

Instituto Superior de Engenharia do Porto



Sistema de Modelização Tridimensional Para Ambientes Não Estruturados

João Ricardo Pereira de Sousa

Tese submetida no âmbito do
Mestrado em Engenharia Electrotécnica e de Computadores

Orientador: Eduardo Alexandre Pereira da Silva (Doutor)

Co-orientador: José Miguel Soares de Almeida (Mestre)

Dezembro de 2009

Resumo

A navegação de veículos autónomos em ambientes não estruturados continua a ser um problema em aberto. A complexidade do mundo real ainda é um desafio. A difícil caracterização do relevo irregular, dos objectos dinâmicos e pouco distintos (e a inexistência de referências de localização) tem sido alvo de estudo e do desenvolvimento de vários métodos que permitam de uma forma eficiente, e em tempo real, modelizar o espaço tridimensional.

O trabalho realizado ao longo desta dissertação insere-se na estratégia do Laboratório de Sistemas Autónomos (LSA) na pesquisa e desenvolvimento de sistemas sensoriais que possibilitem o aumento da capacidade de percepção das plataformas robóticas. O desenvolvimento de um sistema de modelização tridimensional visa acrescentar aos projectos LINCE (*Land INtelligent Cooperative Explorer*) e TIGRE (*Terrestrial Intelligent General proposed Robot Explorer*) maior autonomia e capacidade de exploração e mapeamento.

Apresentamos alguns sensores utilizados para a aquisição de modelos tridimensionais, bem como alguns dos métodos mais utilizados para o processo de mapeamento, e a sua aplicação em plataformas robóticas.

Ao longo desta dissertação são apresentadas e validadas técnicas que permitem a obtenção de modelos tridimensionais. É abordado o problema de analisar a cor e geometria dos objectos, e da criação de modelos realistas que os representam.

Desenvolvemos um sistema que nos permite a obtenção de dados volumétricos tridimensionais, a partir de múltiplas leituras de um Laser Range Finder bidimensional de médio alcance. Aos conjuntos de dados resultantes associamos numa nuvem de pontos coerente e referenciada.

Foram desenvolvidas e implementadas técnicas de segmentação que permitem inspecionar uma nuvem de pontos e classifica-la quanto às suas características geométricas, bem como ao tipo de estruturas que representem.

São apresentadas algumas técnicas para a criação de Mapas de Elevação Digital, tendo sido desenvolvida um novo método que tira partido da segmentação efectuada.

Palavras Chave: Detecção de Obstáculos, Percepção do Terreno, Varrimentos Laser 3D, Mapeamento de Texturas, Navegação Autónoma. LIDAR, LADAR, LRF.

Abstract

The navigation of autonomous vehicles in unstructured environments is still an open problem. The complexity of the real world is still a challenge. The harsh characterization of irregular terrain, of dynamic objects poorly defined (and the lack of references location) has been a subject of study and development of several methods for efficiently, and in real time, modeling the three-dimensional space.

The work along this dissertation is part of the strategy of the Laboratório de Sistemas Autônomos (LSA) in the research and development of sensory systems that allow the increase of the perceptual capacity of robotic platforms. The development of a three-dimensional modeling system aims to add to the projects LINCE (Land INtelligent Co-operative Explorer) e TIGRE (Terrestrial Intelligent General proposed Robot Explorer) greater autonomy and capacity for exploration and mapping.

We present some sensors used for the acquisition of three-dimensional models, as well as some of the most widely used methods for the mapping process, and its applications in robotic platforms.

Throughout this dissertation are presented and validated techniques to obtain three-dimensional models. It is addressed the problem of analyzing the color and geometry of objects, and the creation of realistic models that represent them.

It as been developed a system that allows us to obtain three-dimensional volumetric data, from multiple readings of a bi-dimensional, medium range laser range finder. To the resulting data sets we have associated a consistent and referenced point cloud.

we developed and implemented segmentation techniques that allow inspecting a cloud of points and classifies them as to their geometric characteristics and the type of structures they represent.

Some techniques for the creation of digital elevation maps are presented, having been developed a new method that takes advantage of the segmentation carried out.

Keywords: Obstacle Detection, Terrain Perception, 3D Laser Scanning, Texture Mapping, Autonomous Navigation, LIDAR, LADAR, LRF.

Agradecimentos

No culminar de uma etapa fundamental da minha vida académica, e olhando em retrospectiva, acho de elementar justiça agradecer aos meus pais, à minha irmã e cunhado, queria reconhecer todo o auxílio e incentivo ao longo destes anos. Sem vocês não teria sido possível chegar ao fim deste percurso.

A todos professores que me acompanharam, o meu muito obrigado, pela ajuda e constante motivação que me permitiram nos momentos de maior dificuldade ultrapassar os diferentes obstáculos.

Aos meus colegas de curso agradeço a ajuda, a paciência, a disponibilidade e o espírito de equipa que demonstraram, ajudando a transformar jornadas de trabalho em tarefas divertidas.

João Ricardo Pereira de Sousa

Esta página foi intencionalmente deixada em branco.

”Knowledge is like a candle. When you light your candle from mine, my light is not diminished. It is enhanced and a larger room is enlightened as a consequence.”

Thomas Jefferson

Esta página foi intencionalmente deixada em branco.

Conteúdo

1	Introdução	1
1.1	Âmbito da Dissertação	1
1.2	Motivação e Enquadramento	3
1.3	Objectivos	3
1.4	Estrutura da Dissertação	4
2	Estado da Arte	7
2.1	Tecnologias de Mapeamento	7
2.1.1	Riegl	8
2.1.2	Velodyne	8
2.2	Algoritmos e Processos de Mapeamento	10
2.3	Aplicações em Plataformas Robóticas	10
2.3.1	Kurt3D	11
2.3.2	RTS-HANNA	12
2.3.3	<i>Groundhog</i>	13
2.3.4	Stanley	14
2.3.5	Argo	15
3	Aquisição e Registo de Dados	17
3.1	Sistema de Aquisição de Informação	17
3.1.1	Sensor <i>Laser</i>	17
3.1.2	Sistema de Rotação	20
3.1.3	Erro do Sistema de Rotação	20
3.2	Registo	21
3.2.1	Rotações e Translações	22
3.2.1.1	Transformada <i>Laser</i> para Centro de Rotação	23
3.2.1.2	Transformada Centro de Rotação para Centro Plataforma Robótica	23
3.2.1.3	Transformada Centro da Plataforma Robótica para o Mundo	24
3.2.2	Sistema de Posicionamento Global	25
3.2.3	Sistema de Navegação Inercial	25
3.3	Fusão de Dados	25
3.3.1	Sincronização	26
3.3.2	<i>Laser</i> e Visão	27
4	Segmentação	31
4.1	Análise quantitativa	32
4.1.1	Determinação de <i>Breakpoints</i>	32

4.1.2	Determinação de Pontos de Inflexão	34
4.1.3	Detecção de Círculos	36
4.1.4	A Terceira Dimensão	36
4.2	Análise Qualitativa	38
4.2.1	Técnica de Segmentação <i>Brenneke</i>	39
4.2.2	Técnica de Segmentação <i>Brenneke</i> Extendida	40
5	Técnicas de Mapeamento	45
5.1	Mapas de Elevação Digital	46
5.1.1	<i>Digital Elevation Map</i>	46
5.1.1.1	Altura Máxima	47
5.1.1.2	Altura Média	47
5.1.1.3	Maior Número de Pontos	48
5.1.1.4	Mapa de Elevação Robusto	48
5.1.2	<i>Extended Digital Elevation Map</i>	49
5.1.3	<i>Multi Level Surface Map</i>	50
5.2	Árvores Octais	52
5.3	Mapa de Superfícies e <i>Landmarks</i>	53
5.3.1	Nível Chão	54
5.3.2	Nível <i>Landmark</i>	54
5.3.3	Nível <i>Landmark</i> Suspensa	55
5.3.4	Nível Suspenso	56
6	Resultados	59
6.1	Nuvens de Pontos	59
6.1.1	Nuvem de Pontos Segmentada	61
6.1.2	Nuvem de Pontos com Informação de Cor	63
6.2	Técnicas de Mapeamento	64
6.2.1	Segmentos de Recta	64
6.2.2	Mapa de Superfícies e de <i>Landmarks</i>	65
7	Conclusões e Trabalho Futuro	67
7.1	Conclusões	67
7.2	Trabalho Futuro	68
	Referências	74

