

Dendrobium Modeling in Batik Pattern Generation by using Random Walk and Lindenmayer System

Purba Daru Kusuma
School of Electrical Engineering, Telkom University, Bandung, Indonesia

Abstract: Batik is one of cultural heritage in Indonesia and it consists of various patterns and objects. Most of these objects are plant. By conventional way, batik pattern is generated manually. So, the batik design generating rate is slow. By computational technology, batik pattern can be generated faster and more various. In this research, we propose the new and simpler Dendrobium modeling, so that, it can be generated as batik pattern. Dendrobium is one of orchid plant genus with huge number of species. In this research, we use random walk method as a stochastic approach, so that, there is uncertainty pattern and path in the result. We model all parts of the orchid plant includes: root, stem, leaf, pedicel, peduncle and flower. By using this proposed model, the Dendrobium based batik pattern is successfully generated. In this research, traditional Kawung pattern is used as background pattern. In this research, there are three tests to evaluate the relation between the adjusted variables and the observed variables. In the first test, we observe the relation between the number of stem segments and the number of leaves. In the second test, we observe the relation between the flowering ratio and the number of peduncles. In the third test, we observe the relation between the number of peduncle segments and the number of flowers. Based on the tests in all tests, the relation between the adjusted variable and the observed variable is positive linear.

Key words: Batik, dendrobium, orchid, random walk, generated, flowering

INTRODUCTION

Batik is one of national cultural heritage in Indonesia. It is used, since, pre colonial era until today. At the first time, many batik patterns are used only by royal families. After the reformation by Hamengkubuwono IX as the King of Yogyakarta, all of monarch batik patterns can be used by common people. In Indonesia, there are some cities that have their own trademark batik pattern such as Cirebon, Yogya, Solo, Madura, Pekalongan and Lasem. Each of these cities has its own batik pattern characteristics. Because of its supreme value, batik is admitted as one of world cultural heritage by UNESCO (Maulia, 2009). This international recognition has terminated the dispute between Indonesia and Malaysia about batik claiming (Gelling, 2009).

In conventional way, batik pattern is generated manually. It means that the designer draws the concept and the design and then the design will be implemented into batik fabric by painted (batik tulis), stamped (batik cap) or printed (Florek, 2011). The consequence is batik pattern production rate is slow. The other problem is many batik pattern designer is low educated people. So, even there are lots of batik patterns, the number of the batik objects is limited and the object repetition rate is

very high. For example, even there many flower types, the flower based batik patterns are seemed similar to each other with small modification. Some innovations are proposed by modern batik company such as Batik Keris or Danar Hadi. These companies have enough budget to hire high educated designer, so that, they can produce fresher or more contemporary batik pattern, so that, younger generation accepts these patterns. Unfortunately, even the design process in this company is aided by computer, the computational technology adoption is still low.

In the other side, the computational technology can generate batik pattern automatically. One of the common methods is fractal method. Yulianto *et al.* (2012) used fractal based and normal Gaussian method in batik coloring (Yulianto *et al.*, 2012). Besides batik, fractal method is also used in generating many types of artistic natural pattern (Pickover, 1995). In the previous works, we have adopted interaction force (Kusuma, 2017a), Lindenmayer system (Kusuma, 2017b) and crack dynamic (Kusuma, 2017c) to generate batik pattern computationally. In these researches, batik pattern has been successfully generated fast. In computational way, by adjusting the parameters, the result can be modified and regulated.

The most common batik objects are plant object. Most of plant objects that are used in batik pattern are flower plant, especially, orchid. Orchid is used commonly because of its beauty and its special characteristic that there are many flowers are in one orchid peduncle. The other fact is there are approximately 5,000 orchid species in Indonesia. Unfortunately, the orchid based batik patterns are seemed similar to each other even there are many genus in Orchidaceae family and each genus has its own characteristics.

In the other side, research in orchid modeling is very limited. One research about Dendrobium modeling was not to create the plant visually but to predict the time of its growing phase to give better prediction for orchid farmer (Kongsombut and Chairsricharoen, 2014). The other research developed 3D based orchid model by reconstructing it from real orchid image and developing it by using open GL programming (Li *et al.*, 2017). The other research developed the orchid flower part only by using sketch based modeling (McCord *et al.*, 2008). So, the question is how to develop a whole part of orchid based batik pattern computationally and specifically.

