Particle Systems for Plant Modeling

Particle Systems for Plant Modeling

Yodthong Rodkaew[1], Prabhas Chongstitvatana[1],
Suchada Siripant[2] and Chidchanok Lursinsap[2]
(1: Department of Computer Engineering, Chulalongkorn University, Bangkok, Thailand. 10330)
(2: AVIC, Department of Mathematics, Chulalongkorn University, Bangkok, Thailand. 10330)

Abstract

This paper presents a new algorithm for modeling plant structures. The main motivation of this algorithm is based on certain natural phenomena. A property of the plant structure is to transport and exchange energy, water and sustenance between roots, branches and leaves. The plant structures should be suitable for efficient transportation. The algorithm employs Particle Systems. 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 reconstruct plant structures. In addition, the light effect from an environment is incorporated. The algorithm is effective, it has been tested with various shapes. It is computationally efficient, and has only a few parameters. The resulting images are realistic.

Keywords: Computer Graphics, Particle Systems, Leaf Model, Plant Model.

1 Introduction

It is difficult to model a natural shape such as a tree or a leaf because the richness of complex details. Realistic shape modeling is a significant topic in computer graphics. Augmented plants in architectural images enhance the work as well as in digital movies, music video or computer games. Realistic plant models are hard to create. The problem can not be solved using scanned 3D data from the real shape because the need to interact with the environment.

Fig. 1. Natural structure generated from the algorithm.

The synthetic modeling is suitable for modeling natural shapes. A tree consists of many parts such as branches, roots, leaves and flowers. Each part can be modeled separately and then composed into a whole plant. Complicate hierarchical links during the composition required human attention. Finding rules and fitting parameters appropriate for the rules are not a trivial task [1]. The process consumes a lot of computation power and resources. Usually, a tree is constructed with recursive functions. There are other limitations. Small changes in one part affect the whole shape, for example, changing the branch's angle in a small part may change the whole structure. The rules are also very specific. Once the rules have been discovered for one species of plant, they maybe useless for another.

To develop plant models, our motivation comes from the behavior of nature. The proposed algorithm is based on efficient construction of veins covering several leaf shapes [2]. It also works well for the structure of trees and root systems including interaction with an environment such as light effect (Fig. 1). The modeling method is based on particle systems using rules to define interaction between particles.

The algorithm is explained in Section 2. Section 3 presents the result of generating veins of leaf shapes. Section 4 describes the use of algorithm for modeling trees. Finally, Section 5 gives conclusion.

2 Particle Systems

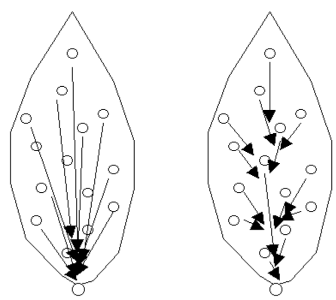
Reeves [3,4] used trajectory of particle systems that explode like fireworks for modeling grass. Our algorithm differs from Reeves's particle systems on the rules governing motion of particles. The algorithm is based on particle systems and rules to construct natural structures. It is suitable for structures such as veins of a leaf, or stems of a tree. The structure is generated by trail of particles. The basic idea of algorithm was proposed for modeling vein image of a leaf in [2]. However, the algorithm works well on trees and other problems of natural shapes.

The motivation of the algorithm is based on observing natural phenomenon. 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 absorbtion from sunlight to help plants grow. During photosynthesis process, the veins are used for transporting energy between leaves and trees. The arborescent veins that immersed inside leaves are useful for both transporting energy and supporting the leaf structure. The diameter of a vein reflects its capability for transportation. The larger diameter of vein transports more energy. The leaf shape depends on veins, in the opposite, the veins depend on the leaf shape. 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 scattered inside the whole shape. The particles produce the same amount of energy and the energy 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 will be many paths from particles to the target. To improve the efficiency, the path should be shared. Algorithms II presents this idea. Particles move to a target and also share their paths by forcing particles to move towards the nearest particles. 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. try to use the same path:
2.1 move towards the nearest particle
2.2 move to the target
3. repeat (2) until all particles reach the target



(2a) (2b)

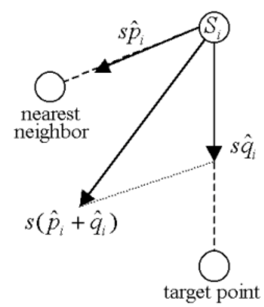


Fig. 2. The diagram of the movement of particles .

