Non-Photorealistic Rendering of Plants and Trees

C. Campos[†]*, R. Quirós, J. Huerta, M. Chover, J. Lluch, R. Vivó*

† Sección Departamental de Lenguajes y Sistemas Informáticos. Universidad de Vigo
Edificio Politécnico, Campus de As Lagoas, 32004 Orense, Spain
e-mail: ccampos@trevinca.ei.uvigo.es

ABSTRACT

The rendering technique described in this work allows non-photorealistic rendering of plants and trees. Our method is based on the general method proposed by Barbara Meier in 1996. RL-Systems are used to model a tree, allowing the generation in a single step of both the tree geometry and the set of brush strokes needed for later rendering. It can be applied to traditional animation, and to project presentation in architecture and interior design. The main advantage of our method is the integration of the processes of particle generation and geometry generation, facilitating the random perturbation of the model parameters, and the combination of elements to produce several levels of detail.

1. INTRODUCTION

The modeling of plants and trees has received an increasing attention in the last few years. The work of Prusinkiewicz and Lindenmayer [8], Reeves and Blau [11], and Lintermann and Deussen [3] has led to the development of integrated systems for modeling and rendering natural environments with plants and trees. These systems have been applied to architecture, interior design, and gardening and landscaping simulations. Our interest in this field focuses on a different application: robot training for operation in agricultural environments, where rendering techniques for trees play a highly relevant roll.

The recent emergence of non-photorealistic rendering techniques [12] introduces alternatives to conventional rendering that aim at increasing the expressiveness of the final images. The most relevant techniques in this field are used to match the look of traditional animation films (usually called *toon shading*) [2], and the look of drafts common in building and interior design presentations [13]. In all these applications, plants and trees are found repeatedly. Therefore, specific techniques are necessary for proper plant and tree rendering.

Our method is based on the general method proposed by Meier [7]. RL-Systems are used to model a tree, allowing the generation in a single step of both the tree geometry and the set of brush strokes needed for later rendering. This method has several advantages over those using surface models for the representation of trees [1]. In the following sections we summarize the techniques involved in rendering and modeling a tree. We also outline possible applications of our method, and present some of our experimental results. We conclude this paper with some directions for future work.

2. PAINTERLY RENDERING

The *painterly rendering* technique used in this work is based on a method proposed by Meier in 1996 [7]. Meier's method uses the following algorithm to obtain the final image of the model (extracted literally from [7]):

```
create particles to represent geometry
create a reference picture using
conventional rendering techniques
sort particles by distance from viewpoint
for each particle, starting with furthest
from viewpoint
   transform particle to screen space
   determine brush stroke attributes
   composite brush stroke into paint buffer
end
```

This method proposed by Meier has been improved by several researchers to support real-time rendering [5] and multiresolution modeling [6]. In all these works, stroke distribution in 3D space is based on a set of particles associated with the geometry. Particle generation is carried through the triangulation of surface patches. Such triangulation is inefficient for plant and tree rendering applications. Instead, we propose integrating the particle distribution into the tree modeling procedure. We do it using an RL-System [10] as described in the following section. The integration of the modeling procedure and the stroke particle system generation makes the process easier, taking great advantage of the hierarchical structure of plants and trees.

This work was supported by grant TIC1999-0510-C02-02 (CICYT, Ministerio de Educación y Ciencia, SPAIN)

3. TREE MODELING

We use an RL-System [10] for tree modeling. They are an improvement of Parametric L-Systems introduced by Prusinkiewicz [9], where a set of random variables is associated to the system. RL-Systems are a particular case of rewriting system whose final string representing the tree geometry is obtained by a derivation process starting from the axiom. The final string obtained is interpreted by a constructing 3D turtle, whose position and orientation in 3D space changes according to specific symbols, placing branches and leaves by means of a shape instancing process. L-Systems have been successfully applied to multiple plant and tree modeling systems [8][9][4].

The distribution of stroke particles is done by the same RL-System used for geometry modeling. To achieve this goal, a last derivation step is added to the system that obtains a set of related stroke particles for every instantiable element, i.e., for every branch and leaf. This technique has multiple advantages over the approach based on surface patches [1]. The most important advantages are:

- It integrates both generation processes, the tree geometry generation and the generation of the set of stroke particles.
- It supports the addition of local perturbations to the particles using the random variables included in the system.
- It simplifies the generation of several levels of detail for the representation of every instantiable element.

