



CNIDOSCOLUS ACONITIFOLIUS MODEL IN BATIK PATTERN GENERATION BY USING RANDOM WALK

Purba Daru Kusuma

School of Electrical Engineering, Telkom University, Bandung, Indonesia

E-Mail: purbodaru@gmail.com

ABSTRACT

This paper discusses the development of Cnidoscolus aconitifolius leaves model as batik pattern. Cnidoscolus aconitifolius is a papaya like tree that is popular in Indonesia. Even this plant leaf are similar to the papaya leaf, basically this plant is different from the papaya family. In the other side, papaya leaf pattern has been used as a plant based traditional batik pattern because its specific finger like leaf morphology. So, developing other finger like leaf batik pattern by using computational technology is challenging. In this research, besides developing the leaf model, the plant branching model is developed too. The reason is to make difference between the Cnidoscolus aconitifolius with the common papaya tree that does not have branching characteristics. This model is developed by using random walk as its basis method. After the basic model is developed, this model then is used to generate batik pattern computationally in web based batik pattern generation application. Based on the tests, there are several research findings. The inter petiole distance has negative correlation with the number of generated leaves. The lobe angle deviation has positive correlation with the leaf circular width. The sting hair step angle deviation has positive correlation with the lobe width. Small value of the sting hair step angle deviation can produce Picuda like leaf. Medium value of the sting hair step angle deviation can produce Estrella like leaf.

Keywords: cnidoscolus aconitifolius, papaya, batik pattern, random walk, L-system.

INTRODUCTION

Batik is one of Indonesian traditional cultural heritage. As a famous cultural heritage, batik is also popular not only in Indonesia but also in the world. Batik also has been recognized as Indonesian cultural heritage by UNESCO [1-4]. This recognition has ended the dispute between Indonesia and Malaysia about batik [3, 4].

Indonesia is a very big country which consists of many tribes and communities. This condition makes the cultural diversity in Indonesia is very rich. This diversity also occurs in batik pattern. There are many local batik patterns in Indonesia, such as Megamendung in Cirebon [5], Sidomukti in Solo, Phoenix in Lasem, and Ranrang in Bali.

The diversity also occurs in objects that are used as batik pattern. Some batik patterns use natural phenomena, such as cloud in Megamendung pattern or rain in Danliris pattern. Other batik patterns use animals as main object such as chicken, peacock, or phoenix. But, most of batik patterns use plant as its main object, such as: flower, leaf, or branch.

One popular plant that is used in traditional batik pattern is papaya leaf. Papaya leaf is adopted because of its special finger like morphology. Even this plant is less beautiful than flower object, such as orchid, its leaf characteristic is very unique compared with the other plants. But, in many batik patterns that use papaya leaf as its basic pattern, the plant part that is adopted is only the leaf. The other papaya plant parts, such as fruit, branch, or stem are not adopted because they are very common.

One popular plant that has similar morphology with papaya is Cnidoscolus aconitifolius [6-9]. This plant leaf is very similar to the papaya leaf with smaller size. Some people name this plant as Japanese papaya while other people name this plant as tree spinach [6, 7].

Basically, this plant is not papaya either spinach. Unfortunately, this plant has not been adopted as batik pattern yet. So, developing Cnidoscolus aconitifolius model as batik pattern is challenging.

In common ways, many batik patterns are developed manually and traditionally. This manual process makes the development is slow. In the other side, computation technology can be used to develop batik pattern faster and more various. There are researches in computation area that focus on batik pattern. Some researches focus on batik pattern development while the others focus on batik pattern recognition. One popular method that is used in batik pattern development is fractal method [10]. Not only batik, fractal is popular tool to develop many natural objects [11, 12].

Based on this opportunity, this research main goal is developing new Cnidoscolus aconitifolius leaf model as batik pattern computationally. Even the main object is the leaf, the branch is also developed. So, this plant model can be differentiated from the papaya model. This model is developed by using random walk method. Random walk is chosen because of its stochastic characteristic so that the leaves that are developed are different to each other even they are similar. Besides random walk method, Lindenmayer system (L-system) method is also used as branching method. L-system is very popular in developing plant or plant like model. So, the research question is how to use these methods as basis to develop Cnidoscolus aconitifolius leaf model as batik pattern

This research is the continuation of the previous researches in computational based batik pattern modeling [13-16]. In the previous works, the L-system [13,16] and random walk [14] methods have been used too. The other



methods that are used in our previous works are social forces model and cellular automata [14, 15].

This paper is organized as follows. In the first section, we describe the background, research question, research purpose, and the paper organization. In the second section, we explain the basic characteristic and morphology of *Cnidoscolus aconitifolius* plant. In the third section, we describe our proposed model. In the fourth section, we describe the implementation of the model in batik pattern generation application. In the fifth section, we discuss the test result analyzes and the research findings. In the sixth section, we make conclusion and future research potentials.

