A Demonstration of MobiTree: Progressive 3D Tree Models Streaming on Mobile Clients

Andra Doran, Sébastien Mondet, Romulus Grigoras, Géraldine Morin, Wei Tsang Ooi, Frédéric Boudon

A Demonstration of MobiTree: Progressive 3D Tree Models Streaming on Mobile Clients

Andra Doran
University of Toulouse, France
doran.andra@gmail.com

Sebastien Mondet
University of Toulouse, France
smondet@n7.fr

Romulus Grigoras
University of Toulouse, France
grig@n7.fr

Geraldine Morin
University of Toulouse, France
morin@n7.fr

Wei Tsang Ooi
National University of Singapore
ooiwt@comp.nus.edu.sg

Frederic Boudon
CIRAD, Montpellier, France
frederic.boudon@cirad.fr

ABSTRACT

We demonstrate MobiTree, a system we built that allows progressive streaming and rendering of 3D tree models on a mobile phone. MobiTree enables user to retrieve not only typical media describing a species (such as text and image), but also a 3D model that yields detail information about the structure of the branches and the foliage. MobiTree adopts our previous proposed progressive representation to speed up display of trees at the mobile client, trading off latency and quality. Progressivity also allows MobiTree to flexibly adopt the level of details to the capability of the mobile devices.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous

General Terms

Design, Human Factors

Keywords

Streaming, Plant Models, Progressive Coding, Progressive Transmission, Mobile Applications

1. INTRODUCTION

We demonstrate MobiTree, an application for remote access to a botanical database, offering nature discovery on mobile devices. In the application we envision, when a user visits a botanical garden, he can request detail information of a particular tree (e.g., by taking a snapshot of the tree's bar code). Details such as current age and origins can then be retrieved for display on the mobile devices. Further, the user can ask "what-if" questions about the growth of the tree. For instance, the user can ask what would the tree look like at 7 years of age if it is grown in a tropical climate. A server could dynamically generate the tree by simulating the growth of the tree under the given conditions [6], and stream the result as a 3D model for displaying and interactive browsing on the mobile client.

This demo focuses on streaming 3D models of plants and its rendering on mobile clients, which is the novel feature of MobiTree. Figure 1 shows a snapshot of our application. We chose to use a 3D representation because it is compact. Using our previously proposed representation [2], a plant model consumes only tens of KB of storage. Further, a 3D representation is flexible, allowing users to view the plants from different view points, and zoom in to examine the details. The 3D representation can be rendered progressively, allowing the application to adapt to the capability of the mobile device. Progressivity also allows users to view a coarse version of a plant while the details are being sent, reducing waiting time for the user. An alternative would be image-based rendering. This technique may be able to model a 3D object with arbitrary view points using elaborate technique, but the progressivity of this model is not direct, and the overall data size is greater than the 3D solution.

Figure 1: Snapshot of MobiTree.

We organize the rest of this paper as follows. We review other botanical applications in Section 2. Section 3 gives an overview of progressive representation of plants [2]. We discuss how foliage is rendered in Section 4 and the technical challenges in rendering plant on mobile devices in Section 5. The demonstration and further applications are described in Sections 6 and 7.

2. RELATED WORK

There are several previous efforts in developing a mobile information system for botanical application. In the electronic field guide project, White et al. [4, 5] and Agarawal et al. [1] propose to use a mobile augmented reality system for botanists on the field to identify existing and new species. Their system uses computer vision techniques to identify a botanical plant through leaves. Zhou and Schneider [7] presented a system for retrieving multimedia learning materials for a botanical garden based on the user's location. These previous works do not support 3D representations of plants and focus their research on the retrieval aspect of the plants.

Our system for streaming of 3D plant models could easily be integrated with these systems to enrich the user experience.

3. PROGRESSIVE REPRESENTATION OF PLANTS

We now briefly describe the progressive model of plants that we use in our application. We first focus on the branching structure of the plant. Each branch is compactly represented as a generalized cylinder, described by (i) an axis curve, modeled as a Bezier curve of degree d, and (ii) a radius along the branch, modeled as a 2D Bezier curve. The branches are organized as a hierarchical data structure according to the structure of the plant. All children branches borne by a parent branch form the parent-children relationship in the data structure. The position u on which each child branch attaches on the parent branch is stored in the structure. The radius of the branch is represented along the branch as a Bezier curve of degree m with control points $(u_i; r_i)$, where u_i, $1 \leq i \leq m$, is an increasing sequence in the interval [0; 1] that determines the location of the branch, and r_i, $1 \leq i \leq m$, is the branch radius for the corresponding given location.