The purpose of this research is developing new Dendrobium Model, so that, it can be implemented in batik pattern generation. Dendrobium is one of orchid genus. It consists of huge number of species. In this research, we adopt random walk and L-system method (Prusinkiewicz and Lindenmayer, 1990) as the basis method. The reason is the L-system has been used in many plant modeling (Boudon *et al.*, 2012; Guan, 2008; Sun *et al.*, 2009). We model all parts of the plant includes root, stem, leaf, pedicel and flower.

Dendrobium morphology: Now a days, there are approximately 20, 000-30, 000 orchid species in the world. One of the orchid genus is Dendrobium. Dendrobium comes from “dendros” and “bios”. Dendros means tree and bios means life. So, Dendrobium can be interpreted as orchid that lives in the living tree. This genus consists of approximately 1,400 species. This genus is spread in Japan, China, Malaka, Indonesia, Papua and Australia (Parnata, 2005). The example of Dendrobium plant is shown in Fig. 1 (Gallis, 2010).

Dendrobium is classified into sympodial orchid. It means that rather than grows vertically as in monopodial orchid, it grows laterally and produces a new shoot along rhizome that develops into a stem with roots and leaves and eventually produces flowers. It also has pseudobulb: a thickened stem.



Fig. 1: Dendrobium (Gallis, 2010)

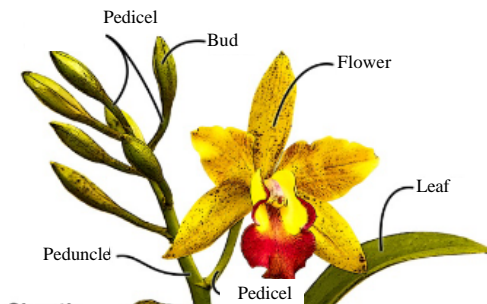


Fig. 2: Orchid flower morphology (S. bottom)

The flower morphology of Dendrobium or general orchid is as follows. Peduncle is the stalk that bears flowers and it connects the flowers group to the stem. Pedicel is small pseudo stalk that connect the flower to the peduncle. The flower bud that is the compact knob like that grows from the peduncle and it can become flower. The flower morphology is shown in Fig. 2 (S. bottom).

The orchid flowers have important parts (S. bottom). Sepal is the outer part and each flower consists of three sepals. The flower also consists of two petals in the side of the flower. It also has lip or labellum at the base of the flower. This detailed part of the flower is shown in Fig. 3 (S. bottom).

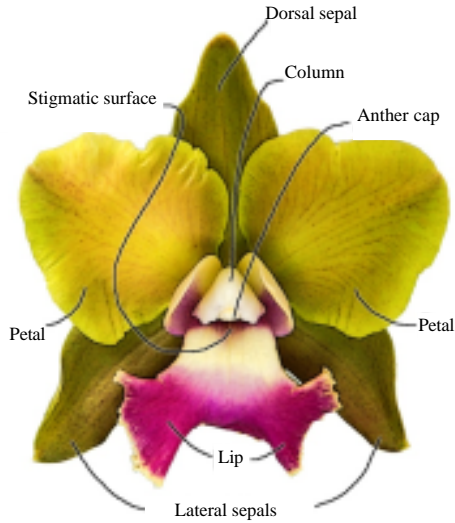


Fig. 3: Detail orchid flower morphology (S. bottom)

MATERIALS AND METHODS

Proposed model: In this research, we propose the all parts of orchid model. It includes: root, stem, leaf, peduncle, pedicel, flower and flower bud. In this model, the upper part and the root are generated separately. So, the model can generate the root first or the upper part first. This two part is connected by one central starting point c in $\langle c_x, c_y \rangle$ tuple. Index x is for horizontal coordinate and index y is for vertical coordinate.