Lista de Figuras

1.1	Processo de mapeamento.	2
2.1	Dispositivo <i>Riegl</i> e mapa gerado.	8
2.2	Dispositivo Velodyne.	9
2.3	Nuvem de pontos recolhida numa área urbana.	9
2.4	Kurt3D.	11
2.5	Exemplo de um cenário urbano “ <i>visto</i> ” pelo Kurt3D.	12
2.6	RTS-HANNA e um mapa gerado.	13
2.7	Groundhog e representação volumétrica.	14
2.8	<i>Robot Stanley</i> e modelo volumétrico.	14
2.9	Plataforma ARGO do laboratório CAS e nuvem de pontos.	15
3.1	<i>Laser range finder</i> Sick LMS-291	18
3.2	Sick LMS-291, princípio de operação e varrimento	19
3.3	Esfera de varrimento do LRF tridimensional	20
3.4	Rotações em torno dos três eixos cartesianos.	22
3.5	Transformadas entre referenciais.	23
3.6	Disposição física do sensor em relação ao suporte.	24
3.7	Comunicação do Sick LMS em modo de <i>raw data</i>	27
3.8	Padrão de xadrez observado simultaneamente pelo <i>laser</i> e câmara.	28
3.9	Resultado da calibração extrínseca do <i>laser</i> e câmara.	29
3.10	Nuvem de pontos de um conjunto de objectos e fotografia dos mesmos.	30
4.1	Círculo máximo de aceitação.	32
4.2	Exemplo de um Varrimento Bidimensional.	34
4.3	Determinação de Pontos de Inflexão.	35
4.4	Ângulos inscritos num arco de circunferência.	36
4.5	Expansão de uma superfície com uma nova linha.	37
4.6	Exemplos de cenários <i>outdoor</i>	38
4.7	Definição de pontos de obstáculo.	40
4.8	Exemplo de um varrimento vertical.	41
4.9	Áreas de Aceitação das diferentes classes de pontos.	42
4.10	Exemplo de um varrimento vertical segmentado.	43
5.1	Mapas de superfícies gerados com base numa nuvem de pontos.	46
5.2	Criação de um mapa de elevação.	48
5.3	Criação de um mapa de elevação extendido.	49
5.4	Exemplo de diferentes células num mapa MLSM.	51
5.5	Árvore octal.	52

5.6	Criação de uma grelha de superfícies.	53
5.7	Criação do nível do tipo Chão de uma célula.	55
5.8	Criação do nível do tipo <i>Landmark</i> de uma célula.	55
5.9	Criação do nível do tipo <i>Landmark</i> Suspensa de uma célula.	56
5.10	Criação do nível de tipo Suspenso de uma célula.	57
5.11	Mapa de Superfícies e <i>Landmarks</i> de um cenário <i>indoor</i>	57
6.1	Cenário <i>indoor</i>	60
6.2	Nuvem de pontos referente ao Cenário 1.	60
6.3	Nuvem de pontos segmentada referente ao Cenário 1.	61
6.4	Vista parcial (lado esquerdo) do Cenário 1 e nuvem de pontos.	62
6.5	Vista parcial (zona central) do Cenário 1 e nuvem de pontos.	62
6.6	Vista parcial (lado direito) do Cenário 1 e nuvem de pontos.	63
6.7	Fotografia do conjunto de obstáculos (Cenário 2).	63
6.8	Nuvem de pontos com informação de cor.	64
6.9	Imagem do modelo aproximado por aproximação a segmentos de rectas.	65
6.10	Imagem do mapa de superfícies e <i>landmarks</i> do cenário 1.	66

Lista de Tabelas

3.1	Alguns <i>Laser Range finders</i> 2D e 3D	18
3.2	Características técnicas do Sick LMS-291.	19
6.1	Classificação da nuvem de pontos	61
6.2	Número de ocorrências das diferentes classificações.	65

Esta página foi intencionalmente deixada em branco.

Acrónimos

2D	Bidimensional
3D	Tridimensional
ABD	<i>Adaptative Breakpoint Detector</i>
ATV	<i>All Terrain Vehicle</i>
CAS	<i>Center for Autonomous Systems</i>
CLM	<i>Concurrent Location and Mapping</i>
DARPA	<i>Defense Advanced Research Projects Agency</i>
DEM	<i>Digital Elevation Map</i>
DGPS	<i>Differential Global Positioning System</i>
ECEF	<i>Earth Centered, Earth Fixed</i>
EDEM	<i>Extended Digital Elevation Map</i>
GPS	<i>Global Positioning System</i>
HDL	<i>High Definition Lidar</i>
HFOV	<i>Horizontal Field Of View</i>
IMU	<i>Inertial Measurement Unit</i>
INS	<i>Inertial Navigation System</i>
IPP	Instituto Politécnico do Porto
ISEP	Instituto Superior de Engenharia do Porto
LADAR	<i>Laser Detection And Ranging</i>
LIDAR	<i>Light Intensity Detection And Ranging</i>
LINCE	<i>Land INtelligent Cooperative Explorer</i>
LMS	<i>Laser Mesurement System</i>
LRF	<i>Laser Range Finder</i>
LSA	Laboratório de Sistemas Autónomos
M-ELROB	<i>Military European Land-Robot Trial</i>
MLSM	<i>Multi Level Surface Map</i>
RADAR	<i>Radio Detection And Ranging</i>
RESOLV	<i>Reconstruction using Scanned Laser and Video</i>
SLAM	<i>Simultaneous Location and Mapping</i>
TIGRE	<i>Terrestrial Intelligent General proposed Robot Explorer</i>
UDP	<i>User Datagram Protocol</i>
VFOV	<i>Vertical Field Of View</i>

Esta página foi intencionalmente deixada em branco.

Capítulo 1

Introdução

Conteúdo

1.1	Âmbito da Dissertação	1
1.2	Motivação e Enquadramento	3
1.3	Objectivos	3
1.4	Estrutura da Dissertação	4

1.1 Âmbito da Dissertação

A navegação de veículos autónomos em ambientes não estruturados continua a ser um problema em aberto. A complexidade do mundo real ainda é um desafio. O relevo irregular, os objectos dinâmicos e pouco distintos e a inexistência de referências de localização são algumas das principais dificuldades.

A detecção de obstáculos é um dos componentes essenciais para o controlo de veículos autónomos. No caso de um cenário de operação *indoor* é frequente a assunção de que qualquer superfície superior ao nível do chão é um obstáculo. Contudo em ambientes *outdoor* esta simplificação não é possível, simplesmente porque o chão é demasiado irregular para ser definido por um plano.

O facto de não haver uma clara distinção geométrica entre superfície atravessável e não atravessável, *landmarks*, e obstáculos, obriga a uma análise qualitativa do modelo tridimensional, mais do que apenas um *gradient* de distâncias e descontinuidades. De uma forma genérica, podemos analisar se um determinado espaço é atravessável, simulando a

colocação de um modelo, com uma aproximação válida da plataforma, sobre o modelo tridimensional do espaço que se pretende analisar, e verificando se todas as rodas estão em contacto com o chão (no caso de ser um *robot* com rodas), bem como se a base do veículo está desimpedida. Este procedimento para além de ser computacionalmente muito exigente, requer um mapa de grande resolução.

O mapeamento digital é um termo genérico para o procedimento de converter um determinado conjunto de dados para um formato digital. No caso concreto desta dissertação, a expressão mapeamento digital significa o acto de representar o ambiente de operação de um *robot* num suporte digital. O sistema autónomo deverá ser dotado da capacidade de adquirir um conjunto de dados que permita a elaboração de um mapa. Uma das maiores

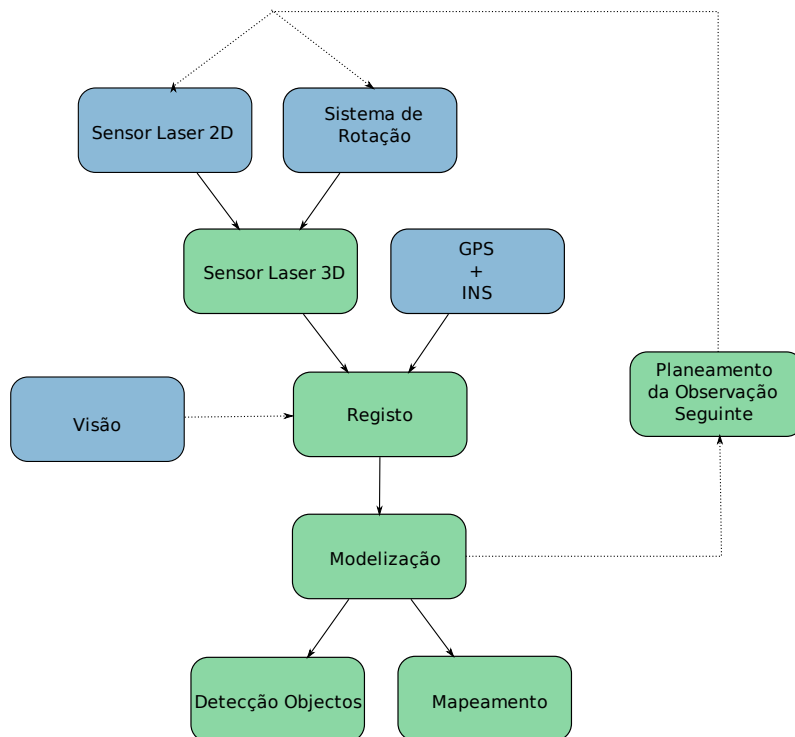


Figura 1.1: Processo de mapeamento.

dificuldades da utilização de informação tridimensional é o grande volume de dados a que é necessário aceder e interpretar. Esta etapa pode tornar-se um processo temporalmente dispendioso e muito exigente em termos computacionais, por isso é essencial catalogar, estruturar e simplificar este conjunto de dados por forma a tornar todo o processo mais eficiente, e utilizável numa navegação em tempo real. O processo que dá origem ao modelo tridimensional aproximado é genericamente descrito na Figura 1.1.

