HIGH-PERFORMANCE 3D MAPPING OF UNKNOWN ENVIRONMENTS USING PARALLEL COMPUTING FOR MOBILE ROBOTS

K. T. D. S. De Silva

198011G

Degree of Master of Science/ Master of Engineering

Department of Computer Science and Engineering

University of Moratuwa
Sri Lanka

August 2021

HIGH-PERFORMANCE 3D MAPPING OF UNKNOWN ENVIRONMENTS USING PARALLEL COMPUTING FOR MOBILE ROBOTS

K. T. D. S. De Silva

198011G

Thesis submitted in partial fulfillment of the requirements for the Degree of Master of Science (Research) in Computer Science and Engineering

Department of Computer Science and Engineering

University of Moratuwa
Sri Lanka

August 2021

DECLARATION

I declare that this is my own work, and this dissertation does not incorporate without acknowledgement any material previously submitted for a Degree or Diploma in any other University or institute of higher learning and to the best of my knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text.

Also, I hereby grant to University of Moratuwa the non-exclusive right to reproduce and distribute my dissertation, in whole or in part in print, electronic or other medium. I retain the right to use this content in whole or part in future works (such as articles or books).

Name of the candidate: K. T. D. S. De Silva

Signature:

Date: 13/12/2021

The above candidate has carried out research for the Master's thesis/dissertation under my supervision.

Name of the supervisor: Dr. C. D. Gamage
Signature of the supervisor:

Date: 13.12.2021

Name of the supervisor: Dr. S. J. Sooriyaarachchi
Signature of the supervisor:

Date: 14/12/2021

ABSTRACT

Autonomous multi-robot systems are a popular research field in the 3D mapping of unknown environments. High fault tolerance, increased accuracy, and low latency in coverage are the main reasons why a multi-robot system is preferred over a single robot in an unpredictable field. Compared with 3D scene reconstruction which is a conceptually similar but resource-wise different technique, autonomous mobile robot 3D mapping techniques are missing a crucial element. Since most mobile robots run on low computationally powered processing units, the real-time registration of point clouds into high-resolution 3D occupancy grid maps is a challenge.

Until recently, it was nearly impossible to perform parallel point cloud registration in mobile platforms. Serial processing of a large amount of high-frequency input data leads to buffer overflows and failure to include all information into the 3D map. With the introduction of Graphical Processing Units (GPUs) into commodity hardware, mobile robot 3D mapping now can achieve faster time performance, using the same algorithmic techniques as 3D scene reconstruction. However, parallelization of mobile robot 3D occupancy grid mapping process is a less frequently discussed topic.

As a Central Processing Unit (CPU) is necessary to run conventional middleware, operating system, and hardware drivers, the system is developed as a CPU-GPU mixed pipeline. The precomputed *free scan mask* is used to accelerate the process of identifying free voxels in space. Point positional information is transformed into unsinged integer coordinates to cope with Morton codes, which is a linear representation of octree nodes instead of traditional spatial octrees. 64-bit M-codes and 32-bit RGBO-codes are stored in a hash table to reduce access time compared to a hierarchical octree.

Point cloud transformation, ray tracing, mapping point coordinated into integer scale, Morton-coded voxel generation, RGBO-code generation are the processes that are performed inside the GPU. Retrieving point cloud information, map update using bitwise operations and map publish are executed within the CPU.

Additionally, a multi-robot system is prototyped as a team of wheeled robots autonomously exploring an unknown, even-surfaced environment, while building and merging fast 3D occupancy grid maps and communicating using a multi-master communication protocol.

Keywords – Multi-robot system, GPU acceleration, 3D mapping, occupancy grids, point cloud registration, ray tracing, free scan mask, Morton order, linear octree

ACKNOWLEDGEMENT

This thesis is a result of the guidance and encouragement of various personnel. Without them, I wouldn’t be able to make it a success.

First and foremost, I would like to thank my research supervisors, Dr. Chandana Gamage and Dr. Sulochana Sooriyaarachchi. They constantly supported me with technical guidance throughout the research and steered my research in the right direction. They also provided me necessary motivation during the Covid-19 lockdown period.

Next, I thank my progress review panel, Dr. Chathura De Silva and Dr. Thilina Lalitharatne, for dedicating their time and providing me valuable feedback for improvement.

This research was funded by Xavier AI Pvt. Ltd., Previously known as Rebirth Technologies Pvt. Ltd., Sri Lanka. A special thanks go to them for their interest in financial investment, and feedback on taking this project into a commercial system.

I would like to thank my friends at the final year project group for laying the groundwork for this research and the initial prototype development. Also, the support and company of the staff of IntelliSense research lab, University of Moratuwa must be appreciated.

Finally, I would like to thank my husband and my family for the encouragement and the immense support they gave me in their best.