CNIDOSCOLUS ACONITIFOLIUS

Cnidoscolus aconitifolius plant is a tropical tree. It has some other common names, such as: Chaya, tread softly, or tree spinach [8]. English calls it cabbage star while French calls it Manioc batard [7]. Its other scientific names are *Jatropha aconitifolia*, *Jatropha napaeifolia*, and *Cnidoscolus napaeifolius* [7]. Its family is Euphorbiaceae [7, 8]. This plant possibly comes from Yucatan region in Mexico [7]. Now, this plant is also widespread in eastern Mexico and Central America. It is also cultivated in Ghana, Nigeria, and Pasific [7].

Cnidoscolus aconitifolius is an evergreen plant [6]. It can grow as a monoecious shrub or small tree [6]. It can grow up to 5 or 6 meter tall [8]. It contains white latex with thick pale trunk [6]. It is usually armed with stinging hairs. Most of this plant is branched [8]. This plant is suitable for sandy, loamy, or clay soils but the preference is well drained soil [8]. This plant also can grow in nutritionally poor soil [8]. This plant morphology is shown in Figure-1. Figure-1a is the appearance of the *Cnidoscolus aconitifolius* plant while Figure-1b is the detailed appearance of the *Cnidoscolus aconitifolius* leaves.

Based on its cultivation methods, there are extensive morphological variations within its species. Ross-Ibara, *et al* identified four cultivated variety of Chaya. These variations are: Estrella, Picuda, Chayamansa, and Redonda [9].

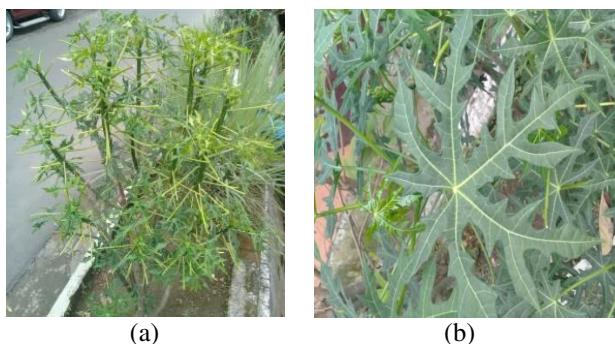


Figure 1. Plant morphology.

The morphology variations of the leaf are as follows [9]. Estrella leaf has five spreading lobes [9]. These lobes are not overlapping dentate. Picuda variant

leaf contains five to nine lobes [9]. Picuda leaf lobe is narrower than Estrella lobe. Chayamansa leaf commonly contains five lobes [9]. The three central lobes are overlapping. In these three variants, the lobe contains stinging hair. Redonda variant leaf contains three lobes without stinging hair [9]. The visualization of these leaves morphology is shown in Figure-2.

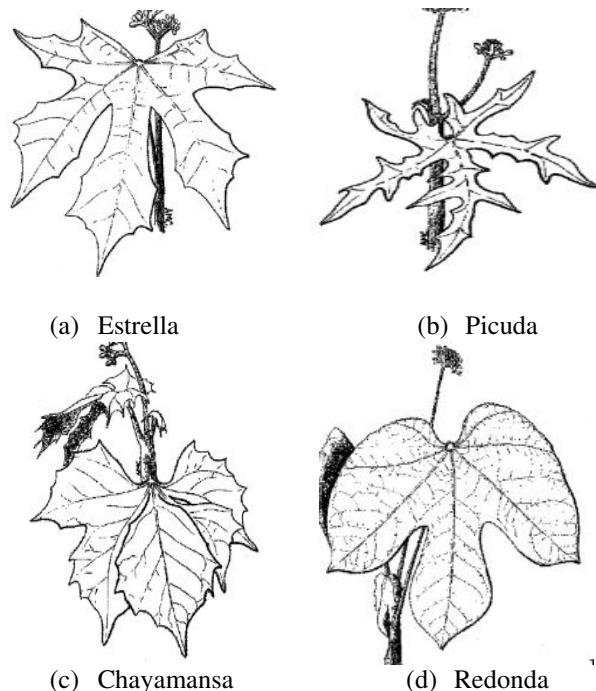
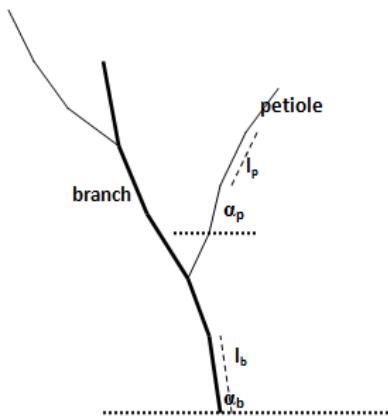


Figure 2. Leaf variation.