A fusão de dados entre a plataforma rotativa e o sensor *laser* 2D dá origem a uma nuvem de pontos tridimensional. Essa nuvem de pontos é depois georeferenciada na etapa de registo, com o acréscimo de informação proveniente dos receptores GPS (*Global Positioning System*) e INS (*Inertial Navigation System*). Terminada esta fase é depois iniciado o processo de modelização propriamente dito que pode englobar várias tarefas diferentes, como o pré-processamento, filtros e segmentação.

Existem um grande número de técnicas utilizada no mapeamento tridimensional, mas que na sua maioria não serão as mais apropriadas para mapeamento 3D em tempo real. Dado o custo de processamento ser muito elevado, algumas dessas técnicas só podem mesmo ser aplicadas *offline* em pós-processamento.

1.2 Motivação e Enquadramento

Ao longo da última década o Laboratório de Sistemas Autónomos (LSA) do ISEP-IPP (Instituto Superior de Engenharia do Porto - Instituto Politécnico do Porto) tem levado a cabo projectos de desenvolvimento de veículos autónomos em diversos cenários de operação [6] [7] [8] e de outros sistemas associados [9] [10]. Uma das linhas de orientação está direccionada na pesquisa e desenvolvimento de sistemas sensoriais que possibilitem o aumento da capacidade de percepção das plataformas robóticas. A necessidade de satisfazer em simultâneo o baixo consumo energético, e o desempenho computacional elevado são desafios que obrigam a uma optimização das arquitecturas de sistema, e algoritmos aplicados. O desenvolvimento de um sistema de modelização tridimensional visa acrescentar aos projectos LINCE (*Land INtelligent Cooperative Explorer*) [11] e TIGRE (*Terrestrial Intelligent General proposed Robot Explorer*) [12] maior autonomia e capacidade de exploração e mapeamento.

Os LINCE *Rover* são os *robots* que dão corpo ao projecto LINCE e estão dotados da capacidade de se movimentarem em diferentes tipos de ambientes, permitindo a exploração e inspecção em cenários não estruturados, tirando partido de técnicas de tele-operação e de cooperação multi-robótica.

O projecto TIGRE é a evolução natural do projecto LINCE, que visa a inspecção de cenários não estruturados mais alargados. A plataforma robótica do projecto TIGRE foi desenvolvida tendo como base a estrutura mecânica de uma ATV (*All Terrain Vehicle*)

comercial. Essa plataforma foi então dotada das capacidades de percepção, e de teleoperação já implementada e validada no projecto LINCE.

1.3 Objectivos

O objectivo desta dissertação visa contribuir para o desenvolvimento de um sistema de mapeamento tridimensional, baseado em informação de sensores *laser*, e em informação da navegação proveniente da plataforma autónoma à qual está associado.

A nossa contribuição vai ocorrer ao longo das seguintes linhas e etapas:

- Desenvolver um sistema de aquisição de informação tridimensional, que seja capaz de fornecer dados acerca da geometria e cor do espaço envolvente.
- Registar o conjunto de dados adquiridos de uma forma coerente, num referencial comum, e que permita a inspecção, processamento e elaboração de mapas tridimensionais.
- Detectar formas e estruturas no conjunto de dados recolhidos, em função da sua forma e dimensão, bem como da suas características cromáticas.
- Elaborar mapas simplificados, facilmente indexáveis, e com informação útil para a identificação de obstáculos, *landmarks* e espaços atravessáveis por uma plataforma robótica.

1.4 Estrutura da Dissertação

Nos capítulos seguintes são descritas algumas técnicas e procedimentos que visam a criação de mapas digitais tridimensionais. No segundo capítulo (capítulo 2) são apresentados alguns sensores utilizados para a aquisição de modelos tridimensional, bem como algumas das técnicas mais utilizadas para o processo de mapeamento e a sua aplicação em plataformas robóticas.

A apresentação do dispositivo de aquisição é feita no capítulo (capítulo 3), onde são também descritos os processo de registo e fusão de dados.

Terminado os processos de aquisição e registo são abordadas no capítulo 4 as técnicas de segmentação implementadas. Essas técnicas estão divididas em duas formas de análise

distintas: uma análise quantitativa e uma outra assente numa classificação qualitativa dos pontos.

No quinto capítulo (capítulo 5) podemos encontrar uma descrição de métodos para a obtenção de uma mapa de elevação digital aproximado, tendo como fonte de informação uma nuvem de pontos segmentada ou não.

Os resultados de todas estas metodologias são demonstradas no sexto capítulo 6. Neste capítulo podemos observar os mapas obtidos pela modelização de dois cenários de exemplo.

Por fim no último capítulo (capítulo 7) são enunciadas as conclusões e trabalho futuro.

Esta página foi intencionalmente deixada em branco.

Capítulo 2

Estado da Arte

Conteúdo

2.1	Tecnologias de Mapeamento	7
2.1.1	Riegl	8
2.1.2	Velodyne	8
2.2	Algoritmos e Processos de Mapeamento	10
2.3	Aplicações em Plataformas Robóticas	10
2.3.1	Kurt3D	11
2.3.2	RTS-HANNA	12
2.3.3	<i>Groundhog</i>	13
2.3.4	Stanley	14
2.3.5	Argo	15

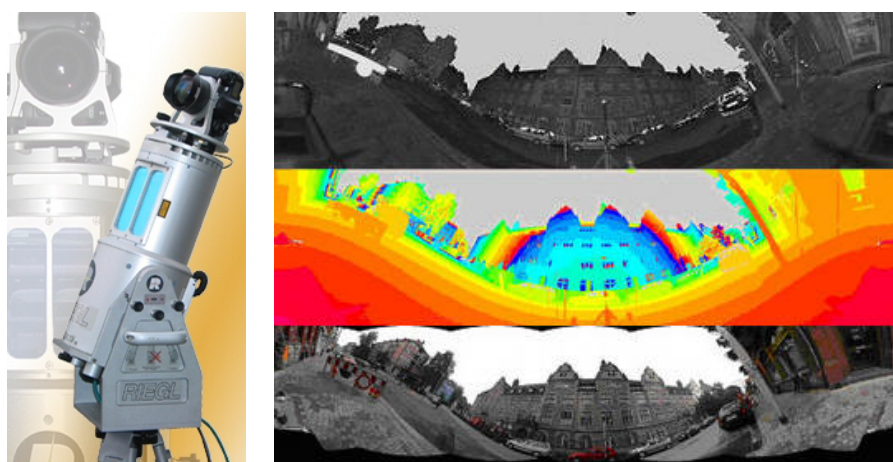
2.1 Tecnologias de Mapeamento

Ao longo dos últimos anos temos assistido ao desenvolvimento de múltiplos equipamentos que visam a aquisição de informação volumétrica de um determinado espaço. Destinados às mais diversas aplicações, como por exemplo os estudos arquitectónicos, o planeamento urbano ou mesmo no estudo de locais de interesse arqueológico, estas ferramentas foram rapidamente identificadas como um excelente método de percepção para aplicações robóticas. Em seguida, apresentamos dois dos mais conhecidos equipamentos utilizados para mapeamento tridimensional.

2.1.1 Riegl

A Riegl[13] é uma empresa austríaca que se dedica à pesquisa, desenvolvimento e produção de *laser range finders* (LRF), medidores de distância e *scanners*. Dentro de uma vasta gama de produtos podemos encontrar *laser range scanners*, preparados para operação num ponto estacionário, ou a bordo de um veículo aéreo ou terrestre. Estes sensores móveis, destinam-se essencialmente à recolha de mapas extensos de grandes áreas urbanas, florestais, plantações agrícolas, parques industriais, etc..

Com um alcance que pode ir das centenas de metros até dois quilómetros, estes sistemas são também capazes de detectar a reflectividade e a cor dos alvos, associando toda essa informação numa nuvem de pontos. Estes dispositivos são também utilizados por vários laboratórios, institutos ou universidades dedicados à investigação e desenvolvimento de sistemas autónomos como forma de mapear e localizar a plataforma robótica no seu espaço de operação.