4. EXPERIMENTAL RESULTS

For the test and validation of the algorithm we have used a ternary tree model. The system starts from an initial branch with a bud in its final end. In the following derivations, each existing bud derives in three branches with new buds in their ends. The penultimate derivation generates buds of different types that will give place to the final leaves of the tree. The last derivation generates the instantiable forms for leaves and branches.

The final RL-System used to render the realistic image of reference is described hereinafter:

```
SYSTEM
DECLARATIONS
   CUniform d 130.0 145.0
   CUniform b  15.0 35.0
   CUniform r 0.8 1.4
   CUniform h 3.0 4.0
   CUniform k 50.0 80.0
   CUniform x 6.0 10.0
   CUniform m 170.0 190.0
   CConstant v 0.707
   CConstant s 1.4144
AXIOM
   BR{40.0,0.7}/{63.0}A{}
RULES
   A{}:NumIt<7#Push{}&{b}BR{15,0.7}A{}Pop{}
   /{d}Push{}&{b}BR{15.0,0.7}A{}Pop{}
   /{d}Push{}&{b}BR{15.0,0.7}A{}Pop{}
   A{}#Push{}&{b}BR{5.0,0.5}B{}Pop{}
   /{d}Push{}&{b}BR{5.0,0.5}B{}Pop{}
   /{d}Push{}&{b}BR{5.0,0.5}B{}Pop{}
   B{}#/{m}BR{3.0,0.5}Push{}&{k}LF{}Pop{}
   /{m}BR{3.0,0.5}Push{}&{k}LF{}Pop{}
   /{m}BR{3.0,0.5}Push{}&{k}LF{}Pop{}
   &{-k}LF{ }
   BR{l,w}:NumIt<7#BR{l*r,w*s}
   BR{l,w}:NumIt=9#Cylinder{l,w,w*v}F{l}
   LF{}#Instance{0,x,x,x}
END
```

In the realized implementation, the module Cylinder {} generates an instantiable object formed by brush strokes distributed on a cone shaft. The module Instance {} generates a single brush stroke to represent a leaf.

The derivation of the previous system produces the realistic image of the tree that has been used as reference image, as is shown in the figure 1.

Figure 1. Original image of sample tree

The use of random variables allows obtaining variations of the same specie of tree. To generate some of the test images a variation of the previous system has been used, where the productions that give place to the leaves are eliminated .

The previous image, together with the particles distribution, allows calculating the position, orientation, size and color of each one of the brush strokes. We have used as brush stroke a polygon to which a channel alpha texture is applied. The color of the polygon is obtained from the image of reference, determining the color of the pixel associated to the center of the brush stroke. In the calculation of the parameters associated to each stroke, random variables had been used, producing the effect of hand-made painting.

The brushes used for the experiment had been extracted from the plug-in *Impressionist*, and simulate the real painting materials effect. The figure 2 shows two of these brushes simulating, chalk and watercolor.

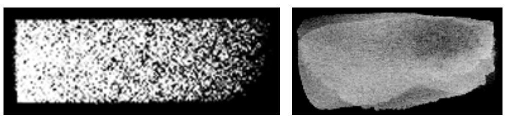

Figure 2. Sample brushes

The painting process traverses the strokes list, beginning with the far away ones. The final results are shown in the figures 3 and 4. In each one of the figures a different brush it has been used, without varying the parameters of the brush strokes. The tree of the figure 5 has been generated using a variation of the RL-System, removing the productions that give place to the leaves.