In the starting process of the upper part, some parameters are used. Variable n_s is used to determine the number of the stems. The value of n_s is determined by using (Eq. 1). In Eq. 1, n_{smin} is the minimum number of stems while n_{smax} is the maximum number of stems. The r_{no} is the probability that the stem will shoot peduncle. After the setup process of the upper part parameters is done, the next process is upper part growth. The general upper part growing process algorithm is shown in algorithm 1. Algorithm 1, i_s is the stem index and $grow_stem()$ is the procedure to grow the stem:

$$n_s = \text{random}(n_{smin}, n_{smax}) \quad (1)$$

Algorithm 1: General upper part growing algorithm:

```

begin
   $i_s \leftarrow 1$ 
  While  $i_s \leq n_s$ 
    begin
       $grow\_stem()$ 
       $i_s++$ 
    end
  end
end

```

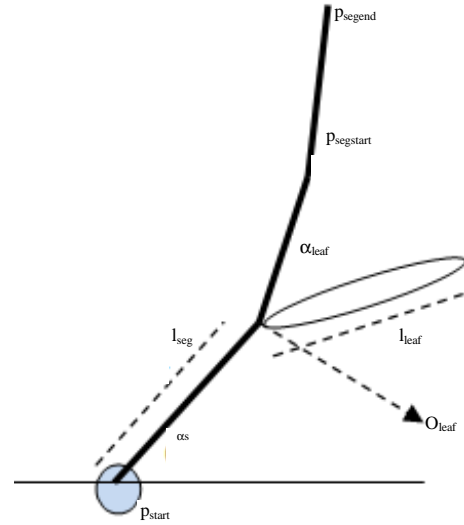


Fig. 4: Stem structure illustration

In the $grow_stem()$ procedure, some variables are set up. These variables are the stem segment length (l_{seg}), the stem angle (α_s), the position of the starting stem segment ($p_{segstart}$), the number of segments (n_{seg}), the starting segment that produces leaf (s_{leaf}), the leaf length (l_{leaf}), the leaf angle (α_{leaf}), the leaf length reduction rate ($r_{leafreduc}$) and the leaf orientation (o_{leaf}). These variables value is determined by using (Eq. 2-8). The stem structure is shown in Fig. 4:

$$l_{seg} = \text{random}(l_{segmin}, l_{segmax}) \quad (2)$$

$$\alpha_s = \text{random}(\alpha_{smin}, \alpha_{smax}) \quad (3)$$

$$p_{segstart} = c \quad (4)$$

$$n_{seg} = \text{random}(n_{segmin}, n_{segmax}) \quad (5)$$

$$s_{leaf} = \text{random}(1, n_{seg}) \quad (6)$$

$$l_{leaf} = \text{random}(l_{leafmin}, l_{leafmax}) \quad (7)$$

$$o_{leaf} = \begin{cases} 1, \text{random}(0, 100) \leq 50 \\ -1, \text{random}(0, 100) > 50 \end{cases} \quad (8)$$

There are several new variables that are used in Eq. 2-8. The l_{segmin} is the minimum stem segment length while the l_{segmax} is the maximum stem segment length. The α_{smin} is the minimum stem angle while the α_{smax} is the maximum

stem angle. The n_{segmin} is the minimum number of stem segments while the n_{segmax} is the maximum number of stem segments. The $l_{leafmin}$ is the minimum leaf length while the $l_{leafmax}$ is the maximum leaf length.

By using these variables, then the stem growth can start. The stem grows stochastically by using random walk method. By using random walk, the stem direction is not deterministic. The stem growth algorithm is shown in algorithm 2.

Algorithm 2; Stem growth algorithm:

```

begin
  iseg ← 1
  while iseg ≤ nseg
  begin
    psegend, x ← psegstart, x + lseg * COS (αseg)
    psegend, y ← psegstart, y + lseg * SIN (αseg)
    Createleaf (psegstart, psegend)
    If lseg > sleaf then
      createleaf ()
      αseg ← αseg + random (-Δseg, Δseg)
      psegstart, x ← psegend, x
      psegstart, y ← psegend, y
      lseg ++
    end
  end
end

