

# Contents

<b>1</b>	<b>A GPU-Enabled Parallel Genetic Algorithm for Path Planning of Robotic Operators . . . . .</b>	<b>1</b>
	Panpan Cai, Yiyu Cai, Indhumathi Chandrasekaran, and Jianmin Zheng	
1.1	Introduction . . . . .	1
1.1.1	Motivation . . . . .	1
1.1.2	Objectives . . . . .	2
1.1.3	Organization of the Chapter . . . . .	2
1.2	Prior Arts . . . . .	2
1.2.1	GA Study . . . . .	2
1.2.2	Parallel GA (PGA) Study . . . . .	3
1.3	GPU-Enabled PGA . . . . .	4
1.3.1	GPU Fundamentals . . . . .	4
1.3.2	GPU PGA Design and Implementation . . . . .	5
1.4	GPU-Enabled PGA Application for Collision-Free Path Planning . . . . .	8
1.4.1	Collision Avoidance Path Planning . . . . .	8
1.4.2	Terminal Condition . . . . .	9
1.4.3	Global Optimization . . . . .	9
1.4.4	Performance . . . . .	11
	Conclusions . . . . .	12
	References . . . . .	12
<b>2</b>	<b>Real-Time Deformation of Constrained Meshes Using GPU . . . . .</b>	<b>15</b>
	Alexandre Kaspar and Bailin Deng	
2.1	Introduction . . . . .	15
2.1.1	Related Work . . . . .	16
2.1.2	Overview . . . . .	17
2.2	Overview of the Method . . . . .	18
2.2.1	Problem Formulation . . . . .	18
2.2.2	Numerical Solution . . . . .	20

2.3	General Implementation Strategies . . . . .	22
2.4	CUDA Implementation Details . . . . .	24
2.4.1	Kernels . . . . .	25
2.4.2	Sparse Linear Algebra . . . . .	28
2.5	Results . . . . .	28
2.6	Limitation and Future Work . . . . .	32
	Conclusion . . . . .	32
	References . . . . .	33
<b>3</b>	<b>GPU-Based Real-Time Volume Interaction for Scientific Visualization Education . . . . .</b>	<b>35</b>
	Yanlin Luo, Zhongke Wu, Zuying Luo, and Yanhong Luo	
3.1	Introduction . . . . .	35
3.2	Related Work . . . . .	36
3.2.1	GPU-Accelerated Volume Rendering . . . . .	36
3.2.2	Volume Interaction . . . . .	37
3.2.3	Volume Illustration . . . . .	38
3.3	The Proposed Method . . . . .	38
3.3.1	Transfer Function Design . . . . .	38
3.3.2	Plane Tools . . . . .	40
3.3.3	Superquadric Tools . . . . .	42
3.3.4	Virtual Lenses . . . . .	44
3.3.5	User Interaction and Implementation . . . . .	45
3.4	Results . . . . .	47
	Conclusions . . . . .	48
	References . . . . .	50
<b>4</b>	<b>Real-Time Separable Subsurface Scattering for Animated Virtual Characters . . . . .</b>	<b>53</b>
	P. Papanikolaou and G. Papagiannakis	
4.1	Introduction . . . . .	53
4.2	Previous Work . . . . .	54
4.3	Separable Subsurface Scattering for Dynamic Surfaces . . . . .	55
4.4	Implementation . . . . .	56
4.4.1	Light's Transmission Through Thin Skin . . . . .	59
4.4.2	Subsurface Scattering Implementation Novelties . . . . .	62
4.5	Comparison with Ground Truth . . . . .	63
	Conclusions . . . . .	65
	References . . . . .	66
<b>5</b>	<b>Adaptive NURBS Tessellation on GPU . . . . .</b>	<b>69</b>
	Yusha Li, Xingjiang Lu, Wenjing Zhang, and Guozhao Wang	
5.1	Introduction . . . . .	69
5.2	Estimating the Tessellation Intervals . . . . .	71
5.2.1	Tessellation Intervals for Rational Bézier Curves . . . . .	71
5.2.2	Tessellation Intervals for Rational Bézier Surfaces . . . . .	75