PROPOSED MODEL

Based on the morphology of *Cnidoscolus aconitifolius* plant, in this paper, we propose the random walk based *Cnidoscolus aconitifolius* model. The model is divided into two parts. The first part is developing the branch and petiole model. The second part is developing the leaf model. In this paper, we develop the Estrella and Picuda variations.

In the branch and petiole model, these parts are built as an array of segments. The segments grow with certain length and direction. The segment length is static. Meanwhile, the direction of the segment varies and the value is determined by using random walk method. The illustration is shown in Figure 3.

**Figure-3.** Branch and petiole illustration.

There are some variables that are used in developing branch model. Variable l_b represents the branch segment length. Variable α_b represents the branch angle relative to the horizontal axis. Variables x_{b1} and y_{b1} represent the starting position of the branch segment. Variables x_{b2} and y_{b2} represent the end position of the branch segment. At the beginning, $y_{b1,0}$ is set 0. The α_b is set 90 degrees. For single run, the image contains several branches that grow vertically. The $x_{b1,0}$ is in any value in the horizontal axis with fixed 500 pixels distance between sequenced branches. During the branch growth, some variables value is set and their value is determined by using Equation 1 to Equation 6.

$$x_{b2,n} = x_{b1,n} + l_b \cos(\alpha_b) \quad (1)$$

$$y_{b2,n} = y_{b1,n} + l_b \sin(\alpha_b) \quad (2)$$

$$\alpha_{b,n+1} = \alpha_{b,n} + \text{rand}(-\Delta_b, \Delta_b) \quad (3)$$

$$\alpha_{b,n+1} = \begin{cases} \alpha_{bmax}, & \alpha_{b,n+1} > \alpha_{bmax} \\ \alpha_{bmin}, & \alpha_{b,n+1} < \alpha_{bmin} \\ \alpha_{b,n+1}, & \text{else} \end{cases} \quad (4)$$

$$x_{b1,n+1} = x_{b2,n} \quad (5)$$

$$y_{b1,n+1} = y_{b2,n} \quad (6)$$

The explanation of these equations is as follows. In equation 1 and 2, the end position of the branch segment is determined based on the values of the starting position of the segment, the segment length, and the branch segment angle. After the end position of the branch segment is determined, the next process is determining the next branch angle. The next branch angle value is determined by using random walk method as it is shown in Equation 3, which is depended on the current branch angle and IID random number that acts as branch deviation angle (Δ_b). Based on Equation 3, the next branch direction can be left or right of the current branch direction. Equation 4 acts as branch angle limiter so that the branch direction is between its minimum and maximum ranges.

Based on Equation 5 and Equation 6, the end position of the current branch segment will become the starting position of the next branch segment.

The branch segment does not always produce leaf petiole. Different to the previous similar works that the leaf growth is determined by the leaf growth probabilistic [13,16], in this model, the possibility of the petiole is determined by the inter petiole distance. The inter petiole distance is represented by using variable d_{intpet} . The unit of this variable is segment. For example, if the value of the d_{intpet} is 5, it means that the petiole will grow at the next five branch segments. The value of this variable is not static. This variable value is always recalculated after a new petiole is created. This inter petiole distance value is determined by using Equation 7. In this equation, the result value is determined randomly which the random value follows uniform distribution that ranges from its minimal value (d_{min}) to its maximum value (d_{max}).

$$d_{intpet} = \text{rand}(d_{min}, d_{max}) \quad (7)$$

When it is the time for branch to shoot new petiole, the petiole grows from the branch. The starting position of the starting petiole is at the end position of the segment. This process is determined by using Equation 8 and Equation 9. Meanwhile the petiole starting angle is determined by using Equation 10.

$$x_{p1,0} = x_{b2,n} \quad (8)$$

$$y_{p1,0} = y_{b2,n} \quad (9)$$

$$\alpha_{p,0} = \begin{cases} \alpha_{b,n+1} - \Delta_{startpet}, & \text{rand}(0,100) < 50 \\ \alpha_{b,n+1} + \Delta_{startpet}, & \text{rand}(0,100) \geq 50 \end{cases} \quad (10)$$

The explanation of Equation 8 to Equation 10 is as follows. In Equation 8, the end horizontal position of the branch will become the initial horizontal starting position of the petiole. In Equation 9, the end vertical position of the branch will become the initial vertical starting point of the petiole. In Equation 10, it is shown that the initial petiole angle can be on the left or the right side of the branch. This deviation angle is represented by variable $\Delta_{startpet}$. This possibility is depended on of the random value that follows uniform distribution.