Fig. 3. The motion of a particle.

2.1 Particles transportation

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 $e_i = 1.0$. The direction of motions of particles 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 leaves 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 step size).

3 Leaf Model

The particles are scattered within a given leaf shape. There are two methods for acquiring shapes: 1) get an outline shape from a real leaf or 2) create a new shape using spline curves. Figure 4 shows the development of the veins from the motion of particles. The color of the trail reflects the amount of energy. The lower energy is presented by a light color. A darker color means higher energy. When particles move near to the petiole in the middle area of a leaf the trails are merged and create a dark color. After generating the vein image, some image improvement is performed to give depth and texture to the leaf image. The detail can be found in [2].

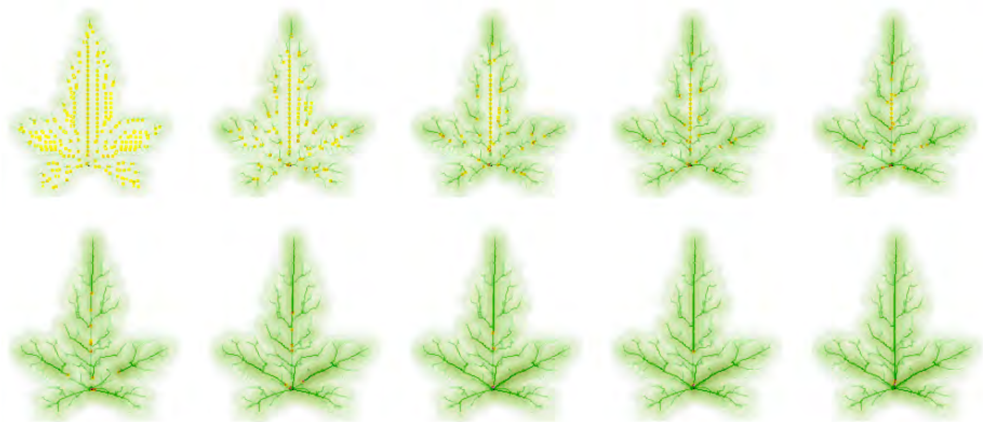


Fig. 4. The development of veins.

Figure 10 shows four different leaves. The outputs from our algorithm look quite realistic. This shows that the algorithm works well for many shapes. The picture of the result contains more information than what can be described in words. The output image can be compared with the real leaf. Both the main and secondary veins look like the real leaf. The small details are also well constructed.

4 Tree Model

A realistic leaf is modeled using the particle systems introduced in the previous section. The same concept can be applied to model a tree. In this section we extend particle systems for tree modeling (Fig. 5).

Many methods have been proposed for tree modeling. The modeling from shape [5] uses L-systems. In [6] the photographs are used to reconstruct 3D tree models. The outline shape of a tree can be given as in [7], the artist controlled spline functions to shape the tree. To use particle systems to model a tree, particle motions are extended to 3D space and the shape is also in 3D boundary. The outline shape of a tree is given. The roots of a tree can also be generated. Figure 6 shows the development of a tree using 3D particle systems.

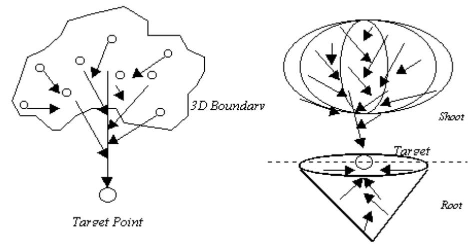


Fig. 5. The diagram of 3D Particle Systems (left) and tree modeling (right).

