Real-Time Rendering of Vegetation and Trees in Urban Environments

Stephan MANTLER, Gerd HESINA, Stefan MAIERHOFER, and Robert F. TOBLER

VRVis Research Center,
Donau-City Str. 1/3, Vienna, {mantler|hesina|maierhofer|tobler}@vrvis.at

1 INTRODUCTION

Vegetation and Trees in Urban settings are getting ever more important due to the increased focus on high quality living environments. Urban planning therefore needs to take vegetation into account at every stage. The availability of cheap 3D graphics hardware makes it possible to consider vegetation and tree rendering in general urban visualization systems.

In our System we have employed a number of different vegetation visualization methods, including point-based rendering, impostors and billboard clouds. Based on these experiences we have designed an urban rendering system that can deal with thousands of trees, facilitating real-time visualization of urban recreation areas such as parks and green-belts. Based on the application different techniques are used in order to provide optimal performance in each situation. Thus it is possible to render single trees with adequate detail for walk-throughs as well as whole woods for fly-over applications.

Such Real-Time walk-throughs and fly-overs of existing or planned urban-zones including huge amounts of vegetation can help urban planners in a variety of situations, e.g. visibility of buildings and street signs or placement of future vegetation.

Figure 1: Improved Billboard-Trees in an Urban Setting.

2 PROBLEM STATEMENT

Rendering trees and vegetation in the context of an urban scene represents a unique challenge: the main purpose of urban models is the geometric representation of buildings and streets and thus most effort has been targeted at these objects. Vegetation and Trees have been limited to very simple representations just to give a hint of their position and type. Nevertheless a complete urban enviornment without vegetation and trees is very unnatural and does not adequately represent a city. Thus it is necessary to render nicely looking trees without spending a huge portion of the available rendering budget just to make these trees look good.

3 RENDERING METHODS FOR VEGETATION AND TREES

In recent years three main methods for rendering vegetation have been developed: billboard-trees, impostor-based trees, and point-based trees.

3.1 Billboard trees

The simplest form of this tree-representation consists of two quadrilaterals that are positioned at a right angle to each other with two views of the tree from orthogonal viewpoints put into the texture of these quadrilaterals. This can be extended to branches and twigs by adding additional quadrilaterals with associated textures, and results in a representation of a tree by a small number of polygons [JAKULIN, MEYER et al., SPEEDTREE].

3.2 Impostor-based trees

Whereas billboard trees are represented by a small number of polygons (e.g. 2) that are fixed in their orientation, impostors are single quadrilaterals with texture that are always oriented towards the viewer. The earliest and simplest impostors were single views of trees that were always reoriented to face the viewer. As this can only be viewed as a hint for a trees existence, and does not adequately represent its shape, modern impostor-based systems change the displayed view depending on the orientation. Thus an impostor system is a dynamic rendering sytem that needs to regenerate the texture of each impostor that has become too inaccurate, as the vieweing angle and the original rendering angle have diverged. [SCHAUFLER & STÜRZLINGER]

3.3 Point-based trees

A different approach to rendering trees and vegetation is the use of points instead of triangles. This is based on the observation that the screen size of triangles that are rendered to represent objects like leaves or twigs is often smaller than a single pixel. By just using a single point instead of a triangle, it is only necessary to transform and work on a single 3-dimensional coordinate instead of three (for each vertex of the triangle) [WEBER & PENN, DEUSSEN et al.].

This approach can be extended to represent larger portions of the tree with only single points, if they only occupy a small number of pixels on the screen. Such a level-of-detail approach can be naturally included in a point-based renderer and can significantly improve rendering speeds.

4 IMPROVED TREES FOR URBAN ENVIRONMENTS

Based on the presented technologies we have employed a number of methods for rendering trees and vegetation. In the following sections we will give a more detailed description of how these techniques can be used, and their respective advantages and drawbacks.

4.1 Improved point-based trees

Representing the leaves of a whole tree by points leads to one specific performance problem: due to the density of leaves in a tree, a large number of points that are drawn onto the screen will be later covered by closer points an thus a signifcant amount of rendering time is wasted on invisible parts of the geometry.

Based on this observation, a method for generating multiple view-dependent representations for each tree is a promising approach for speeding up rendering times.