After the initialization of the petiole, the next step is the petiole growth sub model. Similar to the branch growth, the petiole growth process uses the random walk method. The petiole is an array of petiole segments. Each segment has starting point, end point, and segment angle. The value of these variables is determined by using Equation 11 to Equation 15.

$$x_{p2,n} = x_{p1,n} + l_p \cos(\alpha_p) \quad (11)$$

$$y_{p2,n} = y_{p1,n} + l_p \sin(\alpha_p) \quad (12)$$

$$\alpha_{p,n+1} = \alpha_{p,n} + \text{rand}(-\Delta_p, \Delta_p) \quad (13)$$



$$x_{p1,n+1} = x_{p2,n} \quad (14)$$

$$y_{p1,n+1} = y_{p2,n} \quad (15)$$

The explanation of these equations is as follows. As it is shown in Equation 11 and Equation 12, the end position of the current petiole segment is depended on the its starting petiole segment, the petiole angle, and the petiole segment length. The direction of the petiole growth uses random walk method by using petiole angle (α_p). As it is shown in Equation 13, the next petiole angle value is depended on its current value and the random number that represents the petiole angle deviation (Δ_p). Different to the branch growth, in this petiole growth, the angle is not limited by the minimum and maximum values. As it is mentioned in Equation 14 and Equation 15, the end position of the current petiole segment will become the starting position of the next petiole segment.

After the petiole does not grow anymore, then the leaf is generated by using the leaf model. The originating position of the leaf is at the end position of the last petiole segment. The direction of the leaf lobes is determined by the leaf lobe angle (α_{lobe}). These values are determined using Equation 16 to Equation 18. The illustration of the leaf is shown in Figure-4.

$$c_x = x_{p2,n} \quad (16)$$

$$c_y = y_{p2,n} \quad (17)$$

$$\alpha_{lobe,0} = \alpha_{p,n+1} - 90 \quad (18)$$

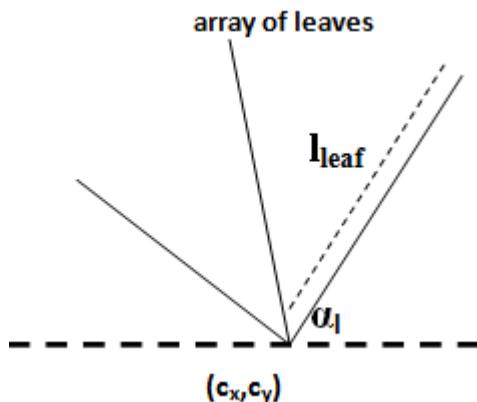


Figure-4. Leaf lobes illustration.

As it is mentioned in the literature about *Cnidoscolus aconitifolius* leaf morphology, the leaf contains five lobes. So, there will be five iterations to make the leaf lobes for each leaf. Each lobe length (l_{leaf}) or l_{lobe} may be different among the lobes. Equation 19 is used to determine the lobe length. Meanwhile, after one lobe is created, the leaf angle will be changed so that the next lobe direction will be different to the current lobe. The next lobe angle value is determined by using Equation 20.

$$l_{lobe,n} = \text{random}(l_{minlobe}, l_{maxlobe}) \quad (19)$$

$$\alpha_{lobe,n+1} = \alpha_{lobe,n} + \text{rand}(\Delta_{minlobe}, \Delta_{maxlobe}) \quad (20)$$

The explanation of these equations is as follows. In Equation 19, it is shown that the lobe length is determined randomly. This random process follows uniform distribution. The lobe length ranges from its minimum length ($l_{minlobe}$) to its maximum length ($l_{maxlobe}$). In Equation 20, the next lobe angle is determined by using random walk method. This value is depended on its current angle and the IID random number that ranges from its minimum deviation angle ($\Delta_{minlobe}$) to its maximum deviation angle ($\Delta_{maxlobe}$).

After the lobe is created, the next process is creating the sting hairs beside the lobe. To simplify the process, this sting hair or finger development model is divided into three processes. In the first process, the sting hair is generated with the assumption that the lobe is stretched from coordinate zero (0, 0) and it is stretched horizontally. So, the lobe angle is assumed zero. After the finger is created, then the finger is rotated relative to the lobe angle. After the finger is rotated, the last process is transforming the finger relative to the center point of the leaf (c_x, c_y). The illustration of the lobe sting hair development that the assumption is it starts from coordinate zero and the lobe angle is zero is shown in Figure-5.