(a) 3D *laser scanner* terrestre. (b) Nuvem de pontos de reflectividade, distância e cor.

Figura 2.1: Dispositivo terrestre Riegl e um exemplo de mapas gerados[13].

2.1.2 Velodyne

A empresa Velodyne[14] oferece um avançado sistema LIDAR (*Light Intensity Detection And Ranging*) de alta definição desenhado especialmente para tarefas de navegação de veículos autónomos, mapeamento, vigilância, automação industrial entre outras utilizações. Com um vasto campo de visão *Horizontal Field Of View* (HFOV) de 360° e com um *Vertical Field Of View* (VFOV) de 26.8° , com uma taxa de rotação de 5 ou 15 Hz

seleccionável pelo utilizador e com mais de 1.3 milhões de pontos por segundo, o Velodyne é sem duvida uma referencia entre este tipo de equipamentos.

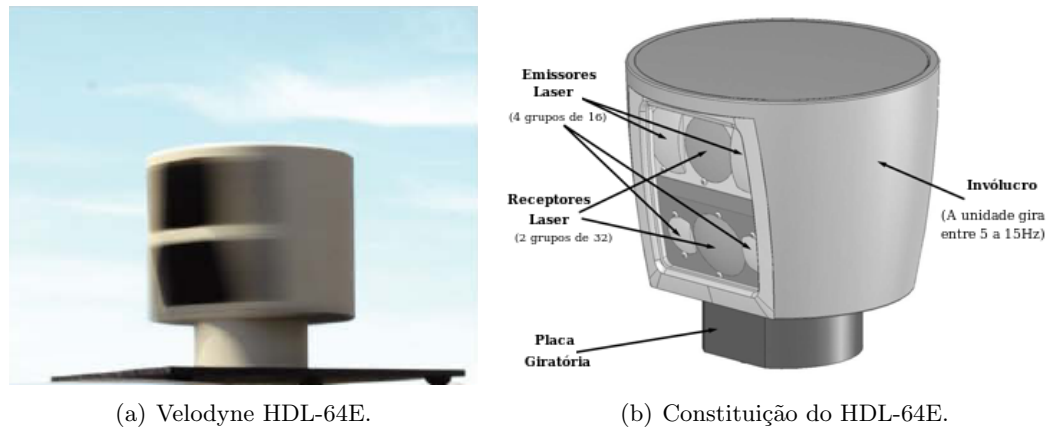


Figura 2.2: O Velodyne HDL-64E em funcionamento e a sua constituição[14].

Este dispositivo distingue-se da concorrência pelo facto de ter vários emissores *laser* e vários receptores(Figura 2.2(b)), em vez das abordagens mais comuns que têm apenas um emissor *laser* e espelhos rotativos. Este método permite recolher um número muito maior de pontos num curto espaço de tempo, possibilitando assim rodar toda a unidade em torno de si mesma a uma elevada velocidade. O alcance vai de 50 a 120 metros dependendo da reflectividade dos obstáculos, e a comunicação é feita a uma taxa de 100 *Mbs* por uma porta *ethernet* UDP(*User Datagram Protocol*).

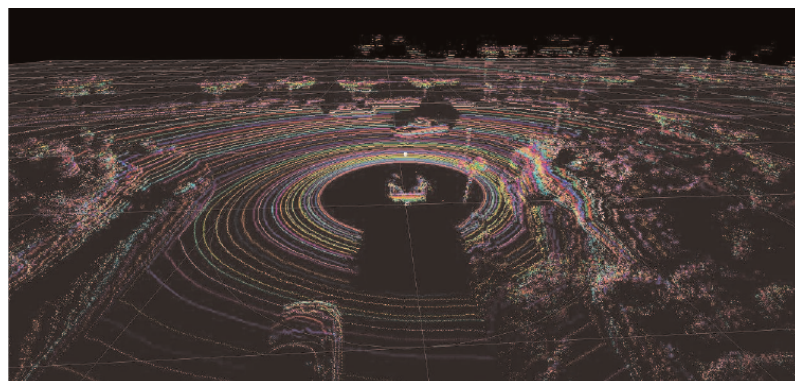


Figura 2.3: Nuvem de pontos recolhida numa área urbana[14].

2.2 Algoritmos e Processos de Mapeamento

Vários têm sido os grupos de investigação que têm centrado o seu trabalho no estudo da representação tridimensional do mundo. O projecto RESOLV (*Reconstruction using Scanned Laser and Video*) destina-se a modelizar ambientes *indoor* para realidade virtual ou tele-presença. O projecto *Avenue* desenvolveu um *robot* para adquirir modelos tridimensionais de espaços urbanos para fins de planeamento urbano e estudos arquitectónicos, mas em nenhum dos casos são usados os dados adquiridos para controlar ou estimar a posição do veículo. Nestes dois exemplos os sensores utilizados são LIDAR ou LADAR (*Laser Detection and Ranging*), dispositivos de elevada precisão mas também muito dispendiosos.

Uma outra corrente utiliza *2d laser scanners*, associando os dados recolhidos pelo sensor com a translação do *robot*, ou colocando-o sobre uma plataforma rotativa, é assim possível adquirir informação tridimensional do ambiente. Um dos percursores desta técnica foi *Sebastian Thrun* [5][15][16][2] que utilizando dois sensores de aquisição 2D, um montado na vertical e outro na horizontal conseguiu construir representações volumétricas do espaço envolvente. O sensor horizontal serve para determinar a posição relativa do *robot* no mapa já conhecido ou em construção, enquanto que o sensor colocado na vertical recolhe os dados que são usados para construir a representação tridimensional.

Oliver Wulf [1] [17], por outro lado, optou por colocar o mesmo sensor bidimensional sobre uma plataforma rotativa, rodando sobre um eixo vertical, este método permite a aquisição de dados estando o veículo parado, ou em movimento possibilitando a implementação de diferentes formas de navegação e controlo fazendo a aquisição em simultâneo com o deslocamento ou de uma forma do tipo *stop-scan-go*. O sucesso desta metodologia depende em grande parte da estimação da posição e atitude do veículo, e consequentemente do *laser range finder*. Os sensores inerciais ou giroscópios são assim componentes indispensáveis e determinantes para uma boa qualidade dos dados adquiridos.

2.3 Aplicações em Plataformas Robóticas

Uma das características fundamentais num veículo autónomo é a sua capacidade de perceber o seu espaço envolvente. É esta capacidade que lhe permite definir trajectórias que serviam a missão da forma mais eficiente possível, preservando a integridade

dos elementos que se opõe à sua passagem, sem esquecer as suas próprias características de locomoção e limitações associadas.

2.3.1 Kurt3D

O Kurt3D[18] (figura: 2.4) é um *robot* móvel semi-autónomo desenvolvido no *Institute of Computer Science* da universidade de *Osnabrück* na Alemanha, destinado a realizar operações de vigilância e de busca e salvamento em cenários urbanos. Esta versão é uma evolução da plataforma Kurt2 equipada com um *laser scanner* 3D. Tendo a capacidade de controlo e processamento de dados provenientes dos sensores, este *robot* é capaz de construir modelos tridimensionais do ambiente em que se encontra.



Figura 2.4: Kurt3D.

Neste projecto os principais tópicos em análise são:

6D SPLAM - Para construir um mapa é necessário que três tarefas corram num ciclo permanente :(1) Planeamento e deslocamento para o próximo ponto de aquisição de dados; (2) Localização da plataforma no mapa já construído; (3) Aquisição e registo de um novo conjunto de dados consistente com o modelo prévio.

Interpretação de dados - Construção de um modelo volumétrico do ambiente e interpretação geométrica dos dados para o reconhecimento de objectos.

Busca e Salvamento - Participação em eventos de demonstração deste tipo de valências.

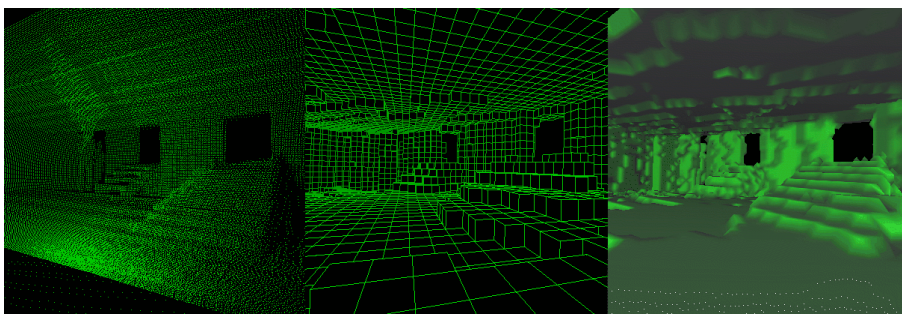


Figura 2.5: Exemplo de um cenário urbano “visto” pelo Kurt3D[19].