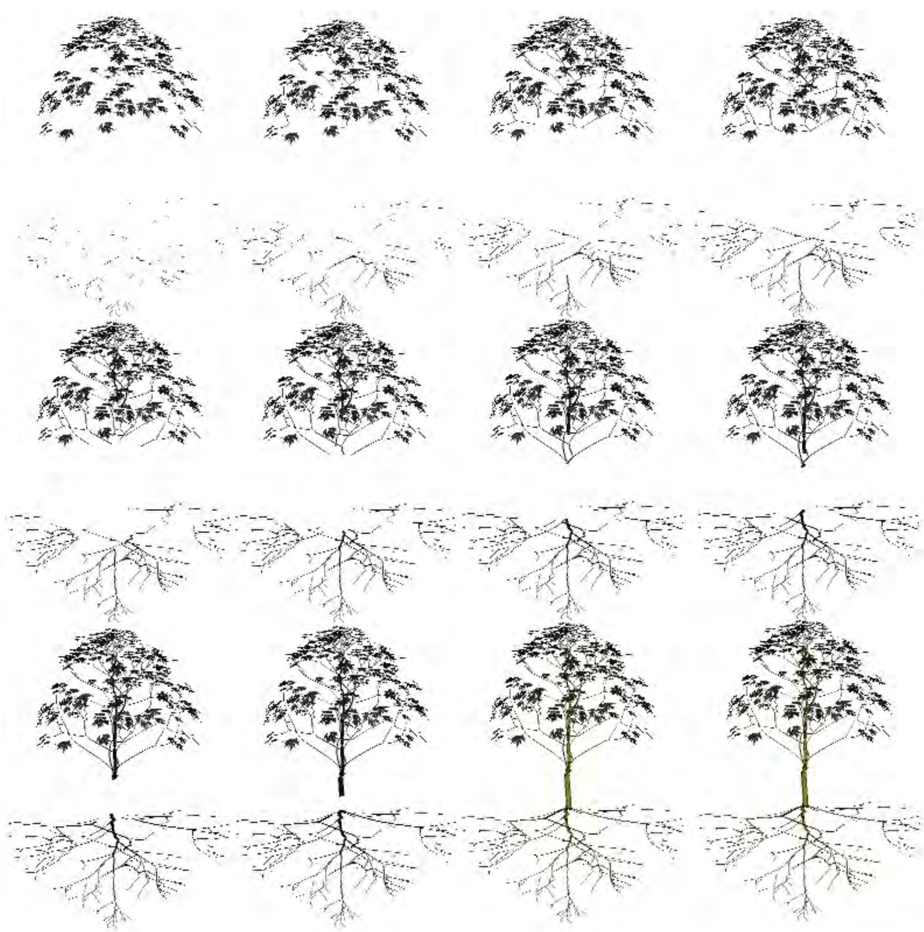


Fig. 6. The development of a tree.

4.1 Shape Boundary

The boundary is represented by a set of polygons. There are two boundaries, the outside boundary and the inside boundary. A volume bounded by the outside boundary and the inside boundary represents the shape. The particles scatter inside the boundary. The point-testing function calculates a cross-section in XZ-plane from the 3D boundary and determines whether a point is inside or outside this cross-section. The point-testing function is used twice for outside and inside boundary (Fig. 7). Using the inside boundary improves realism of the image because inside of the real plant has no leaves.

4.2 Light density effects

Leaves are at appropriate positions for receiving the light. The assumption that the density of particles are uniform in any area inside a given boundary is not true in nature. The density of leaves in an area is based on the amount of received light. To include the effect of light in the model, the light is calculated using ray-tracing. The light intensity is used for calculating the density of particles. The bright area contains more particles than the dark area. Figure 8 shows the light calculation in an 3D array. Because the semi-transparent property of the inside boundary, the shadow cast on the floor in the center is darker compared to the edge. Thus, self-shadow effect is automatically occurred. Figure 9 shows a twin tree.

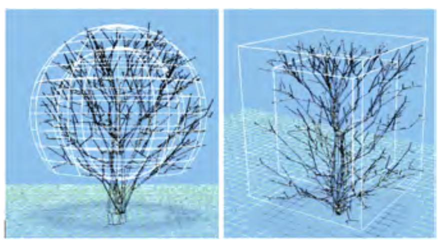


Fig. 7. The boundary substraction in sphere shape (left), and box shape (right).