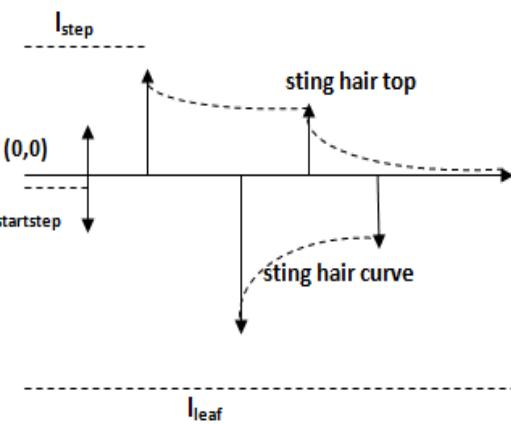


Figure-5. Sting hair illustration.

There are several terms that are used in sting hair generation model. The step is the length of the sting hair relative to the lobe starting position. The sting hair top is the position of the end of the sting hair. The sting hair curve is the curve that connects the sequenced sting hair. The start step is the beginning step where the first sting hair grows. So, the first sting hair step horizontal position is determined by using Equation 21. Meanwhile, the sting hair step angle is determined manually.

$$l_{step,0} = l_{startstep} \quad (21)$$

After the initial sting hair step is determined, the next process is iterating as long as the step length is still



below the lobe length. Equation 22 to Equation 25 is used to determine the next step length and next step angle.

$$l_{step,n+1} = \begin{cases} l_{lobe}, (l_{lobe} - l_{step,n}) < (\omega_{lobe} l_{lobe}) \\ l_{step,n} + rand(l_{minstep}, l_{maxstep}), else \end{cases} \quad (22)$$

$$l_{step,n+1} = \begin{cases} l_{step,n+1}, l_{step,n+1} \leq l_{lobe} \\ l_{lobe}, else \end{cases} \quad (23)$$

$$\alpha_{step,n+1} = rand(\Delta_{minstep}, \Delta_{maxstep}) \quad (24)$$

$$\alpha_{step,n+1} = \begin{cases} \alpha_{step,n+1}, l_{step,n+1} < l_{lobe} \\ 0, else \end{cases} \quad (25)$$

Based on Equation 22, it is shown that the next step length is generated by using random walk method. The next step length is depended on the current step length and the IID random number that follows uniform distribution. Meanwhile, the next step angle is generated randomly between its minimum and maximum step angle and follows uniform distribution.

After the sting hair top position is generated, the next process is generating the sting hair curve. The sting hair curve is an array of dots between sequenced sting hairs. So, at the beginning, the starting point and the end point of the curve must be determined. The starting and end points of the curve is determined by using Equation 26 to Equation 31.

$$r_{startcurve} = \frac{l_{step,n}}{\cos(\alpha_{step,n})} \quad (26)$$

$$x_{startcurve} = l_{step,n} \quad (27)$$

$$y_{startcurve} = r_{startcurve} \cdot \sin(\alpha_{step,n}) \quad (28)$$

$$r_{endcurve} = \frac{l_{step,n+1}}{\cos(\alpha_{step,n+1})} \quad (29)$$

$$x_{endcurve} = l_{step,n+1} \quad (30)$$

$$y_{endcurve} = r_{endcurve} \cdot \sin(\alpha_{step,n+1}) \quad (31)$$

After the starting and end points of the curve are determined, the next step is placing nodes that shape the curve. The horizontal position of the node is represented in x_{node} . These nodes are sequenced along the lobe with fixed distance between sequenced nodes. So the next process is determined the vertical coordinate of the node (y_{node}). This value is determined by using Equation 32 to Equation 35. In Equation 35, the α_{curve} is represented in degree.

$$y_{node} = y_{straight} - y_{red} \quad (32)$$

$$y_{straight} = x_{startcurve} + \frac{(x_{node} - x_{startcurve})(y_{endcurve} - y_{startcurve})}{x_{endcurve} - x_{startcurve}}. \quad (33)$$

$$y_{red} = \omega_{red} \cdot y_{straight} \cdot \sin(\alpha_{curve}) \quad (34)$$

$$\alpha_{curve} = \frac{(x_{node} - x_{startcurve})}{x_{endcurve} - x_{startcurve}} \cdot 180 \quad (35)$$

After the node coordinate is determined, the next process is rotating the the node relative to the lobe angle. This process must be done because the current node position is relative to angle step zero. After the node is rotated, then the node coordinate is recalculated. This transformation process is done by using Equation 36 to Equation 40.