Figure 3. Rendering with chalk brush

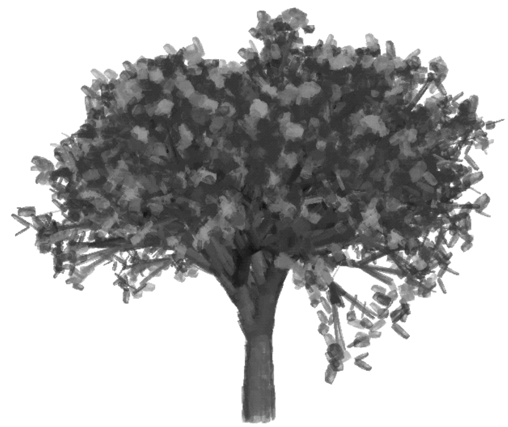

Figure 4. Rendering with watercolor brush

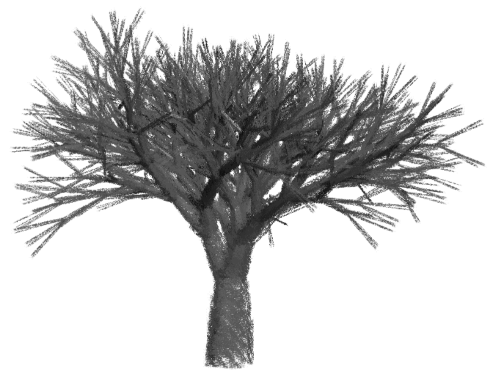

Figure 5. A variation of the RL-System

5. APLICATIONS

Non-photorealistic rendering of plants and trees has both pure aesthetic applications and applications to traditional animation, videogames, architecture, and interior design. Non-photorealistic rendering can also be used for geometric simplification without affecting image rendering quality. As an alternative to conventional rendering techniques, it is also applicable to real-time simulation and training. This application requires speed improvements and a better management of levels of detail.

6. CONCLUSIONS AND FUTURE WORK

The rendering technique described in this work allows non-photorealistic rendering of plants and trees. It can be applied to traditional animation, and to project presentation in architecture and interior design. The main advantage of our method is the integration of the processes of particle generation and geometry generation. The management of the symbolic entities – imposed by the features of RL-

Systems – also facilitates the random perturbation of the model parameters, and the combination of elements to produce several levels of detail. Future work in this area will concentrate on the application of our method to conventional rendering. Our ultimate goal is to use it in simulation applications for agricultural robot training. To achieve this goal it is essential that we guarantee real-time visualization using acceleration techniques and a better approach to level-of-detail generation and management.

6. REFERENCES

[1] O. Deussen, T. Strothotte, "Computer-generated pen-and-ink illustration of trees", Proceedings of SIGGRAPH 2000, pag. 13--18

[2] G. Katanics, T. Lappas, "Deep Canvas: Integrating 3-D Painting and Painterly Rendering", 2000 http://disney.go.com/disneycareers/animation/technology/deep_canvas.html

[3] B. Lintermann, O. Deussen, "Interactive modeling of plants", IEEE Computer Graphics & Applications 19(1), pag. 56--65, January/February 1999

[4] J. Lluch, R. Quiros, B. Castellanos, "Aplicaciones de los Sistemas Paramétricos Aleatorios a la generación de árboles y plantas herbáceas", III Congreso Español de Informática Gráfica (in Spanish), Granada, 1993, June, pag. 195--207

[5] L. Markosian, M. Kowalski, S. Trychin, L. Bourdev, D. Goldstein, J. Hughes. "Real-Time Nonphotorealistic Rendering", *Computer Graphics (Proceedings of SIGGRAPH '97)*, August, 1997

[6] L. Markosian, B. Meier, M. Kowalski, L. Holden, J. Northrup, J. Hughes, "Art-based Rendering with Continuous Levels of Detail", Proceedings of NPAR 2000, Annecy, France, June 2000

[7] B. Meier, "Painterly Rendering for Animation", Proceedings of the 23rd annual conference on Computer graphics, August 4 - 9, 1996, New Orleans, LA USA, pag. 477--484

[8] P. Prusinkiewicz, A. Lindenmayer, "The algorithmic beauty of plants", New York, Ed. Springer-Verlag, 1990

[9] P. Prusinkiewicz, J. Hanan, "Visualization of botanical structures and processes using parametric L-systems", Scientific visualization and Graphics simulations, Ed. D. Thalmann, 1990, pag. 183--201

[10] R. Quirós, J. Lluch, M. Chover, R. Vivó, "Geometric substitution using Random L-Systems", Computers & Graphics, vol. 20, no. 5, 1996

[11] W. T. Reeves, R. Blau, "Approximate and Probabilistic Algorithms for Shading and Rendering Structured Particle Systems", Computer Graphics (SIGGRAPH '85 Proceedings), held in San Francisco CA, 1985, July, vol. 19, pag. 313--322, 1985

[12] C. Reynolds, "Stylized Depiction in Computer Graphics, an annotated survey of online resources", http://www.red3d.com/cwr/npr/

[13] R. Richens, S. Schofield, "Interactive Computer Rendering", Architectural Research Quarterly, Volume 1(1), Autumn 1995