan University. Her fields of specialization are media art, interactive art, virtual reality and research on the sense of touch. Her works and research have been presented at the SIGGRAPH Art Gallery, Ars electronica, and as part of the collection of science centers in Japan.