$$r_{node} = \sqrt{x_{node}^2 + y_{node}^2} \quad (36)$$

$$\alpha_{node} = \tan^{-1}\left(\frac{y_{node}}{x_{node}}\right) \quad (37)$$

$$\alpha_{rot} = \begin{cases} \alpha_{lobe} + \alpha_{node}, first quadrant \\ \alpha_{lobe} - \alpha_{node}, fourth quadrant \end{cases} \quad (38)$$

$$x_{rot} = r_{node} \cdot \cos(\alpha_{rot}) \quad (39)$$

$$y_{rot} = r_{node} \cdot \sin(\alpha_{rot}) \quad (40)$$

The explanation of these equations is as follows. At the beginning, the Euclidean distance between the node and the lobe starting point is determined by using Equation 36. Then, the node angle is determined by using Equation 37. Then, the rotation process is done. There are two curve groups in each lobe. The first group is on the left of the lobe while the second group is on the right of the lobe. Because the assumption is that the lobe is on the horizontal axis, so the groups are in the first quadrant and the fourth quadrant. The rotated angle is determined by using Equation 38 and its value is depended on which quadrant that the node exists. After the rotated angle is determined, the last step is determined the rotated position of the node. This step is determined by using Equation 39 and Equation 40.

After the rotating process is done, the final process is transforming the rotated node relative to the starting point of the lobe (c_x, c_y). This process is done because the rotated node position is still relative to coordinate zero. This process is determined by using Equation 41 and Equation 42.

$$x_{final} = c_x + x_{rot} \quad (41)$$

$$y_{final} = c_y + y_{rot} \quad (42)$$

IMPLEMENTATION

Based on this proposed model, then the Cnidoscolus aconitifolius batik pattern is generated computationally. The batik pattern application is developed by using PHP language so that it is a web based application. The result is the batik pattern image in JPEG format. The image size is 2,000 x 2,000 pixels.

In this application, the Cnidoscolus aconitifolius based batik pattern is combined with the traditional Kawung pattern. The kawung pattern acts as background



pattern while the *Cnidoscolus aconitifolius* pattern acts as foreground pattern. The result is shown in Figure-6.

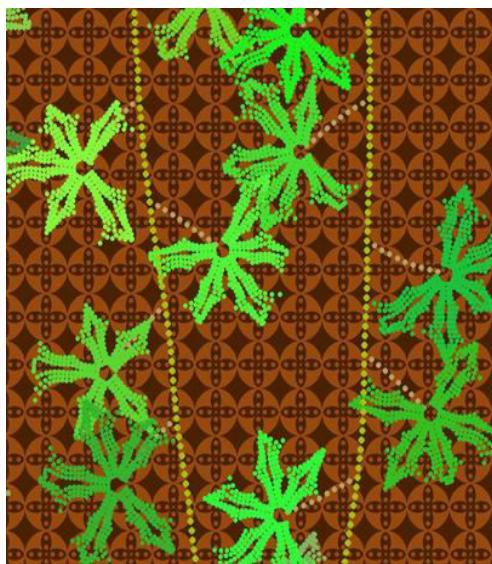


Figure-6. *Cnidoscolus aconitifolius* Batik pattern.

Based on Figure-6, it is shown that the *Cnidoscolus aconitifolius* batik pattern has been successfully generated by using the application. The branch is represented in array of yellow dots. The petiole is represented in array of beige dots. The leaf is represented in array of green dots. In Figure 6, it is shown that the petiole and branch colors are static. Meanwhile, there is variation of the green level among the leaves. This variation occurs by generating randomized green color each time a leaf is generated. The leaf color randomizing method follows uniform distribution.

DISCUSSIONS

After the application can generate *Cnidoscolus aconitifolius* batik pattern successfully, the next section is the test, result analyzes, and the discussion. The several tests are done to observe the relation between the adjusted variables with the result image. So, these tests are observed visually.

The first test is observing the relation between the inter petiole distance with the image result. In this test, the inter petiole distance is set low, medium, and high. The low inter petiole distance means that the inter petiole distance is set from 8 to 12. The medium inter petiole distance means that the inter petiole distance is set from 18 to 22. The high inter petiole distance means that the inter petiole distance is set from 28 to 32. The result is shown in Figure-7. The low, medium, and high inter petiole distance result images are shown in Figure 7 consecutively.

Based on the result in Image 7, it is shown that the inter petiole distance has opposite relation with the number of leaves. Comparing among images, the increasing of the inter petiole distance reduces the number of leaves that are generated in the result image.

