



**Cristina Nader Vasconcelos**

**Algoritmos para Processamento de Imagens e  
Visão Computacional para Arquiteturas  
Paralelas em Placas Gráficas**

**Tese de Doutorado**

Tese apresentada ao Programa de Pós-graduação em Informática  
do Departamento de Informática da PUC-Rio como requisito  
parcial para obtenção do título de Doutor em Informática

Orientador: Prof. Marcelo Gattass

Rio de Janeiro  
Março de 2009



**Cristina Nader Vasconcelos**

**Algoritmos para Processamento de  
Imagens e Visão Computacional para  
Arquiteturas Paralelas em Placas Gráficas**

Tese apresentada como requisito parcial para obtenção do grau de Doutor 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. Marcelo Gattass**

Orientador  
Departamento de Informática – PUC-Rio

**Prof. Paulo Cezar Carvalho**

Co-Orientador  
Instituto Nacional de Matemática Pura e Aplicada – IMPA

**Prof. Waldemar Celes Filho**

Departamento de Informática – PUC-Rio

**Prof. Thomas Lewiner**

Departamento de Matemática – PUC-Rio

**Prof. João Luiz Dohl Comba**

Universidade Federal do Rio Grande do Sul - UFRGS

**Profa. Asla Medeiros e Sá**

Escola de Matemática Aplicada – FGV

**Prof. José Eugenio Leal**

Coordenador Setorial do Centro  
Técnico Científico

Rio de Janeiro, 23 de março de 2009

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

### **Cristina Nader Vasconcelos**

Cristina Nader Vasconcelos recebeu o título de Bacharel em Informática pela Universidade Federal do Rio de Janeiro (UFRJ) em 2003. Iniciou seus estudos de computação gráfica e processamento de imagens durante dois anos de iniciação científica no Instituto de Matemática Pura e Aplicada (IMPA). Tornou-se mestre em Ciência da Computação na área de Computação Gráfica pela Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio) em 2005. Participou de projetos de desenvolvimento de software e pesquisa em processamento de imagens e vídeo nas empresas BioLógica Sistemas e Midiarte. Em 2008, recebeu a bolsa de pesquisa DAAD, vinculada à pesquisa e estadia de três meses em universidade alemã (Max-Planck-Institut für Informatik - MPII). Atualmente participa do laboratório Tecgraf/PUC-Rio onde desenvolveu o estudo apresentado nesta tese.

#### Ficha Catalográfica

Vasconcelos, Cristina N.

Algoritmos para Processamento de Imagens e Visão Computacional para Arquiteturas Paralelas em Placas Gráficas / Cristina Nader Vasconcelos; orientador: Marcelo Gattass. — Rio de Janeiro : PUC-Rio, Departamento de Informática, 2009.

155 f. ; 30,0 cm

1. Tese (doutorado) - Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Informática.

Inclui referências bibliográficas.

1. Informática – Tese. 2. Programação genérica em GPU. 3. Visão Computacional. 4. Processamento de Imagens. 5. Otimização Discreta em Grafos. 6. Quadrees de Regiões. I. Gattass, Marcelo. II. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Informática. III. Título.

CDD: 004

## Agradecimentos

Aos meus orientadores, professores Marcelo Gattass e Paulo Cezar Carvalho por terem me acolhido com todo o apoio e incentivo para a realização deste trabalho.

À Asla Sá, merecidamente apelidada de “madrinha da tese”, pelo companheirismo, revisões e inúmeros debates calorosos sobre os rumos desta pesquisa.

Ao CNPq, à PUC-Rio e ao Tecgraf, pelos auxílios concedidos, sem os quais este trabalho não poderia ter sido realizado.

Aos meus pais, Geni e Anacleto, minhas irmãs, Ana Carolina e Bárbara e a toda a minha família. Meus primeiros e eternos companheiros nessa jornada, pacientes aos meus momentos de aflição e acima de tudo, incentivadores de minhas realizações.

Ao Fabio e aos amigos de longa data que souberam compreender minha ausência pelas longas horas de dedicação a esta pesquisa.

Aos amigos do Tecgraf e da PUC-Rio, do IMPA, da UFRJ e da Biológica Sistemas, na certeza que torcem constantemente por mim e compartilham o interesse pelas ciências e tecnologias.

