Modeling Plant Leaves in Marble-Patterned Colours with Particle Transportation System

Yodthong Rodkaew[1], Prabhas Chongstitvatana[1], Suchada Siripant[2] and Chidchanok Lursinsap[2].

[1]Department of Computer Engineering, Chulalongkorn University. Bangkok, Thailand.
[2]AVIC, Department of Mathematics, Chulalongkorn University. Bangkok, Thailand.

email: yodthong.r@student.chula.ac.th, prabhas@chula.ac.th, ssuchada@chula.ac.th, lchidcha@chula.ac.th

Abstract

This work presents Particle Transportation System for modeling plants in colours. The algorithm is initiated by randomly scattering particles inside a given shape. Each particle contains energy. The transportation rule directs each particle toward a target. When particles are in close proximity, they are combined. The trails of moving particles are used to generate the venetion patterns. The algorithm is effective, it has been tested with various shapes. It is computationally efficient, and has only a few parameters. This algorithm is suitable for the colour production in leaf models. The diffusion of colour occurs by the motion of particles. The mixed colours generate vivid marble patterns. The proposed method is illustrated by the model of the taro vine leaf (*Scindapsus aureus, Epipremnum pinnatum*). The method can be used for other plants.

Keywords: Plant Model, Leaf Model, Vein, Particle Systems.

1. Introduction

The synthetic modeling for natural shapes is a challenging problem in computer graphics and related fields. There are many systems that are faithful to the botanical model for examples; Aono and Kunii (1984), De Reffye (1988), Ramification Patterns (Viennot et al. 1989), L-Systems (Prusinkiewicz and Lindenmayer 1990). Chiba et al. (1994) introduced a system based on virtual heliotropism and dormancy breaks. They introduced colours in (Chiba et al. 1996). There are many other systems for plant modeling such as GreenLab (Hu et al. 2003), AMAP (CIRAD, France), and PlantVR which is based on Parametric L-Systems (Chuai-aree et al. 2003). Rodkaew et al (2002a) modeled leaves with L-systems and genetic algorithms. The results in the plant modeling are useful for the simulation through virtual reality. However it is difficult to model a natural shape such as a tree or a leaf because of the richness of complex details in plant architecture (Barthélémy 2003).

Rodkaew et al. (2002b) proposed a new Particle Transportation System (Fig. 1). The algorithm is proposed for generating leaf vein structures from a given outline shape. The motivation comes from the transportation behavior of nature. The proposed algorithm is based on efficient construction of veins.

Fig. 1. The leaves generated from the algorithm.

Particle Transportation System works well for structure of trees and roots including interaction with an environment (Rodkaew et al. 2003).

This work aims for computer graphics modeling. We extened Particle Transportation System to generate pattern of colours in leaves.

2. Particle Transportation System

Leaf modeling using structure-based approach is very difficult. The leaf venetion structure is extremely complex. Particle Transportation System is developed to generate vein images of leaves. The outline of leaf shape and the target point are the main input for the system. The motivation of the algorithm is based on observing natural phenomenon.

2.1 Motivation

In nature, leaves are the essential part of plants that have a role for photosynthesis process. The green substances inside leaves contain chlorophyll for energy absorption from sunlight to help plants grow. During photosynthesis process, the veins are used for transporting energy between leaves and trunks. The arborescent veins that immersed inside leaves are useful for both transporting energy and supporting the leaf structure. The diameter of a vein reflects capacity. The larger diameter of vein transports more energy. A real leaf shape is controlled by its species which depends on genetic. We assume an energy conservative model. Each part of leaf has equivalent chlorophyll and absorbs equivalent energy. All energy is sent to the plant via veins. Each area can be represented as a particle inside the leaf shape. The particles are scattered inside the whole shape. The particles represent energy which is transported to the petiole (Fig. 2a). The transportation should be efficient. Algorithm I shows the main idea.

Algorithm I

```
1. Set the particles randomly in the boundary of a leaf shape.
2. Move the particles to the target (the petiole).
3. Repeat (2) until all particles reach the target.
```

There are many paths to the petiole. To improve the efficiency, the path must be shared. Algorithms II presents this idea. While particles move toward the target, they are forced to share their paths. This is accomplished by moving particles toward each others, when two particles come close, they are merged together. The two particles combine energy and other attributes. After the combination, they become a new particle. The process is repeated until all particles reach the target. Figure 2b shows this behavior. Several kinds of leaf have this pattern. The next section describes the algorithm in more details.

Algorithm II
1. Set the particles randomly in the boundary of a leaf shape.
2. Sharing a path: move toward the nearest particle and the target, when two particles are close, combine them.
3. Repeat (2) until all particles reach the target.