```

In algorithm 2, there are new variables and procedure. The i_{seg} represents the stem segment index. The $p_{segstart}$ is the segment starting point. The p_{segend} is the segment end point. The Δ_{seg} is the deviation angle of the next stem segment direction. Based on this algorithm, the segment will grow until the number of stem segments is reached. After the new segment is created, the end point of the current segment will become the starting point of the next segment. The create leaf () procedure is used to initiate the segment leaf creation. In the leaf creation, the leaf starting point ($p_{leafstart}$) is the $p_{segstart}$. The leaf end point is determined by using Eq. 9 and 10. By using Eq. 11, it is shown that the next leaf length is lower than the current leaf length. By using (Eq. 12), it is shown that the next leaf direction will be the opposite of the current leaf direction:

$$p_{leafend, x} = p_{segstart, x} + l_{leaf} \cdot \cos(\alpha_{seg} + (\alpha_{leaf} \cdot \alpha_{leaf})) \quad (9)$$

$$p_{leafend, y} = p_{segstart, y} + l_{leaf} \cdot \sin(\alpha_{seg} + (\alpha_{leaf} \cdot \alpha_{leaf})) \quad (10)$$

$$l_{leafend, n+1} = (1 - r_{leafreduc}) \cdot l_{leaf, n} \quad (11)$$

$$o_{leaf, n+1} = \begin{cases} 1, & o_{leaf, n} = -1 \\ -1, & o_{leaf, n} = 1 \end{cases} \quad (12)$$

Besides creating leaf, the process also creates flower peduncle. Peduncle is not created in every leaf creation.

The peduncle creation is created based on the stochastic system and based on the flowering probability (r_{flo}). The peduncle creation is determined by using (Eq. 13). In Eq. 13, A is the action. Peduncle is created if the A is 1 and is not created if the A is 0:

$$A = \begin{cases} 1, & \text{random}(0,100) \leq r_{flo} \\ 0, & \text{random}(0,100) > r_{flo} \end{cases} \quad (13)$$

When the peduncle is created, some flower variables are created. The variables are the number of peduncle segment (n_{pedseg}), the peduncle segment length (l_{ped}), the starting peduncle segment that will create flower (s_{flo}), the starting bud segment (s_{bud}), the pedicel length (l_{pedi}), the pedicel orientation (o_{pedi}), the peduncle angle (α_{pedu}) and the peduncle deviation angle (Δ_{pedu}). These variables values are determined by using Eq. 14-21:

$$n_{pedseg} = \text{random}(n_{pedsegmin}, n_{pedsegmax}) \quad (14)$$

$$l_{pedu} = \text{random}(l_{pedusegmin}, l_{pedusegmax}) \quad (15)$$

$$s_{flo} = \text{random}(1, n_{pedseg}) \quad (16)$$

$$s_{bud} = \text{random}(s_{flo}, n_{pedseg}) \quad (17)$$

$$l_{pedi} = \text{random}(l_{pedimin}, l_{pedimax}) \quad (18)$$

$$p_{pedustart} = p_{segend} \quad (19)$$

$$o_{pedi} = \begin{cases} 1, & \text{random}(0,100) < 50 \\ -1, & \text{random}(0,100) \geq 50 \end{cases} \quad (20)$$

$$\alpha_{pedu} = \alpha_{seg} + \text{random}(-\Delta_{pedu}, \Delta_{pedu}) \quad (21)$$

In Eq. 14-19, some variables are used. The $n_{pedsegmin}$ is the minimum number of peduncle segments while the $n_{maxpedseg}$ is the maximum number of peduncle segments. The $l_{pedusegmin}$ is the minimum peduncle segment length while the $l_{pedusegmax}$ is the maximum peduncle segment length. The $l_{pedimin}$ is the minimum pedicel length while the $l_{pedimax}$ is the maximum pedicel length. The $p_{pedustart}$ is the starting position of the peduncle segment. The peduncle growth algorithm is shown in algorithm 3.