Aos professores com quem tive a oportunidade de aprender, pelas marcas que deixaram em minha formação.

## Resumo

Vasconcelos, Cristina N.; Gattass, Marcelo. **Algoritmos para Processamento de Imagens e Visão Computacional para Arquiteturas Paralelas em Placas Gráficas**. Rio de Janeiro, 2009. 155p. Tese de Doutorado — Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

Diversas tarefas de Visão Computacional (VC) são formadas por operações aritméticas, replicadas sobre grandes volumes de dados. Esta caracterização descreve também as qualidades desejadas ao considerar uma aplicação qualquer como boa candidata a usufruir do crescente poder de processamento do hardware gráfico. Esta tese formula um conjunto de algoritmos de VC sobre representações do conteúdo visual em baixo nível, para serem processados em GPU. Dando suporte às propostas, são apresentadas uma visão geral das GPUs e de padrões de programação paralelos, os quais oferecem blocos construtores para tarefas de visão. Neste sentido, propomos uma definição formal para o padrão de Redução Múltipla e analisamos seu desempenho segundo diferentes fatores de redução e variações para seus arranjos. Apresentamos duas propostas para extração de informações no espaço da imagem em GPU: a MOCT descreve a localização de objetos identificáveis por suas cores em vídeos naturais, enquanto o operador de Redução Regional Múltipla Paralela (MRR) distribui a computação de operadores definidos sobre diferentes regiões de interesse. Descrevemos, como aplicação do MRR, a construção em GPU de um Diagrama Centroidal de Voronoi baseado no algoritmo de Lloyd. Tratamos do conteúdo visual em aglomerados de pixels, mais especificamente, em Quadrees de regiões. Introduzimos a QuadN4tree, um modelo para representação de quadrees que permite a navegação através do sistema de vizinhança das folhas e alcança custos ótimos no levantamento do conjunto de vizinhas de uma folha. Em seguida, propomos uma aceleração para aplicações baseadas em minimização de energia via corte de grafo, introduzindo uma etapa de pré-processamento que agrupa pixels similares em folhas de uma quadtree, com o objetivo de reduzir o tamanho do grafo sobre o qual o corte mínimo é encontrado. O método proposto é aplicado ao problema de segmentação de imagens com iluminação ativa. Algumas contribuições desta tese, descrevendo formulações paralelas a dados, foram publicadas nos artigos incluídos nos apêndices.

## Palavras-chave

Programação genérica em GPU. Visão Computacional. Processamento de Imagens. Otimização Discreta em Grafos. Quadrees de Regiões.

## Abstract

Vasconcelos, Cristina N.; Gattass, Marcelo. **Image Processing and Computer Vision Algorithms for Graphics Cards Parallel Architectures**. Rio de Janeiro, 2009. 155p. Thesis — Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro.

Arithmetically intensive operations, replicated over huge data sets (usually image pixels or scanned data), are an important part of many Computer Vision (CV) tasks, making them good candidates for taking advantage of the processing power of contemporary graphics processing units (GPUs). This thesis formulates a set of CV algorithms that use low level representations of visual content and are tailored for running on GPUs. A general view of GPUs and parallel programming patterns that offers interesting building blocks for CV tasks provides the necessary background for the algorithms. We also propose a formal definition for the Multiple Reduction pattern and evaluate its efficiency according to different reduction factors and layouts. We present two techniques for extracting data from the image space using the GPU: MOCT, a technique for tracking a set of objects identified by their colors from natural videos, and MRR, a technique for distributing the evaluation of a set of operators defined over different regions of interest within an image. As a MRR application we describe a Centroidal Voronoi Diagram construction based on Lloyd's algorithm but entirely computed using GPU resources. We also deal with visual content representations as pixel agglomerations, more specifically, as Regional Quadtrees. We introduce the QuadN4tree: a new model for representing quadtree leaves that allows navigation through their neighborhood systems and achieves an optimal cost for the retrieval of neighbor sets. We also propose modifying the setup for CV applications based on energy minimization via graph cuts, introducing a preprocessing step that groups similar pixels into regional quadtree leaves. This modification aims to reduce the size of the graph for which a minimum cut is to be found. We apply our proposed method to the problem of natural image segmentation by active illumination. Published papers detailing some contributions of this dissertation are included as appendixes. They present data-parallel formulations for the CV tasks we describe.