2.3.2 RTS-HANNA

O grupo *Real Time Systems* do *Institute for Systems Engineering* da Universidade de *Leibniz* em *Hannover* na Alemanha, desenvolveu em 2008 o *robot* autónomo *outdoor* chamado RTS-HANNA para a participação no M-ELROB (Military European Land-Robot Trial). É uma plataforma robótica baseada num veículo licenciado para circular na via pública e com tracção às quatro rodas, sendo por isso ideal para operar em zonas urbanas bem como em terrenos *off-road*. O RTS-HANNA está equipado com vários sensores para tele-operação, operação semi-autónoma e operação completamente autónoma. Os principais sensores são um par de *3D Laser range scanners*, para percepção do ambiente de operação. Este par de sensores é constituído cada um deles por dois SICK LMS-291 colocados horizontalmente em sentidos opostos sobre uma plataforma rotativa, como podemos ver na Figura 2.6(a). Com um resolução de dois graus na horizontal e de um grau na vertical, este sistema obtém um varrimento tridimensional de 360 graus em 1.2 segundos, com mais de 60 000 pontos. Os dados volumétricos são usados para detecção de obstáculos em navegação autónoma, reconhecimento de objectos bem como localização. Em adição múltiplas câmaras incluindo, uma de infravermelhos, *Differential Global Positioning System* (DGPS) e sensores inerciais são usados para controlar o veículo. Os algoritmos de navegação baseiam-se na fusão dos dados provenientes do *laser scanner*, sensor inercial e DGPS.

No modo de operação semi-autónomo, o RTS-HANNA é usado para seguir automaticamente uma pessoa ou outro veículo a uma distância preestabelecida. Diferenciando-se das abordagens mais comuns, esta plataforma não necessita de uma comunicação entre o líder e o perseguidor. O componente base da percepção e reconhecimento automáticos do líder

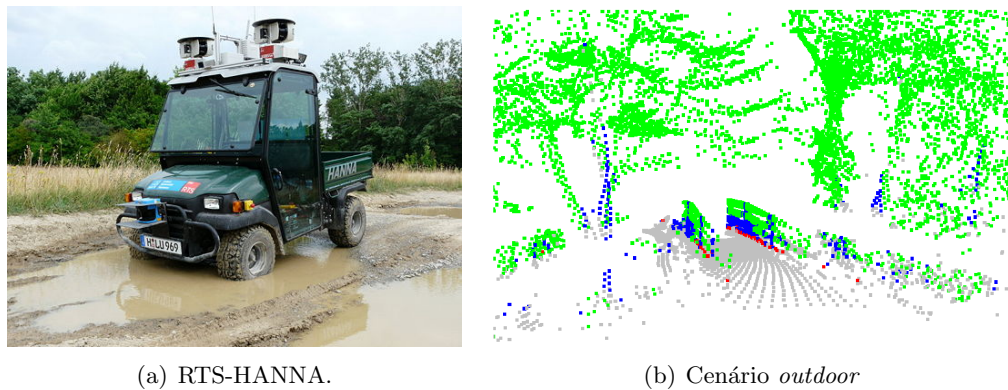


Figura 2.6: Veículo autónomo RTS-HANNA e um mapa produzido pelo mesmo.

e da sua movimentação são os *lasers scanners* 3D, sendo por isso necessária uma directa linha de vista entre líder e seguidor. Estimando a velocidade e posição do líder, o *robot* é capaz de gerar os comandos de velocidade e de direcção por forma a seguir a sua trajectória à distância desejada. Na figura 2.6(b) podemos observar a segmentação efectuada de acordo com a classificação de pontos de superfície, *landmark* e suspensos (pontos a cinzento, azul e verde respectivamente). Neste método de mapeamento é também extraído um mapa bidimensional que vemos representado nos pontos a vermelho. Para isso é extraído um ponto de cada varrimento vertical que são depois colocados no mesmo plano obtendo assim um mapa bidimensional. A construção deste mapa de duas dimensões permite a utilização das técnicas mais comuns de localização utilizadas para ambientes *indoor*.

2.3.3 *Groundhog*

O sistema robótico *Groundhog*[16][2] foi desenvolvido para adquirir mapas volumétricos de minas subterrâneas. Este sistema baseia-se num *robot* móvel operado remotamente e munido de *laser range finders* colocados horizontal e verticalmente. O *Groundhog* não dispõe de informação odométrica, dado o terreno irregular nem de informação sobre o seu posicionamento global, dado que não existe sinal GPS no interior das minas. Estas dificuldades acrescidas obrigam a que a construção de mapas coerentes se baseiem unicamente nos dados recolhidos pelos sensores *laser*. A nuvem de pontos obtida em cada instante é processada com técnicas baseadas em *Simultaneous Location and Mapping* (SLAM) e *Iterative closest Point* (ICP) para a construção de um modelo coerente ao longo do percurso e precavendo eventuais múltiplas observações do mesmo espaço.

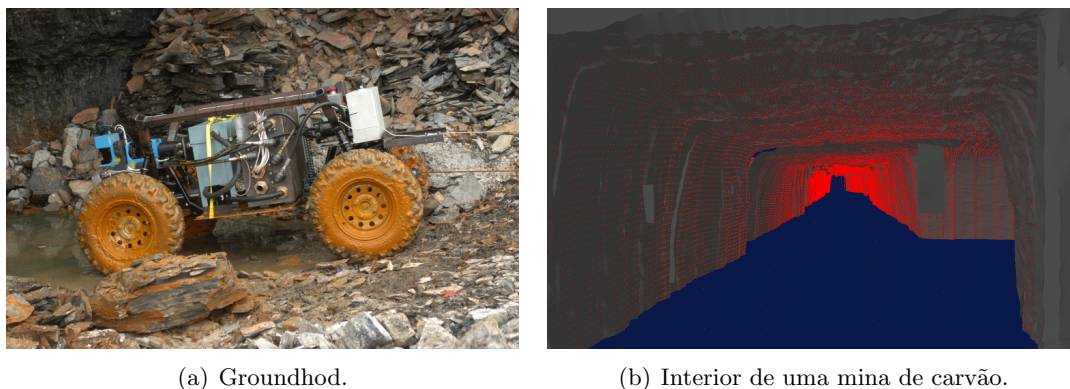


Figura 2.7: Fotografia do Groundhog e uma representação volumétrica de uma mina[20].

2.3.4 Stanley

O *Stanley*[15] é uma plataforma robótica desenvolvida pela Universidade de *Stanford*, para participar no DARPA (*Defense Advanced Research Projects Agency*) *Grand Challenge* tendo vencido a edição de 2005 disputada no deserto de *Mohave* na Califórnia, tendo percorrido 132 milhas em menos de 7 horas e sem qualquer intervenção humana.

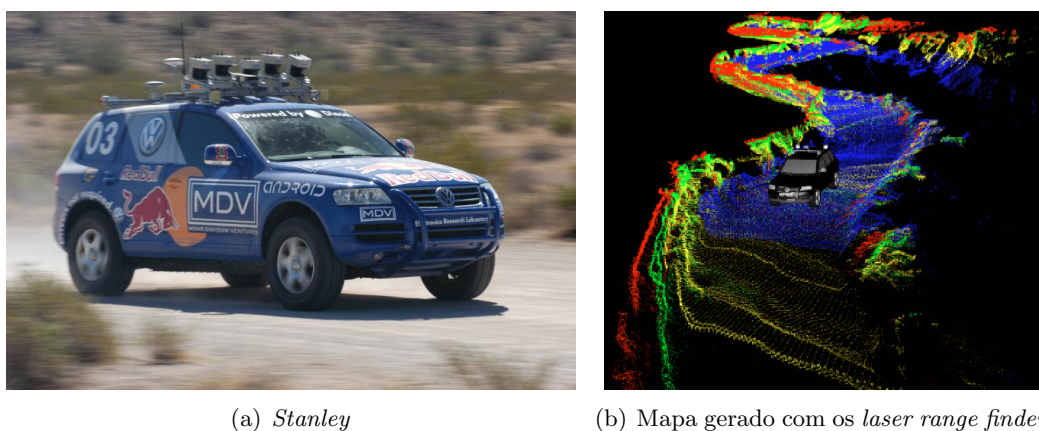


Figura 2.8: O *robot Stanley* e um modelo volumétrico do terreno[15].