TABLE OF CONTENTS

DECLARATION i

ABSTRACT ii

ACKNOWLEDGEMENT iii

TABLE OF CONTENTS iv

LIST OF FIGURES ix

LIST OF TABLES xi

LIST OF ABBREVIATIONS xii

1 INTRODUCTION 1

1.1 Background 1

1.2 Multi-robot systems 2

1.3 Performance measures of 3D mapping 2

1.4 High-performance computing for 3D map generation 3

1.5 The base prototype system 4

1.6 Problem statement 5

1.7 Objectives 5

1.8 Research contribution 6

1.9 Structure of thesis 6

2 LITERATURE REVIEW 8

2.1 Introduction 8

2.2 3D environmental mapping approaches for mobile robots 8

2.2.1 Grid based mapping methods 8

2.2.2 Point cloud map methods 12

2.2.3 Polygonal map methods 14

2.2.4 Summary 15

2.3 3D scene reconstruction approaches using high-performance computing techniques ... 17

2.3.1 Prominent work ... 17

2.3.2 Summary ... 22

2.4 Inter-robot system coordination and communication ... 23

2.5 Autonomous exploration ... 25

2.6 Literature evaluation of the base prototype ... 26

2.6.1 3D map building system ... 27

2.6.2 Inter-robot communication network ... 27

2.6.3 Autonomous navigation system ... 28

2.6.4 Evaluation results ... 28

2.7 Summary ... 28

3 METHODOLOGY ... 30

3.1 Introduction ... 30

3.2 3D mapping ... 30

3.2.1 Selecting a 3D map type ... 31

3.2.2 OctoMap 3D map building system ... 33

3.2.3 Selected methodology ... 38

3.3 Inter-robot communication and multi-robot coordination ... 39

3.3.1 ROS compatible communication methods ... 39

3.3.2 Multi-robot coordination ... 40

3.3.3 Selected methodology ... 42

3.4 Autonomous Exploration ... 44

3.5 Summary ... 45

4 SYSTEM DESIGN ... 47

4.1 Introduction ... 47

4.2 System architecture 47

4.3 System data flow 50

4.4 Implementation tools and technologies 51

4.5 Evaluating CUDA performance compared to single threaded processing.. 51

4.6 Summary 52

5 GPU ACCELERATED 3D OCCUPANCY GRID MAP BUILDING 54

5.1 Introduction 54

5.2 Parallel computing for 3D Occupancy grid map building 54

5.3 System Design of G3DOMap 55

5.3.1 Introduction 55

5.3.2 Architecture 55

5.3.3 Data flow 56

5.4 Preparing data arrays 57

5.5 Ray tracing in free space to mark free occupancy voxels 58

5.5.1 Free scan mask 60

5.5.2 Free space voxel identification 61

5.6 Transformation and mapping into integer scale 63

5.7 Generation of Morton codes and octree node values 64

5.8 Creating octree node values 65

5.9 Updating the map 65

5.9.1 Traditional octree 66

5.9.2 Morton coded octree 67

5.9.3 Comparison with traditional octree 69

5.10 Decoding and publishing the map 70

5.11 Resource consumption and results 71

5.11.1 System parameters 71

5.11.2 Static map building 71

5.11.3 Dynamic environment map building 72

5.12 Comparison between OctoMap and G3DOMap 74

5.12.1 System Parameters 74

5.12.2 Visual comparison 74

5.12.3 Dynamic environment mapping comparison 78

5.12.4 Time comparison 79

5.13 Summary 79

6 3D MAP MERGING BY DIRECT METHOD 82

6.1 Introduction 82

6.2 System design 82

6.2.1 Results 84

6.2.2 System errors 85

6.3 3D map merging using erroneous odometry sensor data 86

6.3.1 Object shifts 87

6.3.2 Different object perspectives 87

6.3.3 Overcoming mapping errors 88

6.4 Summary 89

7 INTER ROBOT COMMUNICATION 90

7.1 Introduction 90

7.2 Multimaster-fkie package 90

7.2.1 Introduction 90

7.2.2 Integration to the system 91

7.3 Summary 92

8 AUTONOMOUS EXPLORATION IN UNKNOWN EVEN-SURFACED ENVIRONMENTS 93

8.1 Introduction 93

8.2 Explore-lite as the navigation planner 93

8.3 Integration to the system and results 94

8.4 Summary 95

9 CONCLUSION 97

9.1 Introduction 97

9.2 Summary of results 97

9.2.1 G3DOMap 97

9.2.2 3D map merging 99

9.2.3 Inter-robot communication 99

9.2.4 Autonomous exploration in unknown even-surfaced environment ... 100

9.3 Future work 100

REFERENCES 101

LIST OF FIGURES

Figure 1.1: Two example applications of autonomous mobile robot mapping. 1
Figure 1.2: Example applications requiring high accuracy in mapping. 3
Figure 1.3: Environment uncertainty of an autonomous vehicle................................. 4
Figure 2.1: Three Stages of 3D MVOG map generation. 10
Figure 2.2: Octree memory structure 11
Figure 2.3: Point cloud to planer representation 15
Figure 2.4: Cube level data structure. 21