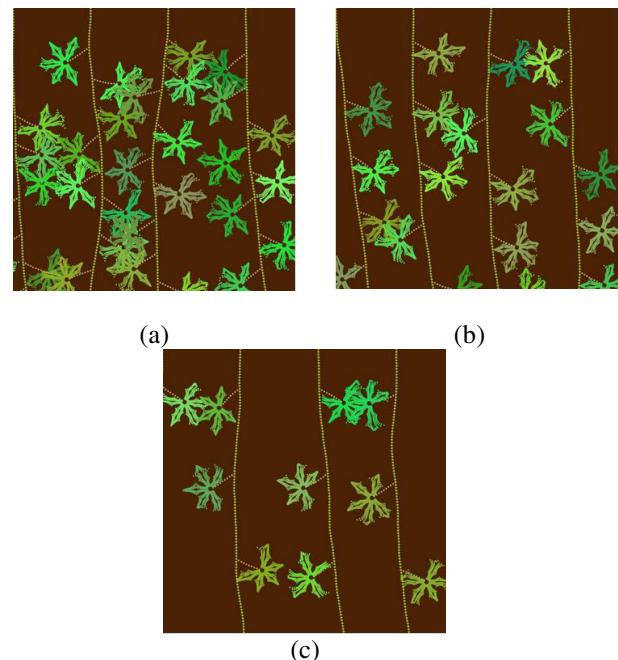


Figure-7. Low, medium, and high inter petiole distance results.

The second test is observing the relation between lobe angle deviation and the leaf morphology. In this test, the lobe angle deviation is set as static value. This variable values are 20, 40, 60, and 80 degrees. The result is shown in Figure-8. In Figure-8, the result represents the lobe angle deviation from 20 to 80 degrees consecutively.

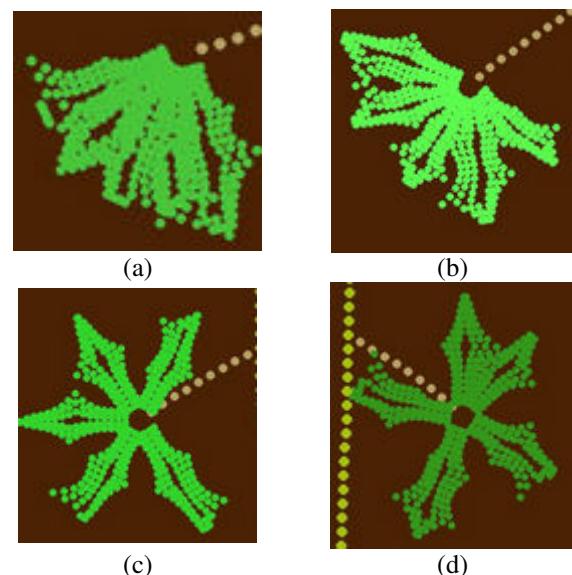


Figure-8. Leaf morphology variation based on the lobe angle variation

Based on the result images in Figure-8, it is shown that the lobe deviation angle has positive correlation with the leaf circular width. When the lobe deviation angle is small, the leaf circular width is narrow as it is shown in Figure-8a. When the lobe deviation angle increases, the leaf circular width goes wider too. High lobe deviation angle triggers



the opportunity that the last lobe will overlap the first lobe as it is shown in Figure-8d.

The third test is observing the relation between the sting hair step angle deviation and the leaf morphology. In this test, the angle deviation is set static. The angle deviations are 10, 20, 30, and 40 degrees. The result is shown in Figure-9 consecutively.

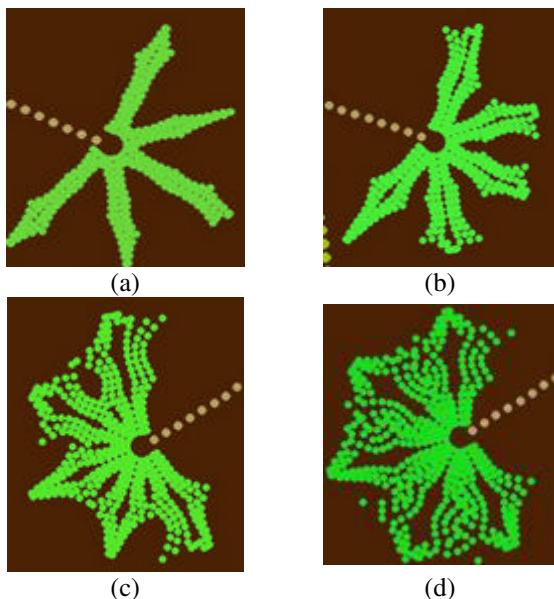


Figure-9. Leaf morphology variation based on the step angle variation.

Based on the result images in Figure 9, the sting hair step angle has positive relation with the lobe width. Lower sting hair step angle will produce narrower lobe width. Meanwhile, higher sting hair step angle will produce wider lobe width. But, high sting hair step angle can make lobe will overlap its neighbor lobe. Based on these images, small value of sting hair step angle deviation makes the Picuda like leaf. Meanwhile, the medium value of sting hair step angle deviation makes the Estrella like leaf.

