

**Luiz Gustavo Bustamante  
Magalhães**

**Multi-resolução de Geometria  
de Terrenos Armazenados em  
Memória Secundária**

**DISSERTAÇÃO DE MESTRADO**

**DEPARTAMENTO DE INFORMÁTICA  
Programa de Pós-graduação em  
Informática**

Rio de Janeiro  
Dezembro de 2005

PONTIFÍCIA UNIVERSIDADE CATÓLICA  
DO RIO DE JANEIRO



**Luiz Gustavo Bustamante Magalhães**

**Multi-resolução de Geometria de Terrenos  
Armazenados em Memória Secundária**

**Dissertação de Mestrado**

Dissertação apresentada como requisito parcial para obtenção do grau de Mestre pelo Programa de Pós-graduação em Informática do Departamento de Informática da PUC-Rio

Orientador: Prof. Waldemar Celes Filho

Rio de Janeiro  
Dezembro de 2005



**Luiz Gustavo Bustamante Magalhães**

**Multi-resolução de Geometria de Terrenos  
Armazenados em Memória Secundária**

Dissertação apresentada como requisito parcial para obtenção do grau de Mestre pelo Programa de Pós-graduação em Informática do Departamento de Informática do Centro Técnico Científico da PUC-Rio. Aprovada pela Comissão Examinadora abaixo assinada.

**Prof. Waldemar Celes Filho**

Orientador

Departamento de Informática — PUC-Rio

**Prof. Marcelo Gattass**

PUC-Rio

**Prof. Paulo Cezar Pinto Carvalho**

IMPA

**Prof. José Eugênio Leal**

Coordenador Setorial do Centro Técnico Científico —  
PUC-Rio

Rio de Janeiro, 13 de Dezembro de 2005

Todos os direitos reservados. É proibida a reprodução total ou parcial do trabalho sem autorização da universidade, do autor e do orientador.

### **Luiz Gustavo Bustamante Magalhães**

Graduou-se em Engenharia de Computação na PUC–Rio em 2002. Desde 2001 trabalha no laboratório de Computação Gráfica da universidade (TecGraf).

#### Ficha Catalográfica

Magalhães, Luiz Gustavo Bustamante

Multi-resolução de geometria de terrenos armazenados em memória secundária/ Luiz Gustavo Bustamante Magalhães; orientador: Waldemar Celes Filho. — Rio de Janeiro : PUC–Rio, Departamento de Informática, 2005.

v., 62 f: il. ; 30 cm

Dissertação (mestrado) - Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Informática.

Inclui referências bibliográficas.

1. Informática – Teses. 2. Visualização de terrenos. 3. Multi-resolução. 4. Gerenciamento de memória. 5. Predição de movimento de câmera. I. Celes Filho, Waldemar. II. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Informática. III. Título.

CDD: 004

## Agradecimentos

Aos meus pais e à minha família, pelo incentivo e pelo apoio incondicional. Sem eles este trabalho não poderia ter sido realizado.

Ao meu orientador Waldemar Celes pelo apoio, pela dedicação, pela paciência, pela orientação impecável e por iniciar uma nova etapa da minha vida profissional.

À Beatriz, pelo carinho, pelo companheirismo e pela compreensão nos momentos em que foi preciso abdicar da diversão e do descanso.

Aos amigos, pelos momentos de diversão e por estarem sempre ao meu lado, mesmo quando estive ausente para a realização deste trabalho.

Aos amigos do TecGraf, pela ajuda na resolução de problemas, pelos momentos de descontração e pelo bom ambiente de trabalho.

Aos membros da banca, pelas críticas e sugestões.

Ao Marcelo Tílio, por incentivar e apoiar a realização deste trabalho.

Ao TecGraf e à PUC-Rio pelos auxílios concedidos.