Algorithm 3: Peduncle growth algorithm:

```

begin
  ipedseg ← 1
  while ipedseg ≤ npedseg
  begin
    Ppeduend,x ← Ppeduend,x + lpedu * cos(αpedu)
    Ppeduend,y ← Ppeduend,y + lpedu * sin(αpedu)
    createflower()
  end
  if ipedseg ≥ Sfl then
  createflower()
  αpedu ← αpedu + random(-Δpedu, Δpedu)
  Ppedustart,x ← Ppeduend,x
  Ppedustart,y ← Ppeduend,y
  end
end
  
```

In algorithm 3, the i_{pedseg} is the peduncle segment index. The createflower() procedure is procedure to create blossoming flower or flower bud. This decision is determined by using Eq. 22. In the createflower() procedure, even it is a blossoming flower or a flower bud, the pedicel must be created. The pedicel end position ($P_{pediend}$) is determined by using Eq. 23 and 24. The next pedicel orientation is determined by using Eq. 25:

$$A = \begin{cases} \text{blossom}, i_{pedseg} \leq S_{bud} \\ \text{bud}, i_{pedseg} > S_{bud} \end{cases} \quad (22)$$

$$P_{pedi,x} = P_{peduend,x} + l_{pedi} \cdot \cos(\alpha_{pedu} + o_{ped}, 90) \quad (23)$$

$$P_{pedi,y} = P_{peduend,y} + l_{pedi} \cdot \sin(\alpha_{pedu} + o_{ped}, 90) \quad (24)$$

$$o_{ped,n+1} = \begin{cases} 1, o_{ped,n} = -1 \\ -1, o_{ped,n} = 1 \end{cases} \quad (25)$$

The next model is the root growth model. This root growth model adopts the fibrous root system as it is developed in the previous work (Kusuma, 2017b) that is developed based on L-system method. In this research in the root system, there are two root types: primary root and secondary root. Primary root is root that grows from the rhizome. Secondary root is root that grows from primary root. This concept is a little bit different rather than the previous work where there is not any root classification (Kusuma, 2017b). The illustration is shown in Fig. 5.

Based on root system illustration in Fig. 5, some variables are used in the root modeling. These variables are: root segment length ($l_{rootseg}$), root angle (α_{root}), root split angle ($\alpha_{rootsplit}$), root deviation angle (α_{root}), split ratio (r_{split}), number of primary root segments ($n_{primroot}$) and number of secondary root segments ($n_{secroot}$). These variables value is set by using Eq. 26-31:

$$l_{rootseg} = \text{random}(l_{rootmin}, l_{rootmax}) \quad (26)$$

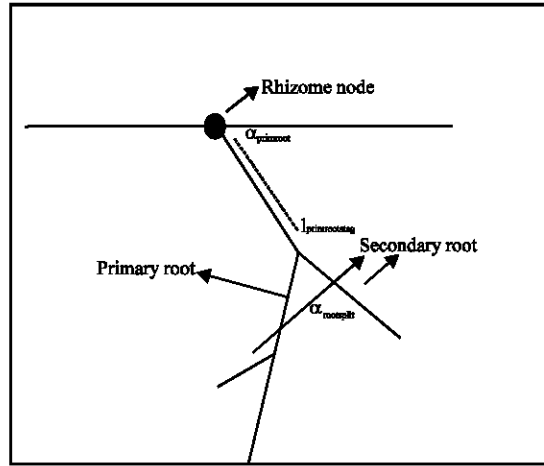


Fig. 5: Root system illustration

$$\alpha_{root,n+1} = \alpha_{root,n} + \text{random}(-\Delta_{root}, \Delta_{root}) \quad (27)$$

$$n_{primroot} = \text{random}(n_{primrootmin}, n_{primrootmax}) \quad (28)$$

$$n_{secroot} = \text{random}(n_{secrootmin}, n_{secrootmax}) \quad (29)$$

$$A = \begin{cases} \text{split}, \text{random}(0,100) \leq r_{split} \\ \text{shoot}, \text{random}(0,100) > r_{split} \end{cases} \quad (30)$$

$$\alpha_{rootsplit} = \alpha_{root} + \text{random}(-\Delta_{devsplit}, \Delta_{devsplit}) \quad (31)$$

In Eq. 26-31, several new variables are used. The $l_{rootmin}$ is the minimum root segment length while the $l_{rootmax}$ is the maximum root segment length. The $n_{primrootmin}$ is the minimum number of primary root segments while the $n_{primrootmax}$ is the maximum number of primary root segments. The $n_{secrootmin}$ is the minimum number of secondary root segments while the $n_{secrootmax}$ is the maximum number of secondary root segments. The $\Delta_{devsplit}$ is the split deviation angle.