Our algorithm for high-performance point based trees uses this idea and therefore has two stages:

Pre-processing stage: In this stage the tree is rendered into a buffer from a number of different view directions that are grouped into multiple sets (around 20 to 30). For each set of directions statistics are maintained for each point that has to be rendered. Based on the number of pixels a specific point has covered during rendering all the directions in a set, the point is assigned an importance. The points are then sorted according to this importance. Thus for each set of view directions, a single sorted list of points is generated that represents the tree as seen from any of the included directions.

Rendering stage: When rendering a tree, the actual view direction is classified with respect to the view direction sets used in preprocessing. The set that most closely represents the actual viewing direction is then used to render the tree. It is now possible to allocate a rendering budget for rendering this tree, by specifying the number of points that should be rendered. Due to the sorting of the point-list according to importance, the most important points are rendered as a representation of the tree.

This tree-rendering algorithm is adequate for rendering a fair number of trees (a few hundred to a few thousand) at medium to far distance as can be seen in figure 2 and 3. Due to the preprcessing, the actual number of points that needs to be rendered is significantly smaller than the total number of points in the tree resulting in a significant speed up compared to standard point-based trees. In our tests the speed-ups were in the range between 10 times and 20 times.

A drawback of point-based trees is their poor visual appearance when they are viewed from close up: they split up into individual points and lose the appearance of a tree. In an urban setting they can thus only be used for trees that are not viewed from close up.

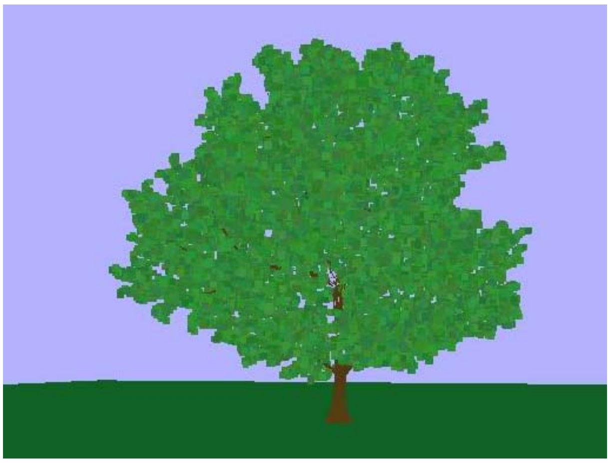

Figure 2: A single point-based tree.

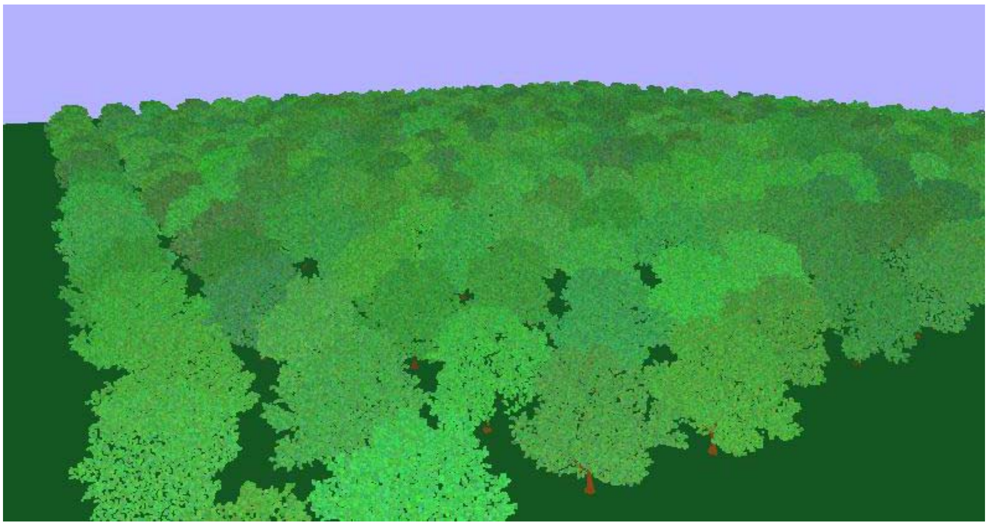

Figure 3: 250 Point-based trees rendered in real-time.