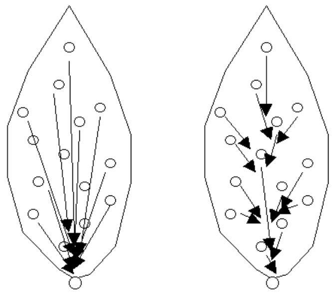

(2a) (2b)

Fig. 2. The diagram of the movement of particles .

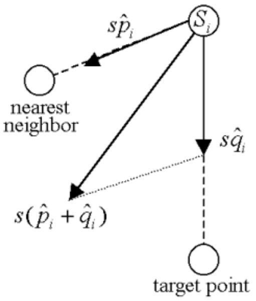


Fig. 3. The motion of a particle.

2.2 Particle Transportation Algorithm

The algorithm creates veins from the trail of particles that are scattered within a leaf shape. Let S be the set of particles. At first, each particle σ_i carries the energy e_i. The direction of motion of a particle is controlled by an equation : $\hat{p}_i + \hat{q}_i$ where $\hat{p}_i$ is a unit vector denoting the direction from a particle to its nearest neighbor and $\hat{q}_i$ is a unit vector denoting the direction from a particle to a presumed target point (Fig. 3).

At each time step all particles move by a distance *s*, when the particles σ_i and σ_j come closer than r they are combined together and the energy is conserved $e_{new} = e_i + e_j$. While moving, the particle creates a trail with the width $w_i = f(e_i)$ hence the thickness of veins is increased when it is closer to the petiole. The process is repeated until only one particle remains. There are many parameters governing the generation of vein images. They are divided into two groups: particle distribution and their motions. The distribution pattern determines the initial positions of particles and their density. The parameters determining the motion include the neighborhood radius, the initial energy of each particle and the speed. The trails of particles generate the vein structure inside the given leaf shape. Fig. 4 shows the development of vein structure using this algorithm. Fig 5 shows the final result after the image enhancement.

ALGORITHM PARTICLE TRANSPORTATION

Given an outline boundary:
S is a set of particles. **P** $\in$ **S** contains: pos (position: a point (x,y)), en (energy), rd (radius). **T** is the target point (at the bottom of a leaf). $\mathbf{W_p}$, $\mathbf{W_q}$ are weight factors

```
1.  Place S inside the boundary
2.  For each P do
3.    N is the nearest particle to P
4.    if P.pos - N.pos < P.rd + N.rd then
      // combine two particles
5.      P.pos ← (P.pos + N.pos)/2
6.      P.en ← P.en + N.en
7.      S ← S - {N}
    // check P reaches the target
8.    if | T - P.pos | < P.rd then // delete P
9.      S ← S - {P}
10.   else // move P
11.     $V_1$ ← normalise(T - P.pos)
12.     $V_2$ ← normalise(N.pos - P.pos)
13.     $V_3$ ← normalise(($W_p \times V_1 + W_q \times V_2$)/($W_p + W_q$))
14.     P.pos ← P.pos + $V_3 \times$ stepsize
15. repeat 2-14 until S = $\phi$
```

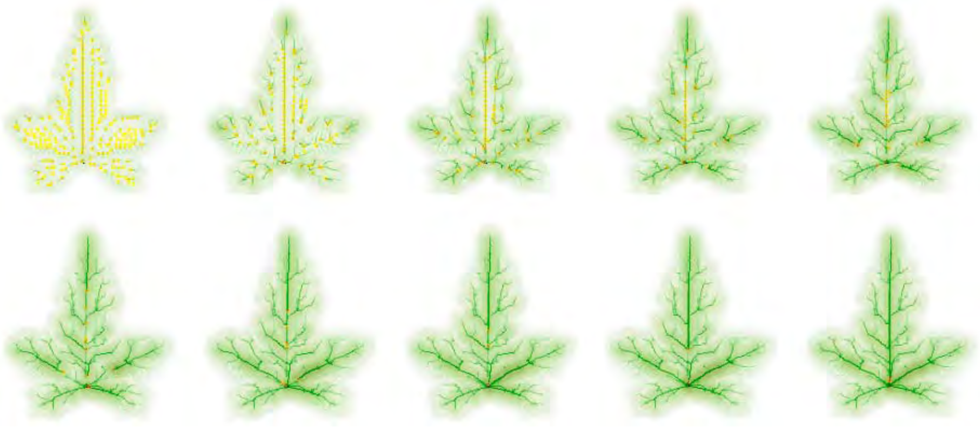


Fig. 4. The development of vein structure.

