Table of Contents

List of Tables **xiii**

List of Figures **xv**

List of Algorithms **xxiii**

1 Introduction **1**
1.1 Motivation . . . 1
1.2 Objectives . . . 3
1.2.1 General Objective . . . 3
1.2.2 Specific Objectives . . . 3
1.3 Methodology . . . 3
1.4 Structure of this Work . . . 4

2 Theoretical Framework **5**
2.1 Polygon Meshes . . . 5
2.1.1 Smoothing . . . 7
2.1.2 Delaunay Triangulation . . . 7
2.1.3 Simplification . . . 9
2.2 GPU Architecture . . . 9
2.2.1 Programming model . . . 11
2.2.2 Heterogeneous Computing . . . 13
2.3 Cleap Library . . . 16
2.4 Mesh File Format . . . 18
2.4.1 .OFF Format . . . 18
2.4.2 .PLY Format . . . 19

3 Literature Review **21**
3.1 Simplification Techniques . . . 21
3.1.1 Vertex-Clustering . . . 21
3.1.2 Vertex-Decimation . . . 22
3.1.3 Vertex-Contraction . . . 23
3.1.4 Edge-Collapse . . . 24
3.2 Simplification Metrics . . . 25
3.2.1 Edge length/Angle . . . 25
3.2.2 *Quadric* error metric . . . 26
3.2.3 Volume Error . . . 28

3.2.4 Energy Function . . . 28
3.3 Algorithms Review . . . 29
3.3.1 Shontz-Nistor . . . 29
3.3.2 Garland-Heckbert . . . 33
3.3.3 Papageorgiou-Platis . . . 34
3.3.4 Melax . . . 37
3.3.5 Franc-Skala . . . 38
3.3.6 Lee-Kyung . . . 39
3.3.7 Hoppe et. al. . . . 41
3.3.8 Hjelmervik-Leon . . . 42
3.3.9 Grund et al . . . 45
3.3.10 Cellier et al . . . 46
3.3.11 Vad'ura . . . 47

4 Algorithm Proposal **49**
4.1 Simplification Scheme . . . 49
4.2 Data structures . . . 50
4.3 Candidate Selection . . . 52
4.4 Mutual Exclusion Mechanism . . . 56
4.5 Integrity Test . . . 58
4.5.1 Inversion Test . . . 60
4.5.2 Border-Edge Test . . . 62
4.5.3 Neighbors Test . . . 63
4.6 Simplification Algorithm . . . 66
4.6.1 Elimination Phase . . . 67
4.6.2 Triangle Repair Phase . . . 68
4.6.3 Edge Repair Phase . . . 70
4.6.4 Connectivity Repair Phase . . . 72
4.7 Algorithm Output . . . 75
4.7.1 Vertices Rearrange . . . 75
4.7.2 Save into a File . . . 76

5 Results **79**
5.1 Quantitative Results . . . 80
5.2 Qualitative Results . . . 96
5.3 Analysis . . . 113

6 Conclusions **115**
6.1 Conclusions . . . 115
6.2 Future Work . . . 117

Bibliography **119**

Appendices **123**

A Parallel Lock-down Mechanism Example **125**
A.1 Example 1: Two threads compete for a shared area . . . 125
A.2 Example 2: Two threads that do not compete for a shared area . . . 133

B Adjacency Table Example **137**

C Edge-Collapse Algorithm Example **141**

D Vertices Array Rearrange Example **149**