## Keywords

General-Purpose Computation on GPUs. Computer Vision. Image Processing. Discrete Graph-Based Optimization. Region Quadtrees.

# Sumário

1	Introdução	11
1.1	Visão Computacional	12
1.2	Contribuições e contexto da pesquisa	14
2	Programação Genérica em Placa Gráfica	16
2.1	Pipeline de Processamento Gráfico	17
2.2	Computação de propósito geral em GPU via Pipeline Gráfico	24
2.3	“GPU Computing”: Computação em GPU	30
2.4	Arquitetura	33
3	Processamento em Baixo Nível Utilizando Padrões Paralelos	41
3.1	Padrões de Programação Paralela	42
3.2	Manipulações em Pré-processamento	48
3.3	Localização de objetos	49
3.4	Localização de objetos com Região de Interesse	59
4	Detecção de Formas: Multirresolução e Agrupamento de Pixels	65
4.1	Multirresolução	66
4.2	Partição de imagens	68
4.3	Representação de imagens com Quadrees	69
4.4	QuadN4trees: sistema de referências entre folhas vizinhas	72
4.5	QuadN4trees em GPU	79
4.6	Agrupamento para detecção de objetos	85
4.7	Quadrees e Cortes em Grafos	93
5	Conclusão e Trabalhos Futuros	98
	Referências Bibliográficas	104
A	MOCT	112
B	CVD	123
C	QuadN4tree	136
D	QuadCut	145

## Lista de figuras

1.1	Paradigmas da Visão Computacional: do sinal aos símbolos	14
2.1	Comparação GFLOP/s CPU × GPU (CUDA08)	17
2.2	Abstração do Pipeline Gráfico	18
2.3	Entrada e saída de dados no processamento de vértices	23
2.4	Entrada e saída de dados no processamento de primitivas	24
2.5	Entrada e saída de dados no processamento de fragmentos	25
2.6	Etapas de uma renderização para computação genérica	26
2.7	Laço sequencial × computações em fluxos de entidades	27
2.8	(da esquerda para a direita) Mapeamento do Domínio da Aplicação ao seu reposicionamento pelo Vertex Shader e a distribuição das saídas do Fragment Shader	28
2.9	Hierarquia de Memória CUDA	33
2.10	Arquitetura Unificada	38
3.1	Operador de redução: pirâmide de texturas	45
3.2	Operador de redução múltipla	48
3.3	Localização de Objetos em Imagens Naturais	50
3.4	Localização de Objetos em Imagens Renderizadas	50
3.5	Arranjos do conjunto de objetos: horizontal, vertical e quadrangular	51
3.6	Criação da Base: subfluxos em arranjos verticais	53
3.7	Obtenção dos resultados globais	54
3.8	Comparação entre CPU, aplicação de (Bru07) repetidas vezes e o MOCT	55
3.9	Análise do fator de redução e arranjos	58
3.10	Hierarquia de fluxos de processamento das regiões de interesse	60
3.11	Base de avaliações Locais	62
3.12	Algoritmo de Lloyd	63
4.1	Pirâmides de Imagens	66
4.2	Quadtrees e suas representações	71
4.3	QuadN4: modelo de referências para folhas vizinhas	73
4.4	Construção da QuadN4tree para a Figura 4.2	73
4.5	Diferentes possibilidades para tomada de referências atendendo às propriedades do modelo QuadN4	74
4.6	Navegação pelas Bordas Norte, Leste, Oeste e Sul	74
4.7	Atualização de referências	78
4.8	Máscara de Folhas	81
4.9	Raio atravessando a pirâmide partindo da raiz da árvore	82
4.10	Consulta de Características das folhas	83
4.11	Armazenando a QuadN4tree na GPU	84
4.12	Grafos para o problema de classificação	88
4.13	Textura armazenando parcelas da função de energia.	92
4.14	Grafo sobre as folhas de uma quadtree	95

4.15 QuadCut: resultados mantém detalhes finos ao mesmo tempo que a otimização opera em conjunto reduzido de variáveis

97

## Lista de tabelas

4.1	Tempo de processamento (em milisegundos)	85
-----	--	----