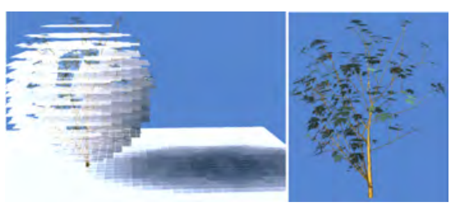


Fig. 8. Light calculation in 3D array (left), the model with light effect (right)

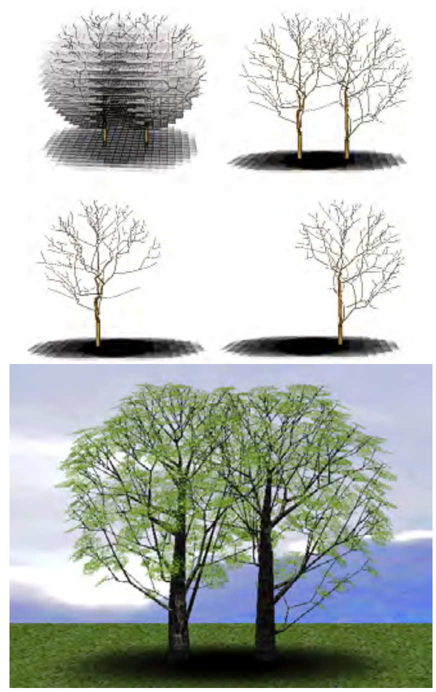


Fig. 9. The twin tree.

5 Conclusion

This work presents an algorithm for modeling natural structures. The algorithm is inspired by nature using the idea based on particle systems. It is easy to synthesize vein images from a given leaf shape and branches from a given boundary. The proposed algorithm uses a few parameters to control the quality of the output. The trails of the motion of interacting particles create very realistic natural structures. The particle system is suitable for plant modeling because of the following reasons:

- Realism – the output image looks very realistic.
- Simplicity – the method require small number of parameters.
- Construction from shape – it is easy to model from a given shape.
- Efficient computation – function fitting to shape is not required.
- Require less human assistance – no need to write the complex rules.
- Incorporate effect from environment – structured model for light-shadow effect is possible.

However, there are some kind of leaves and trees that can not be constructed using this algorithm. We are searching for other suitable parameters and extend the algorithm to handle more shapes. The growing factor of plants is worth consideration because there are many well-known theories in botany such as phototropism, gravitoprism, photoperiodism and plant hormone. The particle system is one of the tools that is useful for simulating interacting complex system of nature.

Fig. 10. Comparing the real leaves (top) with the leaves generated by particle systems (bottom).

Fig. 11. Integrated model of leaves and trees.

References

1 Y. Rodkaew, S. Siripant, C. Lursinsap, P. Chongstitvatana, T. Fujimoto and N.Chiba, "Modeling Leaf Shapes using L-system and Genetic Algorithms", NICOGRAPH2002, 2002.

2 Y. Rodkaew, S. Siripant, C. Lursinsap and P. Chongstitvatana, "An algorithm for generating vein images for realistic modeling of a leaf", Computational Mathematics and Modeling, CMM2002, 2002.

3 W. T. Reeves, "Particle Systems - A Technique for Modeling a Class of Fuzzy Object", ACM Transactions on Graphics 2(2), pp. 91-108 (April 1983).

4 W. T. Reeves, "Approximate and problabilistic algorithms for Shading and Rendering Structured Particle Systems", Proceeding of SIGGRAPH'85 (Sans Francisco, California, July 22-26, 1985), In Computer Graphics Proceedings, Annual Conferrence Series, 19(3), pp. 313-322, ACM SIGGRAPH 1985.

5 P. Prusinkiewicz, M. James and R. Mech, "Synthetic topiary", Proceedings of SIGGRAPH 94 (Orlando, Florida, July 24-29, 1994). In Computer Graphics Proceedings, Annual Conference Series, 1994, ACM SIGGRAPH, pp. 351-358.

6 I. Shlyakhter, M. Rozenoer, J. Dorsey and S. Teller, "Reconstructing 3D Tree Models from Instrumented Photographs", IEEE Computer Graphics and Applications, May/June 2001.

7 P. Prusinkiewicz, L. Mundermann, R. Karwowski, B. Lane, "The use of positional information in the modeling of plants", pp. 289-300, ACM SIGGRAPH 2001.