RESULTS AND DISCUSSION

Implementation: After being developed, the proposed Dendrobium model is implemented into batik pattern generation application. This application is developed on web platform, so that, it can be accessed easily. This program is developed by using PHP language. The result is batik image which its width is 2,000 pixels and its height is 2,000 pixels. In this research, the color selection that is used in batik pattern generation is described in Table 1. The single rhizome Dendrobium Model is shown in Fig. 6. After the single rhizome Dendrobium image is created, then the array of Dendrobium based batik pattern is generated. This array of Dendrobium can be seen as

Table 1: Color selection

Objects	R	G	B
Lower background	247	222	208
Upper background	216	156	11
Stem	126	207	137
Leaf	5	105	19
Peduncle	167	195	12
Pedicel	167	195	12
Sepal	240	14	211
Bud	216	156	11
Root	186	163	102



Fig. 6 : Single rhizome result

two dimensional matrix with each cell contains one rhizome Dendrobium. The algorithm to create array of Dendrobium is shown in algorithm 4.

Algorithm 4: Array of dendrobium algorithm:

```

begin
  px ← pstart, x
  while px < pend, x
  begin
    py ← pstart, y
    while py < pend, y
    begin
      creat dendrobium (px, py)
    end
  end
end

```

In algorithm 4, some variables and procedure are used. The createDendrobium() procedure is the procedure that is used to create single rhizome Dendrobium. The p is the position of the rhizome. The p_{start} is the array starting point and the p_{end} is the array end point.



Fig. 7: Final result image

Besides the Dendrobium object in batik pattern, the background object is concerned too. In this research, we adopt the Kawung pattern. Kawung is one of Javanese traditional pattern. The Kawung pattern generation algorithm is shown in algorithm 5.

Algorithm 5: Background image generation algorithm:

```

begin
  create_oval (kx, ky+dk, lshort, llong)
  create_oval (kx+dc, ky, llong, lshort)
  create_circle (kx-dc, ky, lcircle)
  create_circle (kx+dc, ky, lcircle)
  create_circle (kx, ky+dc, lcircle)
end

```

In algorithm 5, some variables and procedures are used. The create_oval procedure is used to create oval object. The create_circle procedure is used to create circle object. The k is the position of the center point. The d_k is the distance between the center point and the center of the Kawung object. The d_c is the distance between the center point and the inner circle object. The l_{short} is the width of the Kawung. The l_{long} is the length of the Kawung. The l_{circle} is the diameter of the inner circle. The result image is shown in Fig. 7.

After the proposed model is implemented into batik pattern generation application, several tests are done. The purpose of these tests is to observe the relationship between the adjusted variable and the observed variable. The first test is to observe the relation between s_{leaf} and n_{leaf}. The second test is to observe the relation between the r_{fl0} and n_{pedu}. The third test is to observe the relation

Table 2: Relation between n_{seg} and n_{leaf}

n_{seg}	n_{leaf}
1	8.2
2	9.8
3	14
4	15
5	21
6	31
7	29
8	30
9	39.7
10	40.7

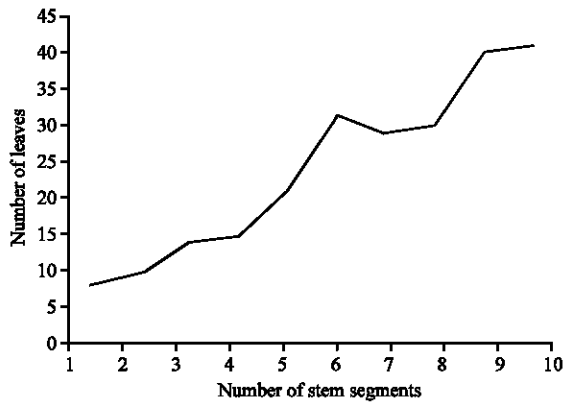


Fig. 8: Relation chart between n_{seg} and n_{leaf}

between n_{pedseg} and n_{flo} . In these tests, some variables are set default. In these tests, there is only one rhizome. The n_{min} is 5 and n_{max} is 10.