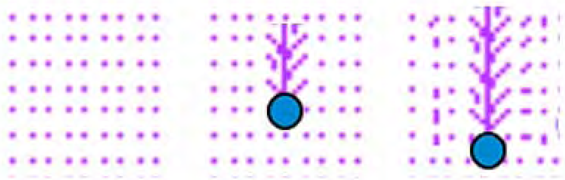
Fig. 5. The image enhancement of the output.

3. Particle Transportation Systems for Colour Model

We propose the colouring method based on Particle Transportation System. The vein structures are created from the trails of particle motions. This structure is related to the colour of a leaf. The direction of colour motion is related to the pattern of the vein structure. The colour pattern in a leaf is ellipsoid or diffused colour travels through the veins. The leaf of taro vine (*Scindapsus aureus, Epipremnum pinnatum*) is an example of this case.

3.1 Vector Map

The colour pattern in a leaf is usually either ellipsoid or diffused colour which travels through the veins. We introduce Vector Map to implement this model. The vector map consists of two-dimensional array of vectors. The array of vectors cover the area of leaf. Initially, the vectors are set to zero vectors. When the particles move, the vectors will be changed due to the influence from the motion of particles. The vectors near a particle will point in the same direction of the motion of the particle. The length of vector indicates the distance between the particle and the vector. In Fig. 6, demonstrates the movement of particle (the big dot) moving from top to bottom over the vector map (Fig. 6a-6c).



(a) (b) (c)

Fig. 6. The movement of particle over Vector Map.

3.2 Modeling Process

The process consists of six steps (see Fig. 7). The taro vine leaf is used for illustration of the method.

1. Generate an outline using spline curves.
2. Generate vein structures by Particle Transportation System.

3. Calculate the vector map.
4. Perform colour diffusion (see Section 3.3)
5. Tuning colour by picking real colour tone from the real leaf.
6. Add textures.

3.3 Colour diffusion in Vector Map

Colours are dropped into Vector Map (see Fig. 8). The colour diffuses from the center of a leaf to the edge by reversing the direction of vectors. The calculation of colour motion is as follows:

```
ALGORITHM COLOURVECTORMAP

newVectorMap.colour ← VectorMap.colour
for each i ← 1 to VectorMap.width do
  for each j ← 1 to VectorMap.height do
    $i_x$ ← VectorMap[i, j].vector.x + i
    $i_y$ ← VectorMap[i, j].vector.y + j
    $c_1$ ← VectorMap[i, j].colour
    $c_2$ ← newVectorMap[$i_x$, $i_y$].colour
    newVectorMap[$i_x$, $i_y$].colour ← MixingColour( $c_1$, $c_2$ )
  next j
next i
```

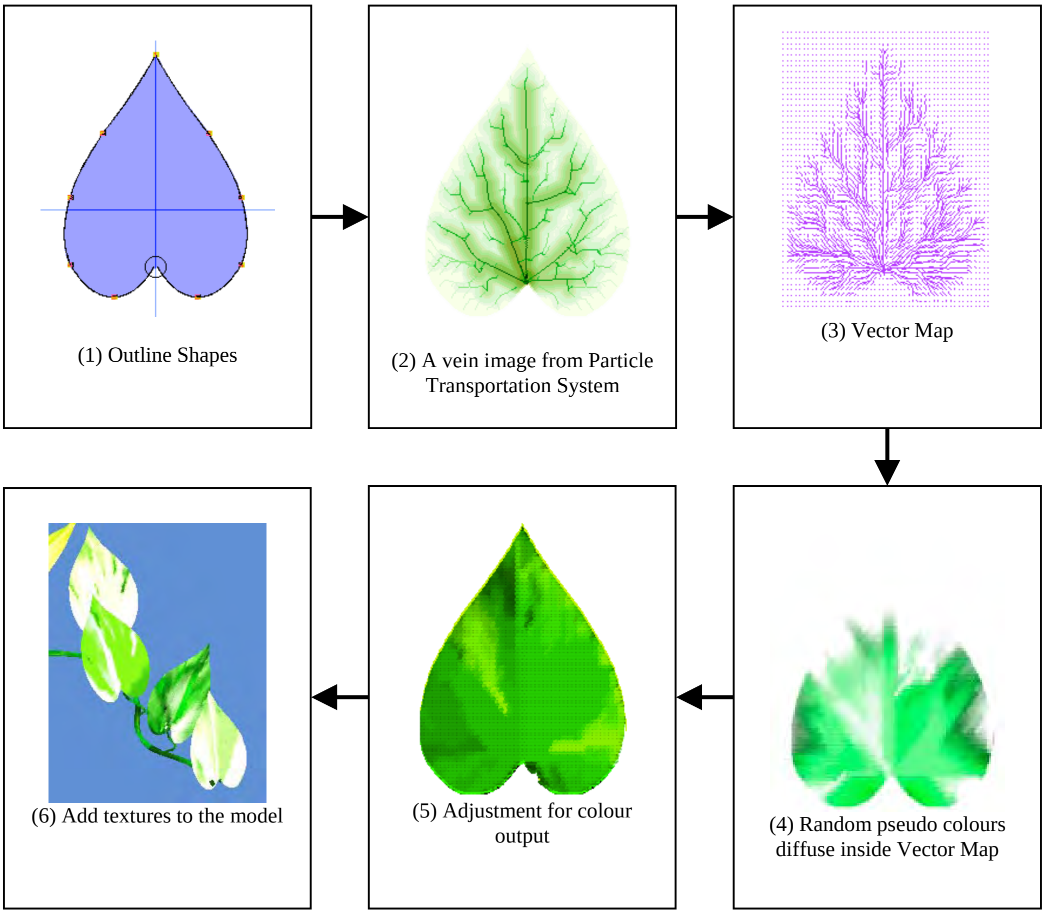
The colour C_1 at (i,j) is mixed with the colour C_2 at (ix,iy) using MixingColour function as in Eq.1.