O *Stanley* é baseado no *Volkswagen Touareg R5*. Com um interface electrónico, é possível controlar directamente a acção do acelerador e do travão e com um motor DC acoplado à coluna de direcção é possível controlar electronicamente a movimentação do veículo. Para o modo de navegação autónoma, o *Stanley* depende de um GPS, de um *Radio Detection And Ranging* (RADAR), e de 5 *laser range finders* colocados no tejadilho com ângulos de inclinação ligeiramente diferentes para cobrir toda a frente do *robot* num alcance até 25 metros. Os dados provenientes dos 5 sensores são fundidos entre si, e

também com a informação do GPS e RADAR para construir um único mapa coerente. Em função do ponto de destino, dos obstáculos detectados, e do relevo do terreno é gerada a trajectória do *robot*.

2.3.5 Argo

O *Center for Autonomous Systems* (CAS) é um centro de investigação em sistemas autónomos em *Sidney* na Austrália. Entre muitos outros projectos desenvolveu a plataforma robótica que podemos observar na Figura 2.9. Este veículo[21][22] desenhado para navegar autonomamente é dotado de um sensor inercial, GPS, *laser range finder* e de uma câmara. A informação proveniente de cada um destes sensores é processada por forma a ser fundida e representada segundo o mesmo referencial.

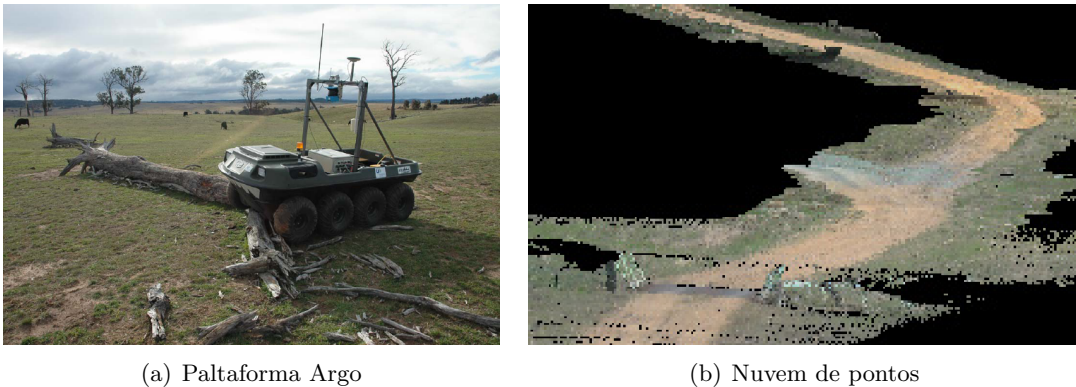


Figura 2.9: Plataforma ARGO do laboratório CAS e nuvem de pontos[23].

Dessa forma, a nuvem de pontos recolhida pelo *laser*, que nos dá informação sobre as características volumétricas do espaço envolvente, pode ser associado com a informação da câmara, formando uma nuvem de pontos colorida de acordo com a geometria e características cromáticas dos objectos detectados.

Esta página foi intencionalmente deixada em branco.

Capítulo 3

Aquisição e Registo de Dados

Conteúdo

3.1 Sistema de Aquisição de Informação	17
3.1.1 Sensor <i>Laser</i>	17
3.1.2 Sistema de Rotação	20
3.1.3 Erro do Sistema de Rotação	20
3.2 Registo	21
3.2.1 Rotações e Translações	22
3.2.2 Sistema de Posicionamento Global	25
3.2.3 Sistema de Navegação Inercial	25
3.3 Fusão de Dados	25
3.3.1 Sincronização	26
3.3.2 <i>Laser</i> e Visão	27

3.1 Sistema de Aquisição de Informação

3.1.1 Sensor *Laser*

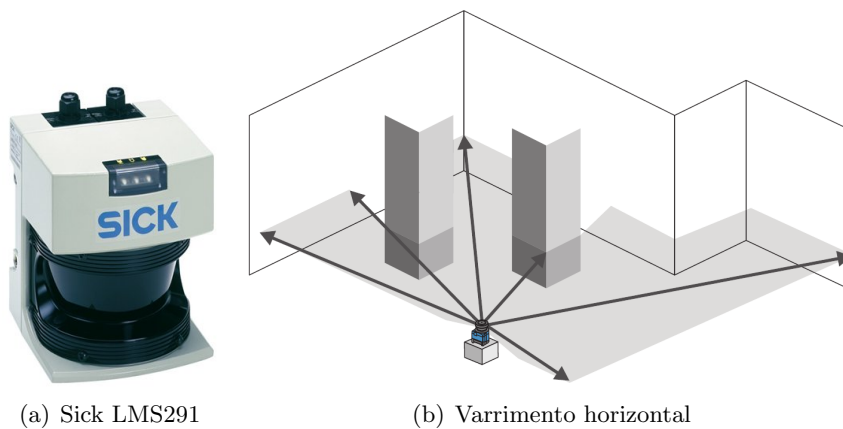
Vários são os sensores *laser*, de medição de distâncias, que se destinam à tarefa pretendida de mapeamento. Como podemos ver na Tabela 3.1.1 alguns dos diferentes dispositivos existentes têm características fundamentais bem diversas, que poderão significar mais variáveis ou deficiências em função das aplicações a que se destinam. Embora todos eles sejam incluídos em aplicações na área da robótica é importante destacar os sensores Velodyne[14]

Fabricante	Iiris 3D	IBEO Lux	Velodyne	Riegl Z420	Sick lms291	Leica 4400
Classe <i>Laser</i>	1	1	1	1	1	1
Comprimento de Onda (nm)	1540	905	905	904	905	905
Alcance (m)	800	200	120	200	80	700
Precisão (mm)	7	40	20	10	10	20
Abertura Angular V.	40	3.2	26.8	360	-	360
Abertura angular H.	40	100	360	80	180	80
Dimensões	3D	3D	3D	3D	2D	3D

Tabela 3.1: Alguns *Laser Range finders* 2D e 3D

, desenhado para ser integrado em veículos autónomos e o sensor IBEO Lux[24], que foi desenvolvido como o intuito de ser utilizado em veículos não autónomos, mas como sistema de segurança e de auxílio à condução.

O dispositivo de aquisição utilizado é o SICK *Laser Measurement System* (LMS)[25]. Trata-se de um sistema de medição *laser* inicialmente projectado para aplicações industriais. O sensor recolhe informação acerca do espaço que o envolve a duas dimensões, com um campo de visão radial, usando feixes *laser* na gama de infra-vermelhos. O sistema não requer qualquer tipo de reflectores ou marcos de posição.

Figura 3.1: *Laser range finder* Sick LMS-291[25]

O feixe *laser* pulsado é desviado por um espelho interno rotativo. Este sistema permite fazer um varrimento em leque utilizando apenas um emissor e um receptor. O processo de reflexão dos dois feixes (emitido e reflectido) é ilustrado na Figura 3.2(a). Os dados relativo à medição são disponibilizados em tempo real através do interface de comunicação RS232 ou RS422.

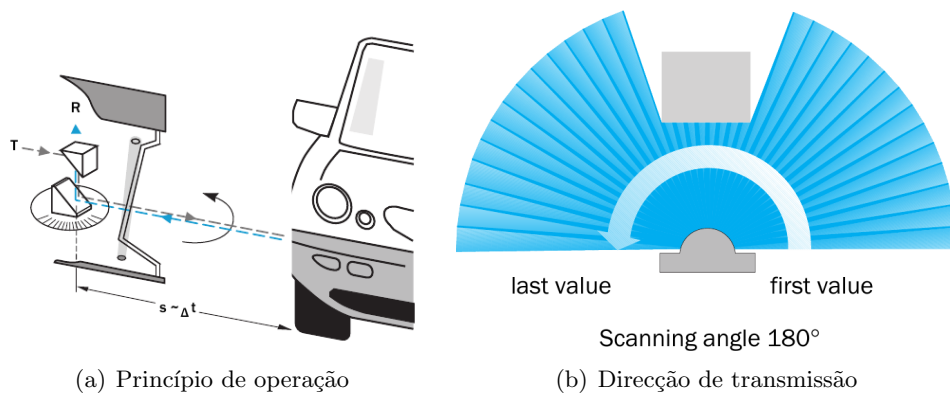


Figura 3.2: Sick LMS-291, princípio de operação e varrimento[25]

O LMS-291 é dotado de um sistema automático de correção de nevoeiro que permite, em utilizações *outdoor*, eliminar as falsas leituras recorrentes de gotas de água ou flocos de neve. Esta correção é conseguida através de um processo baseado na análise da orientação dos pixels do sinal recebido.

Na tabela 3.1.1 podemos ver algumas das principais características técnicas do referido sensor:

Abertura Angular:	180°
Resolução angular:	1 ... 0.25°
Tempo de Resposta:	13 ... 53 ms
Resolução:	10 mm
Erro sistemático:	+/- 15 mm
Erro Estatístico(1 sigma):	5 mm
Classe do <i>Laser</i> :	1
Grau de Protecção:	IP 65
Alcance Máximo:	80 m
Máximo alcance com 10% reflectividade:	30m
Interface de Comunicação:	RS-232, RS-422
Velocidade de Transmissão:	9,6 / 19,2 / 38,4 / 500 kBaud
Alimentação:	24 V DC +/- 15%
Potência:	20 W
Peso:	4.5 kg
Dimensões (C x L x A):	156 x 155 x 210 mm
Com Correção de Nevoeiro	Sim

Tabela 3.2: Características técnicas do Sick LMS-291.