## Resumo

Magalhães, Luiz Gustavo Bustamante; Celes Filho, Waldemar. **Multi-resolução de geometria de terrenos armazenados em memória secundária**. Rio de Janeiro, 2005. 62p. Dissertação de Mestrado — Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

A visualização de grandes terrenos é um assunto desafiador em Computação Gráfica. O número de polígonos necessário para representar fielmente a geometria de um terreno pode ser muito alto para ser processado em tempo real. Para resolver tal problema, utiliza-se um algoritmo de multi-resolução, que envia para o processador gráfico (*GPU*) somente os polígonos mais importantes, sem que haja perda da qualidade visual. A quantidade de dados é outro grande problema, pois facilmente excede a quantidade de memória *RAM* do computador. Desta forma, um sistema de gerenciamento de dados que não estão em memória principal também é necessário. Este trabalho propõe uma solução simples e escalável para visualizar a geometria de grandes terrenos baseada em três pontos-chave: uma estrutura de dados para representar o terreno em multi-resolução, um sistema eficiente de visualização e um sistema de paginação e predição dos dados. A estrutura de dados utilizada, assim como em outros trabalhos similares, é a *quadtree*. Esta escolha justifica-se pela simplicidade, além da eficiência e do baixo consumo de memória de uma implementação em vetor. Cada nó da *quadtree* representa um ladrilho do terreno. A implementação é dividida em duas linhas de execução (*threads*), uma para o gerenciamento dos ladrilhos e outra para a visualização. A linha de execução de gerenciamento de ladrilhos é responsável por carregar/remover ladrilhos para/da memória. Ela utiliza um mecanismo de predição de movimento da câmera para carregar ladrilhos que possam ser utilizados em um futuro próximo e remover ladrilhos que provavelmente não serão necessários. A linha de execução de visualização é responsável por visualizar o terreno, fazendo o cálculo do erro projetado, eliminando ladrilhos não visíveis e balanceando a estrutura de *quadtree* para eliminar falhas ou vértices T na superfície do terreno. A visualização pode ser feita de duas formas distintas: baseada no erro máximo tolerado ou na quantidade máxima de polígonos a ser processada.

## Palavras-chave

Visualização de terrenos; multi-resolução; gerenciamento de memória; predição de movimento de câmera.

## Abstract

Magalhães, Luiz Gustavo Bustamante; Celes Filho, Waldemar. **Multi-resolution of Out-of-Core Terrain Geometry**. Rio de Janeiro, 2005. 62p. MSc. Dissertation — Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

The visualization of large terrains is a challenging Computer Graphics issue. The number of polygons required to faithfully represent a terrain's geometry can be too high for real-time visualization. To solve this problem, a multi-resolution algorithm is used to feed the graphics processor only with the most important polygons, without visual quality loss. The amount of data is another important problem, as it can easily exceed a computer's RAM. Thus, a system to manage out-of-core data is also required. The present work proposes a simple and scalable solution to visualize the geometry of large terrains based on three key points: a data structure to represent the terrain in multi-resolution, an efficient visualization system and a data paging and prediction system. Similarly to other works, the system uses a quadtree data structure due to its simplicity, along with the efficiency and the low memory use of an array-based implementation. Each node of the quadtree represents a tile of the terrain. The implementation is divided in two threads, one to manage the tiles and the other for visualization. The tile-management thread is responsible for loading/unloading tiles into/from the memory. This thread uses a camera-movement prediction mechanism to load tiles that can be used in the near future and to remove tiles that probably will not be necessary. The visualization thread is responsible for viewing the terrain, computing the projected error, eliminating tiles that are not visible and balancing the quadtree structure in order to eliminate cracks or T-vertices on the terrain's surface. The visualization can be made by means of a fidelity-based or a budget-based approach.

## Keywords

Terrain visualization; multi-resolution; memory management; camera-movement prediction.

## Sumário

1	Introdução	<b>9</b>
1.1	Organização do Texto	10
2	Trabalhos Relacionados	<b>11</b>
3	Estrutura de Dados	<b>18</b>
3.1	Definição da Estrutura	18
3.2	Árvore Quaternária em Vetor	19
3.3	Pré-processamento	23
4	Algoritmo de Visualização	<b>27</b>
4.1	Visualização em Memória Principal	27
4.2	Visualização em Memória Secundária	36
5	Resultados Experimentais	<b>45</b>
5.1	Descrição dos Testes	45
5.2	Testes de Desempenho da Visualização	46
5.3	Testes de Desempenho do Gerenciamento de Ladrilhos	50
6	Conclusão	<b>56</b>
6.1	Trabalhos Futuros	57
	Referências Bibliográficas	<b>59</b>

”If debugging is the process of removing bugs, then programming must be the process of putting them in.”

**Dijkstra.**