$$\begin{aligned} R_{new} &= w_1 R_1 + w_2 R_2 \\ G_{new} &= w_1 G_1 + w_2 G_2 \\ B_{new} &= w_1 B_1 + w_2 B_2 \end{aligned} \qquad \text{Eq. 1.}$$

The colour is separated into RGB components. The colour RGB_{new} produced by RGB_1 merged with RGB_2 . We use $w_1 = 0.7$ and $w_2=0.3$. Figure 9 shows the colour diffusion process. The different marble patterns are possible by random drop of pseudo colours.

4. Results

The synthetic taro vine leaves are shown in Fig. 10. The vine models (ivy and taro) are shown in Fig. 11. The stem of the vines are created by spline functions. The shadow in Fig. 11 are created by the projection of polygon surfaces. The leaf models are deformed polygons and are varied by size. The leaf textures are pre-generated using Particle Transportation System. The images are rendered with OpenGL.



(1) Outline Shapes

(2) A vein image from Particle Transportation System

(3) Vector Map

(4) Random pseudo colours diffuse inside Vector Map

(5) Adjustment for colour output

(6) Add textures to the model

Fig. 7. The six steps for synthesis taro vine leaf.

Fig. 8. The colour diffusion inside Vector Map.

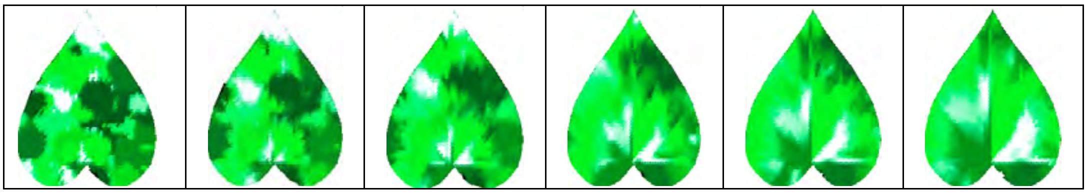


Fig. 9. Add more colour drop while colours are diffused inside Vector Map.

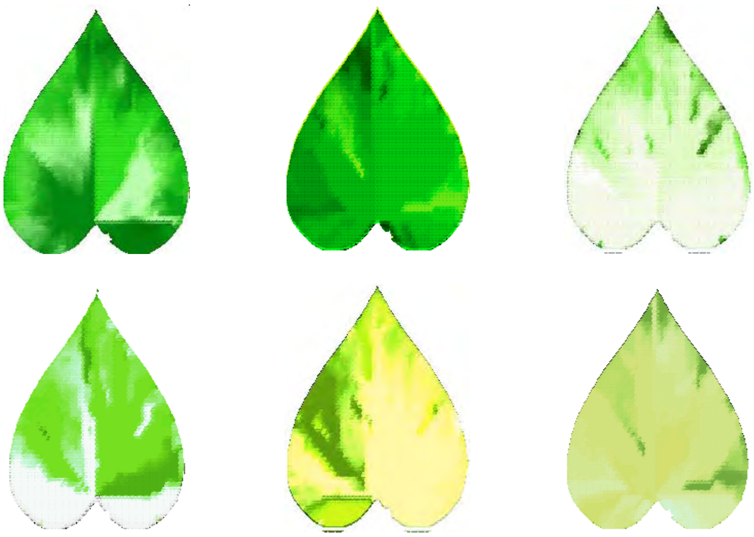


Fig. 10. Taro Vine (Marble Queen) *Scindapsus aureus* leaves from the algorithm.