5.3	Creating Transition Regions . . . . .	76
5.3.1	Extracting Bézier Patches . . . . .	76
5.3.2	Filling the Gaps . . . . .	77
5.4	Implementations on GPU . . . . .	77
5.5	Experiment Results . . . . .	80
5.5.1	Comparisons to Zheng and Sederberg's Method . . . . .	80
5.5.2	Run Time on CPU and GPU . . . . .	81
	Conclusions . . . . .	83
	References . . . . .	84
<b>6</b>	<b>Graphics Native Approach to Identifying Surface Atoms of Macromolecules . . . . .</b>	<b>85</b>
	Huagen Wan, Yunqing Guan, and Yiyu Cai	
6.1	Introduction . . . . .	85
6.2	Prior Work . . . . .	87
6.3	Algorithm Overview and Implementation . . . . .	88
6.3.1	Algorithm Overview . . . . .	88
6.3.2	Implementation . . . . .	89
6.3.3	Improvements . . . . .	91
6.4	Experimental Results and Discussions . . . . .	92
	Conclusions . . . . .	93
	References . . . . .	95
<b>7</b>	<b>A Scalable Software Framework for Stateful Stream Data Processing on Multiple GPUs and Applications . . . . .</b>	<b>99</b>
	Farhoosh Alghabi, Ulrich Schipper, and Andreas Kolb	
7.1	Introduction . . . . .	100
7.2	Related Work . . . . .	102
7.3	The Framework . . . . .	103
7.3.1	Basic Concepts . . . . .	104
7.3.2	Distributed Graph . . . . .	105
7.3.3	Multiple Graph Instantiation . . . . .	106
7.4	Experimental Evaluation . . . . .	108
7.4.1	Comparison of Preliminary Implementations . . . . .	109
7.4.2	Scalability and Feedback . . . . .	112
7.5	Applications . . . . .	113
7.5.1	Information Security Using Crypto- and Steganography . . . . .	113
7.5.2	Crystallography Using a pnCCD Camera . . . . .	114
	Conclusion . . . . .	117
	References . . . . .	117

<b>8</b>	<b>The Design of SkyPACS: A High-Performance Mobile Medical Imaging Solution . . . . .</b>	<b>119</b>
	Tananan Pattanangkur, Sikana Tanupabrunson, Katchaguy Areekijseree, Sarunya Pumma, and Tiranee Achalakul	
8.1	Introduction . . . . .	120
8.2	Imagery Procedure . . . . .	121
8.3	Features of SkyPACS . . . . .	122
8.4	Software Design . . . . .	123
8.5	Implementation and Deployment . . . . .	127
8.6	Product Comparisons . . . . .	128
	Conclusion . . . . .	130
	Appendix A: Screenshots . . . . .	131
	References . . . . .	132
<b>9</b>	<b>Collision Detection Based on Fuzzy Scene Subdivision . . . . .</b>	<b>135</b>
	David Mainzer and Gabriel Zachmann	
9.1	Introduction . . . . .	135
9.1.1	Our Contributions . . . . .	136
9.2	Previous Work . . . . .	137
9.2.1	Approaches Using Bounding Volume Hierarchies . . . . .	137
9.2.2	GPU-Based Collision Detection . . . . .	138
9.3	Sweep-Plane Technique Using PCA for Collision Detection . . . . .	138
9.3.1	Thread Management . . . . .	140
9.4	Object Subdivision Using Fuzzy C-Means . . . . .	141
9.5	GPU-Based Collision Detection . . . . .	143
9.5.1	Accuracy and Limitations . . . . .	145
9.6	Results . . . . .	146
9.6.1	Benchmarking . . . . .	146
	Conclusions and Future Work . . . . .	148
	References . . . . .	149
<b>10</b>	<b>Smoothed Particle Hydrodynamics Applied to Cartilage Deformation . . . . .</b>	<b>151</b>
	Philip Boyer, Sean LeBlanc, and Chris Joslin	
10.1	Introduction and Background . . . . .	151
10.2	Materials and Methods . . . . .	154
10.2.1	Elastic Solid Forces . . . . .	154
10.2.2	Rigid Boundary Collision Handling . . . . .	157
10.2.3	Implementation . . . . .	158
10.3	Results . . . . .	159
10.3.1	Solid Rod and Falling Wedge Tests . . . . .	159
10.3.2	Cartilage Simulation . . . . .	162
	Conclusions and Future Work . . . . .	163
	References . . . . .	163