In the first test, the relation between the number of segments (n_{seg}) and the number of leaves (n_{leaf}) are observed. In this test, n_{seg} acts as the controlled variable and the n_{leaf} acts as the observed variable. The n_{seg} ranges from 1-10 with step unit is 1. The result is shown in Table 2. The trend is shown in Fig. 8.

Based on data in Table 2, it is shown that when the number of stem segment increases, the number of leaves increases too. Based on data in Fig. 8, it is shown that the trend is linear. Based on these statements, it can be said that the relation between number of stem segments and the number of leaves is positive linear.

The second test is to observe the relation between the flowering ratio (r_{flo}) and the number of peduncle (n_{pedu}). In this test, the r_{flo} acts as the controlled variable and the n_{pedu} acts as the observed variable. The n_{segmin} is set 8 and the n_{segmax} is set 10. The r_{flo} ranges from 10-100 with the step unit is 10. The result is shown in Table 3 and the chart view is shown in Fig. 9.

Based on data in Table 3, it is shown that when the flowering ratio increases then the number of peduncles increases too. Based on data in Fig. 9, it is shown that the

Table 3: Relation between r_{flo} and n_{pedu}

r_{flo}	N_{pedu}
10	4.4
20	6.9
30	12.5
40	15.1
50	18.4
60	23.9
70	29.1
80	29.4
90	39.1
100	38.0

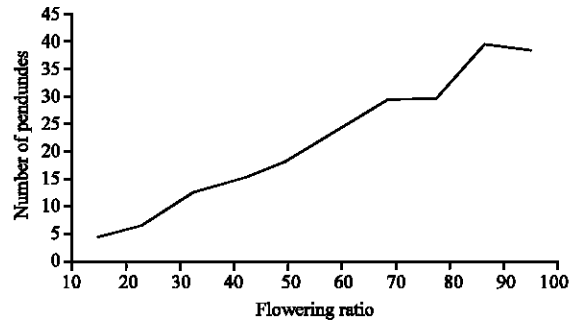


Fig. 9: Relation chart between r_{flo} and n_{pedu}

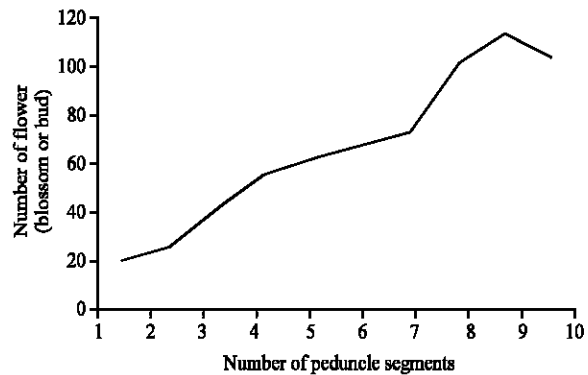


Fig. 10: Relation chart between n_{pedseg} and n_{flo}

trend is linear. Based on these statements, it can be said that the relation between flowering ratio and the number of peduncles is positive linear.

The third test is the relation between the number of peduncle segments (n_{pedseg}) and the number of flowers (n_{flo}). The n_{pedseg} acts as the controlled variable while the n_{flo} acts as the observed variable. The number of flowers includes the number of blossoming flowers and the number of flower buds. In this test, the r_{flo} is set 50. The n_{pedseg} ranges from 1-10 with step size is 1. The result is shown in Table 4 and the chart is shown in Fig. 10.

Based on data in Table 4, it is shown that when the number of peduncle segments increases, the number of flowers increases too. Based on data in Fig. 10, it is shown

Table 4: Relation between n_{pedunc} and n_f

n_{pedunc}	n_f
1	19.2
2	25.8
3	41.5
4	54.3
5	61.5
6	65.8
7	72.1
8	99.1
9	113.0
10	103.2

that the trend is positive linear. Based on these statements, it can be said that the relation between the number of peduncle segments and the number of flowers is positive linear.

CONCLUSION

Based on the explanation in the previous sections, it can be concluded that the proposed model can produce the Dendrobium model. The plant parts that are created includes: root, stem, leaf, flower peduncle, flower pedicel, flower and flower bud. This model has been implemented into batik pattern generation application and by using this application, Dendrobium based batik pattern can be generated. In this research, the traditional Kawung pattern is used as the background pattern.