Each branch, described by control points, is then transformed to a normalized representation such that the end points are at $(0, 0, 0)$ and $(0, 0, 1)$. The reverse transformation to obtain the original branch, along with the attachment point u, is called an *instance* of the branch. The normalized branches are then differentially coded as follows. We first group similar Bezier curves together, compute an average curve (called *model*) for each group, and encode the differences between the control points of a Bezier curve and the average curve (called *detail vectors*). Grouping similar curves together reduces the differences, which can then be quantized with fewer bits. For rendering, we must first decode the model, then the instance, and finally add the details for obtaining the original branch.

Progressivity comes in two ways. First, there are increasingly more branches, as more instances are received. Second, the position of the branches becomes increasingly accurate, as more detail vectors are received. Note that adding detail vectors to a branch causes the position of all children branches to be recursively re-calculated.

For this application, we have used various plant models, digitized using a Polhemus 3Space Fastrack electromagnetic device [3].

4. RENDERING OF FOLIAGE

Leaves have been added to the model, but unlike suggested in [2], the leaves are not attached on the branches, but treated in an independent structure. This independence allows to start rendering leaves, even if the actual bearing branch has not been reached yet at the current partial representation. This choice has been made because of the visual importance of leaves compared to branches.

To render the leaves, we draw a polygon at the specified location and orientation, and we map a texture of the leaf with transparency.

5. ADAPTING TO MOBILE DEVICES

Rendering a 3D plant model on a mobile device is challenging due to its relatively low computational capability and its power constraints. Few mobile devices benefits from graphics hardware acceleration. Most of mobile devices do not have a floating point unit but rather work with fixed point arithmetic. Finally, we need to present the content on a small screen at a low resolution. The problem is to find a compromise between computation precision, computation speed, energy consumption, and rendered quality.

The progressivity of our plant models allows us to easily trade-off between the resource usage and rendered quality. On a low end phone, we can limit either the number of total branches, or the number of details transmitted and rendered, resulting in a simpler model of the tree. We prioritize the importance of binary chunks based on the size of the branch, ensuring reasonably visual quality even when a small amount (down to 1%) of branches is rendered.

To further reduce the computational complexity, we always render the instance together with its detail vectors. This method ensures that once rendered, the position of a branch is accurate, therefore preventing recalculation of the branch position and recursive recalculation of its descendant branches.

6. DEMONSTRATION

In the technical demonstration, participants can view different 3D models of trees progressively and interact with the model, by rotating and zooming, to explore various aspects of the tree. Users can also retrieve botanical facts about the species. MobiTree runs on various Java ME-enabled mobile phones.

7. OTHER APPLICATIONS

Besides the botanical garden scenario we described, the progressive streaming and rendering of plants are useful in other applications. One such example is nature-oriented educational games, that would bring together technology and nature for children. The system can also be integrated into electronic field guides for botanist, as proposed by White et al [5]. We also envision that such system can be useful for architects, civil engineers, and landscape designers, where knowledge of how trees grow in a given environment, especially the future shape, is critical to the trees' effectiveness to provide shades, serve as noise barriers, and beautify the cityscape.

8. REFERENCES

[1] G. Agarwal, P. Belhumeur, S. Feiner, D. Jacobs, J. Kress, R. Ramamoorthi, N. Bourg, N. Dixit, H. Ling, D. Mahajan, R. Russell, S. Shirdhonkar, K. Sunkavalli, and S. White. First Steps Toward an Electronic Field Guide for Plants. *Taxon*, 55:597–610, 2006.

[2] S. Mondet, W. Cheng, G. Morin, R. Grigoras, F. Boudon, and W. T. Ooi. Streaming of Plants in Distributed Virtual Environments. In *Proceedings of ACM Multimedia '08*, pages 1–10, Vancouver, Canada, October 2008.

[3] H. Sinoquet, P. Rivet, and C. Godin. Assessment of the three-dimensional architecture of walnut trees using digitising. *Silva Fennica*, 31(3):265–273, 1997.

[4] S. White, S. Feiner, and J. Kopylec. Virtual vouchers: Prototyping a mobile augmented reality user interface for botanical species identification. In *IEEE Symposium on 3D User Interfaces*, pages 119–126, Alexandria, VA, 2006.

[5] S. White, D. Marino, and S. Feiner. Designing a mobile user interface for automated species identification. In *Proceedings of the ACM CHI'07*, pages 291–294, San Jose, CA, 2007.

[6] H. Yan, M. Kang, P. De Reffye, and M. Dingkuhn. A dynamic, architectural plant model simulating resource-dependent growth. *Annals of Botany*, 93(5):591, 2004.

[7] R. Zhou and G. Schneider. Location-based botany guide: A prototype of web-based tracking and guiding. In *Proceedings of Mobilware'09*, pages 72–86, Berlin, Germany, April 2009.