Fig. 11. Various views of generated vines.

5. Conclusion

In this work, we introduce Particle Transportation System. It is exteneded to model marble-patterned colour using colour diffusion process. The proposed method is illustrated with the taro vine leaf. (or called Marble Queen, *Scindapsus aureus*, *Epipremnum pinnatum*). Particle Transportation System is suitable for plant modeling. It has many advantages:

- The model can be constructed for a given shape.
- It is easy to use and requires small number of parameters.
- The computation is efficient.
- It is extensible.
- It can incorporate the effect from the environment such as shadow effect.

However, there are several kinds of leaves and trees that can not be constructed using this algorithm. In future work, we are searching for other suitable parameters and extension of algorithms which can handle more shapes.

References

AONO. M., AND KUNII, T. 1984, Botanical Tree Image Generation, In *IEEE Computer Graphics and Applications*, 4 (5), 10-34.

BARTHELEMY, D. 2003, Botanical Background for Plant Architecture Analysis and Modeling, In *Proceedings of Plant International Symposium on Plant Growth Modeling, Simulation, Visualization and their Applications*, HU, B.-G., AND JAEGER, M., Eds, Beijing, China PRC, 13-16 OCT 2003, 1-20.

CHIBA, N., OHSHIDA, K., MURAOKA, M., AND MIURA, M. 1994, Visual simulation of botanical trees based on virtual heliotropism and dormancy break, In *The Journal of Visualization and Computer Animation*, 5,(1), 3-15.

CHIBA, N., OHSHIDA, K., MURAOKA, M., AND SAITO, N. 1996, Visual simulation of Leaf Arrangement and Autumn Colours, In *The Journal of Visualization and Computer Animation*, 7, 79-93.

CHUAI-AREE, S., JAEGER, W., BOCK, H.G., SIRIPANT, S., AND LURSINSAP, C. 2003, PlantVR: An Algorithm for Generating Plant Shoot and Root Growth Using Applied Lindenmayer Systems, In *Proceedings of Plant International Symposium on Plant Growth Modeling, Simulation, Visualization and their Applications*, HU, B.-G., AND JAEGER, M., Eds, Beijing, China PRC, 13-16 OCT 2003, 43-53.

De REFFYE, P., EDELIN, C., FRANCON, J., JAEGER, M., AND PUECH, C. 1988, Plant Models faithful to Botanical Structure and Development, In *Computer Graphics Proceedings*, 22 (4), 151-158.

HU, B.-G., De REFFYE., P., ZHAO, X., YAN, H.-P., AND KANG, M.-Z. 2003, GreenLab: A New Methodology towards Plant Functional-Structural Model -- Structural Aspect, In *Proceedings of Plant International Symposium on Plant Growth Modeling, Simulation, Visualization and their Applications*, HU, B.-G., AND JAEGER, M., Eds, Beijing, China PRC, 13-16 OCT 2003, 21-35.

PRUSINKIEWICZ, P., AND LINDENMAYER, A. 1990, *The algorithmic beauty of plants*, Springer-Verlag.

RODKAEW, Y., CHONGSTITVATANA, P., SIRIPANT, S., AND LURSINSAP, C. 2003, Particle Systems for Plant Modeling, In *Proceedings of Plant International Symposium on Plant Growth Modeling, Simulation, Visualization and their Applications*, HU, B.-G., AND JAEGER, M., Eds, Beijing, China PRC, 13-16 OCT 2003, 210-217.

RODKAEW, Y., SIRIPANT, S., LURSINSAP, C., CHONGSTITVATANA, P., FUJIMOTO, T., AND CHIBA, N. 2002a, Modeling Leaf Shapes using L-system and Genetic Algorithms, In *proceedings of NICOGRAPH2002*, Tokyo, Japan, 73-78.

RODKAEW, Y., SIRIPANT, S., LURSINSAP, C., AND CHONGSTITVATANA, P. 2002b, An algorithm for generating vein images for realistic modeling of a leaf, In *Proceedings of Computational Mathematics and Modeling*. Bangkok, Thailand, 2002.

VIENNOT, X.B., EYROOLS, G., AND JANEY N. 1989, Combinatorial Analysis of Ramified Patterns and Computer Imagery of Tree, In *Computer Graphics*, 23 (3), 31-40.