The Virtual Forest

Daniel Wesslén*
University of Gävle

Stefan Seipel†
University of Gävle

Figure 1: Screenshots from two different forest scenes.

1 Introduction

Offline rendering of computer generated trees has received much research in the last decade, resulting in some impressive imagery [Deussen et al. 1998]. Realtime rendering for simple cases has also been explored, both for highly detailed trees viewable from a distance [Weber and Penn 1995] and the textured low-polygon versions seen in current games.

We present a system that renders an animated synthetic forest in realtime for interactive walkthroughs. Up close, trees are detailed enough that individual leaves are animated, while continuous level-of-detail adjustments keep processing load low when viewed from afar.

2 Creation

Tree creation is based on the algorithm described by Weber and Penn [1995] but adapted to better suit nordic trees such as spruce and pine, cleaned from many implementation artifacts, and generally made easier to handle for tree designers and developers.

The most important enhancement to tree creation is the addition of a health parameter, which is required for modeling of realistic pine trees. Moderate reductions in health cause leaves to be randomly omitted from the creation process. Larger reductions remove branches as well.

3 Display

A continuous level-of-detail algorithm keeps rendering running smoothly independently of the distance from which trees are viewed. Up close, individual leaves and branches are rendered as textured primitives. When the camera moves further away these are gradually replaced by simple points and lines. Moving even further, they are removed when the average projected size of the primitives fall below one pixel.

Display differs from [Weber and Penn 1995] in two important aspects—the original algorithm had to traverse the entire tree to select which parts of it were to be rendered, and leaves were singly-colored complex shapes. We use prearranged vertex buffers and select parts of the tree by computing index ranges in these. Leaves are rendered as textured quads, improving storage efficiency and detail. Shapes may be applied to the leaves by alpha testing.

Entire trees are animated in a simple swaying motion around the base by the vertex shader and leaves are individually animated in a rotation around their parent stem. While this is simplistic, the results are pleasing.

Shaders also increase the diversity of the forest by allowing slight changes to the size and color of tress instanced from the same base model and by reducing memory requirements due to per-tree rather than per-vertex storage of some parameters.

4 Results

Figure 1 contains screenshots from our simulation, these were captured at 1600×1200, running in excess of 50 frames per second on a Radeon 9800 XT. Since essentially all work is performed by the graphics card the CPU is irrelevant during rendering.

The system allows for realtime frame rates on affordable consumer cards for relatively complex scenes.

References

DEUSSEN, O., HANRAHAN, P., LINTERMANN, B., MĚCH, R., PHARR, M., AND PRUSINKIEWICZ, P. 1998. Realistic modeling and rendering of plant ecosystems. *Computer Graphics 32*, Annual Conference Series, 275–286.

WEBER, J., AND PENN, J. 1995. Creation and rendering of realistic trees. In *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, ACM Press, 119–128.

*e-mail: dwn@hig.se
†e-mail: ssl@hig.se