<b>11 A GPU-Based Real-Time Algorithm for Virtual Viewpoint Rendering from Multi-video</b>	167
Kyrylo Shegeda and Pierre Boulanger	
11.1 Introduction	167
11.2 Common Plane Sweeping Algorithm	169
11.2.1 Depth-Map Estimation Algorithm	169
11.2.2 Pixel Similarity Function	171
11.2.3 Projective Block Matching	171
11.2.4 Virtual Viewpoint Rendering	172
11.2.5 GPU-Accelerated Algorithm and Its Implementation	174
11.2.6 Constructing OpenGL Model-View and Projection Matrices	179
11.3 Experimental Results	180
Conclusion	183
References	184
<b>12 A Middleware Framework for Programmable Multi-GPU-Based Big Data Applications</b>	187
Ettikan K. Karuppiiah, Yong Keh Kok, and Keeratpal Singh	
12.1 Introduction	187
12.2 Related Work	190
12.3 Middleware Framework Design	192
12.3.1 Big Data Needs	192
12.3.2 Presentation Layer	194
12.3.3 Interface Layer	194
12.3.4 Middleware Layer	194
12.3.5 Orchestration Engine (with Example of Use Case)	195
12.3.6 Storage	196
12.3.7 Mi-AccLib and Analytics Component	197
12.4 Implementation	199
12.5 Results	201
Conclusions	204
References	205
<b>13 On the Efficient Implementation of a Real-Time Kd-Tree Construction Algorithm</b>	207
Byungjoon Chang, Woong Seo, and Insung Ihm	
13.1 Background and Our Contribution	207
13.2 Optimizations for the Large-Node Stage	208
13.2.1 Triangle Sorting with Respect to Splitting Planes	209
13.2.2 AABB Computations for Active Large Nodes	212
13.3 Optimizations for the Small-Node Stage	213
13.4 Experimental Results	213
Concluding Remarks	216

Appendix: A Single-Kernel Implementation for the Triangle-Sorting Process (Sect. 13.2.1.2) . . . . .	217
References . . . . .	219
<b>14 Fast Approximate k-Nearest Neighbours Search Using GPGPU . . .</b>	<b>221</b>
Niko Lukač and Borut Žalik	
14.1 Introduction . . . . .	221
14.2 Related Work . . . . .	223
14.3 Parallel Multi-probe LSH . . . . .	224
14.3.1 Locality-Sensitive Hashing . . . . .	224
14.3.2 Parallel MLSH Using CUDA . . . . .	227
14.4 Results . . . . .	230
Conclusion . . . . .	232
References . . . . .	233
<b>15 Soft Computing Methods for Big Data Problems . . . . .</b>	<b>235</b>
Shafaatunnur Hasan, Siti Mariyam Shamsuddin, and Noel Lopes	
15.1 Introduction . . . . .	236
15.2 Related Work . . . . .	237
15.3 GPU Machine Learning Library Implementation . . . . .	238
15.3.1 Parallel Multiple Back-Propagation . . . . .	238
15.3.2 Parallel Self-Organizing Map . . . . .	239
15.4 Experimental Setup . . . . .	239
15.4.1 Dataset Preparation . . . . .	239
15.4.2 Performance Measurement . . . . .	240
15.5 Experimental Result and Analysis . . . . .	242
15.5.1 Speed Analysis . . . . .	243
15.5.2 Classification Analysis . . . . .	245
Conclusion . . . . .	246
References . . . . .	246
<b>16 Numerical Solution of BVP on GPU with Application to Path Planning . . . . .</b>	<b>249</b>
Lumír Janošek, Martin Němec, and Radoslav Fasuga	
16.1 Introduction . . . . .	249
16.2 Harmonic Potential Field . . . . .	250
16.3 Iterative Methods . . . . .	251
16.4 Implementation . . . . .	252
16.5 Results . . . . .	254
Conclusion . . . . .	256
References . . . . .	256
<b>17 Fast Multi-Keyword Range Search Using GPGPU . . . . .</b>	<b>259</b>
Amirul Abdullah, Amril Nazir, Mohanavelu Senapan Soo Saw Meng, and Ettikan Karuppiah	
17.1 Introduction . . . . .	259
17.2 Background . . . . .	261

17.2.1	Keyword Search . . . . .	261
17.2.2	Binary Search . . . . .	262
17.2.3	Multi-keyword Search (P-ary Search) . . . . .	262
17.3	Implementation . . . . .	264
17.3.1	Data Packing . . . . .	265
17.3.2	Memory Coalescing . . . . .	266
17.3.3	Shared Memory . . . . .	267
17.4	Experimental Evaluation . . . . .	268
17.4.1	Response Time . . . . .	268
17.4.2	Speed-Ups . . . . .	271
17.4.3	Throughput . . . . .	272
	Conclusions and Future Work . . . . .	272
	References . . . . .	273
	<b>Index . . . . .</b>	<b>275</b>

<http://www.springer.com/978-981-287-133-6>

GPU Computing and Applications

Cai, Y.; See, S. (Eds.)

2015, XVIII, 280 p. 127 illus., 85 illus. in color.,

Hardcover

ISBN: 978-981-287-133-6