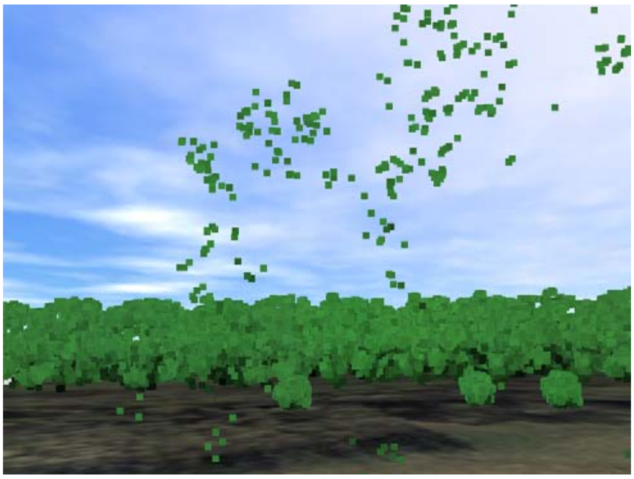

Figure 4: A point-based tree falls apart when viewed from to close-up.

4.2 Improved billboard trees

Billboard trees, although based on a simple algorithm are complicated by one fact: choosing adequate quadrilaterals for representing an operation that cannot be easily automated. Therefore these types of trees were often used with representation quadrilaterals specified by hand. Recently a general algorithm for automatically generating billboard representations of arbitrary objects has been introduced by Decoret et al. [DECORET et al.]: the object is transformed into a dual space where each plane is represented by a single point. Clusters in this dual space are then successively represented by a single quadrilateral until the complete object has been covered.

Directly applying this algorithm to trees results in inadequate representations, as the visual importance of the different features of a tree is not evident in the geometry: a trunk is visually significantly more important than a few branches that are hidden by leaves although these leaves might be larger than the trunk in total.

In order to overcome these problems we introduced a number of heuristics for placing the quads for representing the whole tree. This includes special treatment of the trunk, post-optimization of the set of quadrilaterals for representing the tree, as well as preference for vertically oriented quads in order to get good representations for walk-throughs.

Figure 5: Walkthrough scenario with improved billboard trees.

The improved billboard trees are still very quick to render, as they require only 10 to 40 textured triangles per tree, but avoid the extreme problems of point based trees when viewed from close-up. There are still visible problems when the textured triangles that represent a tree are viewed edge on as seen in some examples in figure 5, but overall the visual impression of these trees is adequate for both fly-overs and walk throughs (figure 1 and 5).

5 CONCLUSION AND FUTURE WORK

We demonstrated the use of different rendering methods in order to improve rendering speeds and rendering quality for rendering of vegetation and trees for urban environments. Based on different requirements these methods can be used to achieve adequate quality in a variety of cases.

Nevertheless, there are still a huge number of cases where the realism of the presented rendering methods is inadequate. Specifically if any of the rendered tree or vegetation models is viewed at a close distance a variety of artefacts become obvious. In order to overcome these problems we are currently developing a new multiresolution representation of vegetation for rendering vegetation and trees at any resolution and distance.

6 ACKNOWLEDGEMENTS

This work has been done at the VRVis research center, Vienna, Austria (http://www.vrvis.at), which is partly funded by the Austrian government research program Kplus. Thanks to the City of Vienna for the use of their city model in this research project.

7 REFERENCES

DECORET, X, DURAND, F, SILLION, F., and DORSEY, J: Billboard Clouds for Extreme Model Simplification. SIGGRAPH 2003 Proceedings, ACM Press, 2003.

DEUSSEN, O, COLDITZ, C, STAMMINGER, M, and DRETTAKIS, G: Interactive Visualization of Complex Plant Ecosystems. IEEE Visualization 02 Conference Proceedings, Oct. 2002.

JAKULIN, A: Interactive Vegetation Rendering with Slicing and Blending. Eurographics 2000 Conference Proceedings, 2000.

MEYER, A, NEYRET, F, and POULIN, P: Interactive Rendering of Trees with Shading and Shadowing. Eurographics Workshop on Rendering Conference Proceedings, Springer Verlag Wien-New York, Jul. 2001.

SCHAUFLER, G, and STÜRZLINGER, W: A Three-Dimensional Image Cache for Virtual Reality. Computer Graphics Forum, Volume 15, Issue 3, p. 227, Aug. 1996.

SPEEDTREE: Interactive Data Visualization Inc. Product homepage. http://www.idvinc.com/speedtree/

WEBER, J, and PENN, J.: Creation and Rendering of Realistic Trees. SIGGRAPH '95 Conference Proceedings, pp. 119-128, Aug. 1995.