Figure 3.1: Abstract process flow of OctoMap Server 34
Figure 3.2: Missing scans in an OctoMap generated by a rotating robot for a simulated environment. 36
Figure 3.3: An incomplete color registration in a colored OctoMap 36
Figure 3.4: ROS compatible communication system architectures. 41
Figure 3.5: Extended clustered architecture of a multi-master system 44
Figure 3.6: Overall system Methodology 46
Figure 4.1: Overall system architecture 49
Figure 5.1: Data Flow Diagram of G3DOMap 56
Figure 5.2: Bird eye view of a point cloud obtained from a simulated scene............ 59
Figure 5.3: Effect of the distance to the point cloud density 60
Figure 5.4: The free scan mask for Microsoft Kinect 360 60
Figure 5.5: Ray tracing in free space to identify free voxels 61
Figure 5.6: Morton code for an octree structure. 66
Figure 5.7: Traditional octree hierarchy with children node identifiers 67
Figure 5.8: Tree update time with the number of nodes 70
Figure 5.9: The Gazebo simulated world... 72
Figure 5.10: Number of tree nodes after registering each of first 240 scans 73
Figure 5.11: Full map comparison of G3DOMap and OctoMap 75
Figure 5.12 A snapshot series of the two mapping approaches 76
Figure 5.13: 3D occupancy grids of complex objects at different resolutions. 77
Figure 5.14: Object shift (a) observed in G3DOMap. 78
Figure 5.15: Invalid boundaries 78

Figure 5.16: Dynamic environment responses of the two mapping techniques 79
Figure 5.17: Point cloud registration times at different resolutions........................... 81
Figure 6.1: System architecture of 3D map merging 84
Figure 6.2: Maps merged with known initial poses. 85
Figure 6.3: Identified errors in direct map merging.. 86
Figure 6.4: Errors observed in 3D maps: 87
Figure 6.5: Different object perspectives in occupancy grid maps 88
Figure 8.1: ROS message passing in explore-lite. 93
Figure 8.2: A snapshot from explore-lite navigation: 95
Figure 8.3: CostMaps published for navigation... 95
Figure 8.4: A series of snapshots of a single robot exploring an office room using explore-lite 96

LIST OF TABLES

Table 2.1: Mobile robot 3D mapping techniques related work summary 18

Table 2.2: GPU-accelerated scene reconstruction related work summary 22

Table 2.3: Inter-robot system coordination and communication related work summary 23

Table 2.4: Autonomous robot exploration related work summary 25

Table 3.1: A summary of techniques used in autonomous robot 3D mapping and scene reconstruction 31

Table 3.2: Comparison of 3D map types 32

Table 3.3: Average time consumption by the processes in OctoMap........................ 38

Table 3.4: Selected Methodology for 3D map generation 38

Table 3.5: A comparison of available communication methods in ROS................... 40

Table 3.6: Comparison of Multi-robot coordination schemes 41

Table 3.7: Selected methodology for multi-robot coordination and inter-robot communication 44

Table 3.8: Selected methodology for the autonomous exploration 45

Table 4.1: CUDA parameters of NVIDIA Jetson TX2... 52

Table 4.2: Simple vector equation time comparison between CPU and GPU........... 52

Table 5.1: Octree hierarchy derived from Morton code 69

Table 5.2: Process breakdown and time comparison in static map model 71

Table 5.3: Process breakdown and time comparison in dynamic map model........... 73

Table 7.1: Steps to configure Multimaster-fkie into the system................................ 92

LIST OF ABBREVIATIONS

AP	Access Point
API	Application Programming Interface
CPU	Central Processing Unit
CUDA	Computed Unified Device Architecture
DSM	Digital Surface Model
Gb	Giga bytes
GPS	Global positioning system
GPU	Graphics Processing Unit
ICP	Iterative closest Point
IMU	Inertial Measurement Unit
IP	Internet Protocol
LUT	Look Up Table
MANET	Mobile Adhoc Network
MAV	Micro Aerial Vehicle
M-code	Morton Code
MCL	Monte Carlo Localization
MVOG	Multi Volume Occupancy Grid
NBV	Next Best View
NDT	Normal Distribution Transform
OS	Operating System
ROS	Robot Operating System
RGBO-code	Red, Green, Blue, Occupancy Code
RRT	Rapidly Exploring Random Trees
SAR	Search And Rescue
SDF	Signed Distance Function
TSDF	Truncated Signed Distance Function
SIMD	Single Instruction Multiple Data
URI	Uniform Resource Identifier