3.1.2 Sistema de Rotação

O sistema de rotação foi desenvolvido com o intuito de transformar o LRF 2D disponível num sistema tridimensional. Para conseguir isso, o referido sensor foi colocado na posição horizontal sobre uma plataforma rotativa. O ângulo de rotação β , de acordo com a Figura 3.3, acrescenta a terceira dimensão necessária à construção da nuvem de pontos.

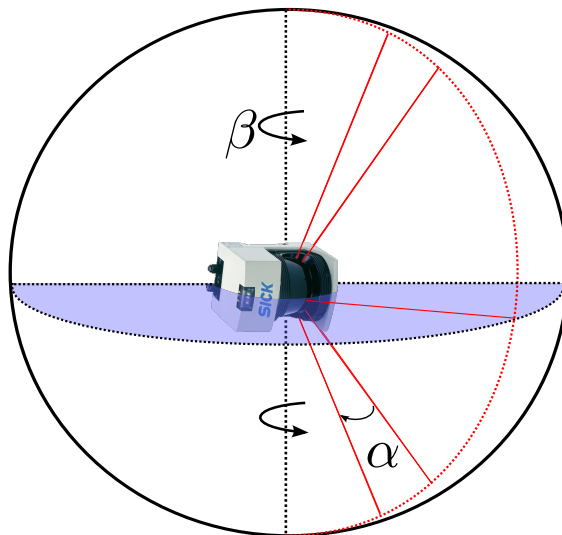


Figura 3.3: Esfera de varrimento do LRF tridimensional.

Supondo que o centro de rotação do espelho do LRF coincide com o eixo de rotação do sensor, então os vários varrimentos verticais de ordem j poderiam ser expressos num sistema de coordenadas cartesianas de acordo com a equação 3.1, em que o index de cada ponto de um varrimento individual é i , e em que α é o ângulo de passo da rotação do espelho interno β é o ângulo de passo do sistema de rotação.

$$\begin{cases} \hat{x}_{ij} = r_{ij} \cos\beta_j \cos\alpha_i \\ \hat{y}_{ij} = r_{ij} \sin\beta_j \cos\alpha_i \\ \hat{z}_{ij} = r_{ij} \sin\alpha_i \end{cases} \quad (3.1)$$

3.1.3 Erro do Sistema de Rotação

A análise de erros sistemáticos e a compensação dos efeitos provocados por estes desempenham um papel essencial em 3D *scanning*. Neste caso em particular, a sincronização entre o motor responsável pelo movimento de rotação da base e o sensor *laser* tem uma

grande influência na qualidade da nuvem de pontos tridimensional, tal como mencionado por *Wagner* [26] e *Herbert* [27]. Em especial o varrimento a grande velocidade é fortemente afectado por uma pobre sincronização, como podemos ver no exemplo seguinte. Consideremos um *scanner* 3D que utilize um sensor *laser* como o Sick LMS com uma resolução de 1° e com uma velocidade de rotação de $\omega_s = 75^\circ/s$ do motor. O erro máximo sistemático e devido a uma falha de sincronismo de $\Delta t = 100 \text{ ms}$ para um objecto a uma distância de $d = 10 \text{ m}$ pode ser calculada assim:

$$e = d \cdot \sin(\omega_s \cdot \Delta t) \quad (3.2)$$

resultando num erro sistemático máximo de $e = 1.3 \text{ m}$ para o exemplo descrito. É desta forma essencial implementar um método em tempo real que permita obter blocos de dados consistentes no tempo.

3.2 Registo

O processo de mapeamento é um processo contínuo de aquisição e processamento de dados, em que o modelo volumétrico do espaço a representar aumenta com a soma de varrimentos tridimensionais sucessivos. Por sua vez, cada varrimento tridimensional pode ser visto como uma sequência de varrimentos bidimensionais, em que cada *scan* individual sofre uma rotação e/ou translação face ao anterior.

Ao processo encarregue por formar um mapa conciso e coerente a partir de múltiplos varrimentos individuais dá-se o nome de registo. Genericamente é o processo responsável por referenciar todos as leituras individuais num único sistema de coordenadas, tendo em conta as transformações entre a posição e atitude do sensor em relação à plataforma robótica e a movimentação da mesma.

Numa abordagem *stop-scan-go* são gerados vários mapas locais que são o resultado de observações a partir de pontos e orientações diferentes. Em seguida é formado um mapa global integrando os diferentes mapas locais.

3.2.1 Rotações e Translações

De uma forma genérica, a transformação de um ponto P^a no referencial \mathbf{a} para um outro referencial \mathbf{b} , segundo um ordem de rotação *yaw* (ψ), *pitch* (θ) e *roll* (ϕ) respectivamente, é dada por:

$$P^b = P_a^b + C_a^b P^a \quad (3.3)$$

O ponto P^a é o ponto original em \mathbf{a} , P^b é o ponto transformado que pretendemos obter referenciado em \mathbf{b} , P_a^b é o offset entre os referenciais \mathbf{a} e \mathbf{b} (a localização da origem de \mathbf{a} em \mathbf{b}) e C_a^b é a matriz que expressa as rotações segundo os três eixos capazes de alinhar \mathbf{b} com \mathbf{a} .

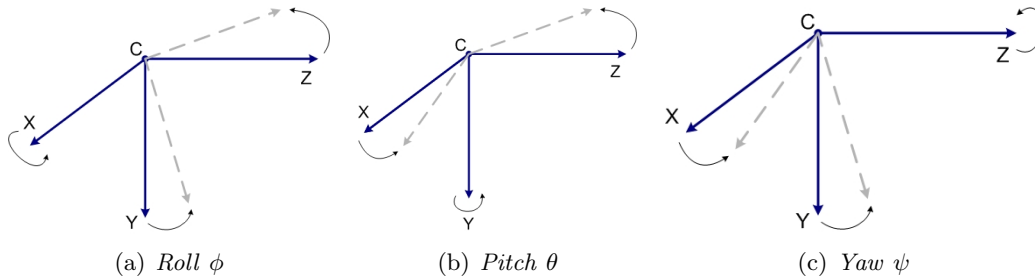


Figura 3.4: Rotações em torno dos três eixos cartesianos.

A matriz de rotação C_a^b é definida por:

$$C_a^b = \begin{bmatrix} \cos\psi\cos\theta & \cos\psi\sin\theta\sin\phi - \sin\psi\cos\phi & \cos\psi\sin\theta\cos\phi + \sin\psi\sin\phi \\ \sin\psi\cos\theta & \sin\psi\sin\theta\sin\phi + \cos\psi\cos\phi & \sin\psi\sin\theta\cos\phi - \cos\psi\sin\phi \\ -\sin\theta & \cos\theta\sin\phi & \cos\theta\cos\phi \end{bmatrix} \quad (3.4)$$

Os modelos tridimensionais criados, dependem da associação de dados provenientes de diferentes sensores que fazem parte da plataforma robótica. Esses sensores estão dispostos em locais diversos, na plataforma, e com diferentes orientações, o que não permite relacionar directamente os dados de cada um. É assim, necessário referenciar todos esses sensores a um ponto comum. Esse ponto é o centro volumétrico da estrutura, dados que é também o ponto escolhido para referenciar o veículo num mapa local ou geo-referenciado.

As transformações necessárias para obter uma nuvem de pontos geo-referenciada e coerente ao longo da trajectória, são expressas na figura 3.5, em que \mathbf{T} indica uma transformação de um referencial para outro. T_{LR} é a transformação do centro de rotação do *laser* para o centro de rotação do motor, T_{RP} é a transformação do centro de rotação do

motor para o centro da plataforma e T_{PM} é a transformação do centro da plataforma e o referencial do mundo.

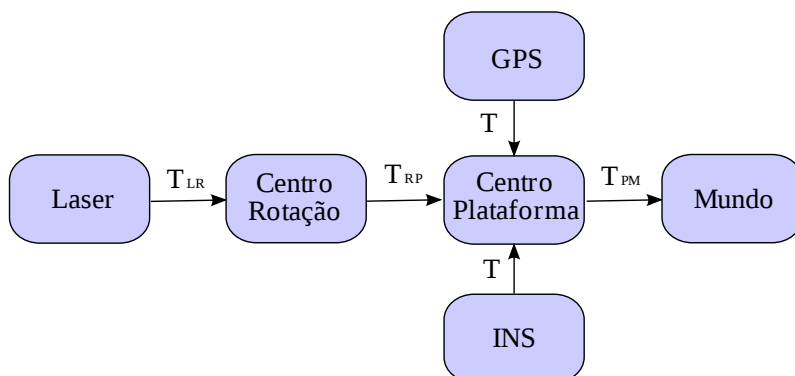


Figura 3.5: Transformadas de referenciais entre o sensor *laser* e o mundo.