CONCLUSIONS AND FUTURE WORK

Based on the explanation in this paper, it is concluded that the proposed *Cnidoscolus aconitifolius* model has been developed by using random walk method. This proposed model also has been implemented into web based batik pattern generation application. This application also generates *Cnidoscolus aconitifolius* batik pattern successfully. In this application, the *Cnidoscolus aconitifolius* model is combined with the traditional Kawung pattern.

Based on several tests that have been done, there are several research findings. The inter petiole distance has negative correlation with the number of generated leaves. The lobe angle deviation has positive correlation with the leaf circular width. But, higher lobe deviation angle may trigger result that the last lobe will overlap the first lobe.

The sting hair step angle deviation has positive correlation with the lobe width. The Picuda like leaf can

be generated by using small value of the sting hair step angle deviation. The Estrella like leaf can be generated by using medium value of the sting hair step angle deviation. Meanwhile, higher sting hair step angle deviation may trigger condition where a lobe will overlap its neighbor lobe.

There are still many research potentials in computational based batik pattern generation. Besides *Cnidoscolus aconitifolius* leaf, there are many leaves that have unique morphology so that it can be modeled as an artistic object. Especially in batik pattern, the research in leaf model can also enrich the plant based batik pattern so that the plant based batik pattern is not only flower. Leaf can be as beautiful as flower.

REFERENCES

- [1] E. Maulia. 2019. Batik Selected for UNESCO Cultural Heritage List. Jakarta Post. September 8th. <http://www.thejakartapost.com/news/2009/09/08/batik-selected-unesco-cultural-heritage-list.html>.
- [2] S. Schonhardt. 2009. Battle for Batik. CNN. October 14th. 2009. <http://travel.cnn.com/explorations/shop/battle-batik-621293/>.
- [3] P. Gelling. 2009. Score One for Indonesia in The War Over Batik. The New York Times. September 14th. <https://www.nytimes.com/2009/09/15/world/asia/15iht-batik.html>.
- [4] T.S. Barley. 2009. Indonesia and Malaysia Battle Over Batik. Jakarta Globe. October 1st. <http://jakartaglobe.id/archive/indonesia-and-malaysia-battle-over-batik/>.
- [5] J.E. Tandjung. 2015. Can We Protect Mega Mendung Batik?. The Jakarta Post. January 26th. <http://www.thejakartapost.com/news/2015/01/26/can-we-protect-mega-mendung-batik.html>.
- [6] *Cnidoscolus aconitifolius*. Useful Tropical Plants. <http://tropical.theferns.info/viewtropical.php?id=Cnidoscolus+aconitifolius>.
- [7] *Cnidoscolus aconitifolius* (Mill.) I.M. Johnstone. PROTA4U. [https://www.prota4u.org/database/protav8.asp?g=pe&p=Cnidoscolus+aconitifolius+\(Mill.\)+I.M.Johnstone](https://www.prota4u.org/database/protav8.asp?g=pe&p=Cnidoscolus+aconitifolius+(Mill.)+I.M.Johnstone).
- [8] *Cnidoscolus aconitifolius* (Mill.) I.M. Johnst. Plants for a Future. <https://www.pfaf.org/user/Plant.aspx?LatinName=Cnidoscolus+aconitifolius>.



- [9] J. Ross-Ibarra and A. Molina-Cruz. 2002. The Ethnobotany of Chaya (*Cnidoscolus aconitifolius* ssp. *Aconitifolius* Breckon): A Nutritious Maya Vegetable. *Economic Botany*. 56(4): 350-365.
- [10] N. Margried. 2015. Batik Fractal Community: Creative Engagement through Technology. *Procedia - Social and Behavioral Sciences*. 184: 214-222.
- [11] S. Banerjee. 2009. Nature's Geometry. *Breakthrough*. 13(4).
- [12] B.B. Mandelbrot. 1977. *The Fractal Geometry of Nature*. W.H. Freeman and Company. New York.
- [13] P.D. Kusuma. 2017. Fibrous Root Model in Batik Pattern Generation. *Journal of Theoretical and Applied Information Technology*. 95(14).
- [14] P.D. Kusuma. 2017. Interaction Forces-Random Walk Model in Traditional Pattern Generation. *Journal of Theoretical and Applied Information Technology*. 95(14).
- [15] P.D. Kusuma. 2016. Implementation of Pedestrian Dynamic in Cellular Automata Based Pattern Generation. *International Journal of Advanced Computer Science and Applications (IJACSA)*. 7(3): 65-70.
- [16] P.D. Kusuma. 2017. Simplified Coral Modeling in Batik Pattern Generation. *Journal of Theoretical and Applied Information Technology*. 96(10): 3102-3114.