Based on the tests that have been done, the relation between adjusted variable and the observed variable is as follows. The relation between the number of stem segments and the number of leaves is positive linear. The relation between the flowering ratio and the number of flower peduncles is positive linear. The relation between the number of peduncle segments and the number of flowers is positive linear.

RECOMMENDATIONS

There is big research potential in developing digital batik by using computational technology. It is because there are many nature objects that has not implemented into batik pattern. The objects that have been implemented into batik pattern are generic and far from specific. So, developing and modeling specific nature objects into batik pattern or other artistic pattern is still challenging.

REFERENCES

Boudon, F., C. Pradal, T. Cokelaer, P. Prusinkiewicz and C. Godin, 2012. L-Py: An L-system simulation framework for modeling plant architecture development based on a dynamic language. *Front. Plant Sci.*, 3: 1-7.

Florek, S., 2011. The batik process. Australian Museum, Darlinghurst, Australia.

Gallis, H.A., 2010. The genus Dendrobium: Overview with notes on cultural requirements. Sandshills Orchid Society, Fayetteville, North Carolina.

Gelling, P., 2009. Score one for Indonesia in the war over Batik. *The New York Times*, *The New York Times*, New York, USA.

Guan, K., 2008. The research of the improved 3D L-system and its application in plant modeling. *Proceedings of the 2008 IEEE International Conference on Cybernetics and Intelligent Systems*, September 21-24, 2008, IEEE, Chengdu, China, ISBN:978-1-4244-1673-8, pp: 718-723.

Kongsombut, K. and R. Chairicharoen, 2014. Probabilistic growth model for dendrobium orchid. *Proceedings of the 2014 Asia-Pacific Annual Summit and Conference on Signal and Information Processing Association (APSIPA)*, December 9-12, 2014, IEEE, Siem Reap, Cambodia, pp: 1-5.

Kusuma, P.D., 2017b. Fibrous root model in batik pattern generation. *J. Theor. Appl. Inf. Technol.*, 95: 3260-3269.

Kusuma, P.D., 2017a. Graph based simplified crack modeling in batik pattern generation. *J. Theor. Appl. Inf. Technol.*, 95: 5035-5046.

Kusuma, P.D., 2017c. Interaction forces-random walk model in traditional pattern generation. *J. Theor. Appl. Inf. Technol.*, 95: 3294-3302.

Li, H., M. Luo and X. Zhang, 2017. 3D reconstruction of orchid based on virtual binocular vision technology. *Proceedings of the 2017 4th International Conference on Information Science and Control Engineering (ICISCE)*, July 21-23, 2017, IEEE, Changsha, China, ISBN:978-1-5386-3014-3, pp: 1-5.

Maulia, E., 2009. Batik selected for UNESCO cultural heritage list. *The Jakarta Post*, Jakarta, Indonesia.

McCord, G., B. Wunsche, B. Plimmer, G. Gilbert and C. Hirsch, 2008. A pen and paper metaphor for orchid modeling. *Proceedings of the 3rd International Conference on Computer Graphics Theory and Application (GRAPP)*, January 22-25, 2008, GRAPP, Funchal, Madeira, Portugal, pp: 119-124.

- Parnata, A.S., 2005. [Cultivation Guide and Orchid Care]. AgroMedia Pustaka, Jakarta, Indonesia, (In Indonesian).
- Pickover, C.A., 1995. The Pattern Book: Fractals, Art and Nature. World Scientific, Singapore, Asia, Pages: 425.
- Prusinkiewicz, P. and A. Lindenmayer, 1990. The Algorithmic Beauty of Plants. 1st Edn., Springer-Verlag, New York, USA., ISBN-13: 9780387972978, Pages: 228.
- Sun, R., J. Jia and M. Jaeger, 2009. Intelligent tree modeling based on L-system. Proceedings of the IEEE 10th International Conference on Computer-Aided Industrial Design and Conceptual Design, November 26-29, 2009, IEEE, Wenzhou, China, ISBN:978-1-4244-5266-8, pp: 1096-1100.
- Yulianto, R., M. Hariadi and M.H. Purnomo, 2012. Fractal based on noise for batik coloring using normal Gaussian method. IPTEK. J. Technol. Sci., 23: 34-40.