3.2.1.1 Transformada *Laser* para Centro de Rotação

A transformada T_{LR} expressa a transformação entre o centro do espelho rotativo do sensor e o centro de rotação do suporte do sensor. É uma transformação fixa que nunca sofre alterações e depende exclusivamente da disposição física do sensor em relação ao seu suporte.

$$P^R = P_L^R + C_L^R P^L \quad (3.5)$$

O *offset* P_L^R pode ser facilmente medido, tendo como base as dimensões do sensor e a localização do espelho no seu interior. A rotação C_L^R expressa a posição do sensor em relação ao seu suporte, ou seja, este sofre uma rotação de -90° segundo o eixo das abcissas.

3.2.1.2 Transformada Centro de Rotação para Centro Plataforma Robótica

A transformada T_{RP} expressa a transformação entre o centro de rotação do suporte e o centro volumétrico da plataforma robótica. Esta transformação depende da posição e orientação que o suporte apresenta dentro da plataforma, mas também, e principalmente, a rotação em *yaw* que sofre o suporte.

$$P^P = P_R^P + C_R^P P^R \quad (3.6)$$

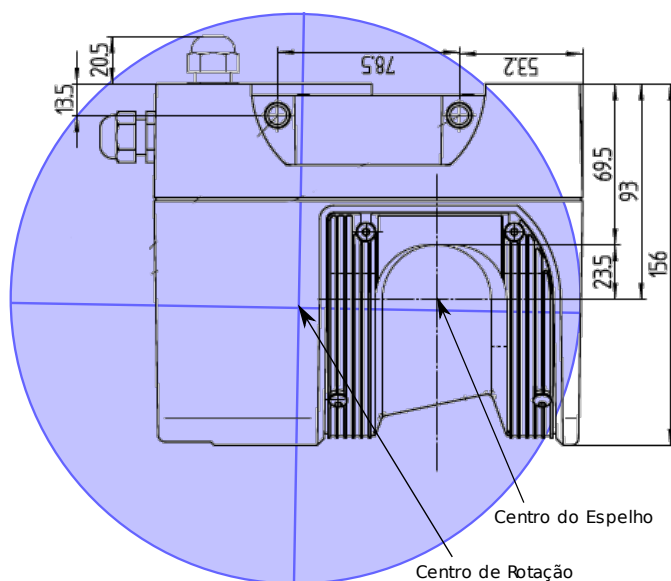


Figura 3.6: Disposição física do sensor em relação ao suporte.

A rotação C_R^P expressa não só a orientação física do suporte, mas também o movimento rotativo em yaw do sensor. Embora seja possível medir directamente o *offset* P_R^P , a medição directa da rotação C_R^P do suporte é muito mais difícil. Por forma a obter uma transformação mais precisa, é necessário utilizar uma técnica de calibração de dados como a demonstrada por *Underwood* [23]. O método consiste na observação de uma estrutura geométrica conhecida, ou parcialmente conhecida, em que os *offsets* são obtidos pela minimização dos erros entre os dados lidos pelo sensor e a estrutura conhecida.

3.2.1.3 Transformada Centro da Plataforma Robótica para o Mundo

A transformada T_{PM} é responsável por geo-referenciar a nuvem de pontos gerada. Trata-se da transformação do referencial da plataforma para o mundo, e isso é conseguido usando a informação proveniente do sistema de posicionamento global e do sistema inercial.

$$P^M = P_P^M + C_P^M P^P \quad (3.7)$$

A atitude do veículo apresenta os ângulos de *yaw*, *pitch* e *roll* essenciais para obter C_P^M e a posição (dada pelo sistema de coordenadas da mão direita) providencia o *offset* P_P^M .

3.2.2 Sistema de Posicionamento Global

O *Global Positioning System* (GPS) tornou-se um métodos mais comuns para a navegação. O baixo custo dos receptores mais recentes aliado à considerável exactidão obtida, proporcionou uma utilização generalizada em aplicações terrestres, marítimas e aéreas. Inicialmente desenvolvido para aplicações militares pelo Departamento de Defesa Norte-Americano, o GPS rapidamente foi adoptado pela comunidade civil através da utilização do sinal civil disponibilizado. Para qualquer utilizador com um receptor GPS, o sistema fornece informação precisa de localização e tempo.

O sistema GPS é composto por três componentes: satélites que orbitam à volta da Terra; estações de monitorização e controlo; e os receptores de GPS de cada utilizador. Os satélites emitem sinais do espaço que são recebidos e identificados pelos receptores. Cada receptor fornece então localização tridimensional (latitude, longitude e altitude) mais o tempo.

3.2.3 Sistema de Navegação Inercial

O sistema de navegação inercial é um sistema de navegação *dead reckoning* que se destina a fornecer informação acerca da posição e velocidade de um veículo. O seu componente fundamental é um *Inertial Measurement Unit* (IMU), trata-se de uma unidade constituída por três acelerómetros e três giroscópios montados numa tríade ortogonal, podendo ainda em alguns casos dispor de um magnetómetro.

O princípio de funcionamento da navegação inercial, tem por base a lei de movimento de *Newton*, que diz que um corpo mantém-se num estado de repouso ou de movimento uniforme até que seja perturbado pela acção de uma força externa. A aplicação dessa força gera aceleração, que é medida pelos acelerómetros. O valor da aceleração é então integrada duas vezes obtendo o valor da posição em relação às condições iniciais. Depois dos dados processados o INS fornece informação acerca da posição, velocidade e atitude.

3.3 Fusão de Dados

A fusão de dados é, de uma forma simples, o processo que se encarrega de associar consistentemente o conjunto de dados proveniente de múltiplas fontes, permitindo obter informação mais precisa e de uma forma mais eficiente do que obtida e processada

individualmente.

Estas diferentes fontes de informação têm taxas de comunicação e de amostragem muito diferentes, o que dificulta a associação. A forma mais simples de contornar este problema é a de fazer uma leitura alternada de cada sensor, o problema deste processo é que, para além de não termos uma taxa consistente, ficamos limitados pelo dispositivo que tiver a menor frequência de amostragem.

Uma possibilidade de ultrapassar estes problemas é fazer a leitura dos vários dispositivos separadamente, e a taxas diferentes num sistema operativo de tempo real. Cada leitura deverá ser associada a uma etiqueta de tempo, tirando partido da máxima frequência de amostragem de cada um, e permitindo, numa segunda etapa, processar todo o conjunto de dados.

3.3.1 Sincronização

Por forma a atribuir um carimbo temporal correcto, e para compensar erros sistemáticos de leitura, o conhecimento acerca do princípio de funcionamento, e da implementação do sensor é essencial. Dado que a implementação depende do sensor em si, vamos analisar o LRF da série LMS 291 da empresa Sick. O Sick LRF bidimensional consiste num dispositivo de medição *laser* de uma dimensão, e num espelho que roda continuamente a uma velocidade constante. Os 180° de abertura angular separam o processo em dois *slots* temporais. O primeiro é usado para emitir e receber o feixe *laser*, o segundo *slot*, em que não é possível adquirir dados porque o espelho aponta para o interior do dispositivo, é usado para comunicação e transferência de dados. A resolução angular $\Delta\varphi$ pode ser calculada através da taxa de amostragem f_L e da velocidade de rotação do espelho ω_m da seguinte forma:

$$\Delta\varphi = \frac{\omega_m}{f_L} \quad (3.8)$$

resultando numa resolução angular de $\Delta\varphi = 1^\circ$ para uma frequência de amostragem de $f_L = 27kHz$ e uma velocidade de rotação de $\omega_m = 75\pi \text{ rad/s}$ (tempo de rotação de 13.3 ms). Como a taxa de amostragem e a velocidade de rotação são constantes, para atingir uma resolução angular de 0.5° ou de 0.25° são agregados dois varrimentos com resolução de 1° desfasados de 0.5° , ou quatro varrimentos com desfasamentos de 0.25° .

Este padrão de leitura não afecta o tempo de leitura, nem a precisão de varrimentos estáticos, mas é um problema para o processo de *timestamping*, porque só depois de

efectuar todos os *scans* é que os dados são transferidos. Para ultrapassar esta dificuldade, o sensor Sick LMS oferece a solução ideal, que é um modo de funcionamento *raw data*. Ao contrário do modo de funcionamento normal, em que só no final dos vários varrimentos individuais é que o conjunto de dados é transmitido, neste modo de funcionamento os dados são transmitidos logo após a aquisição de cada *scan*.

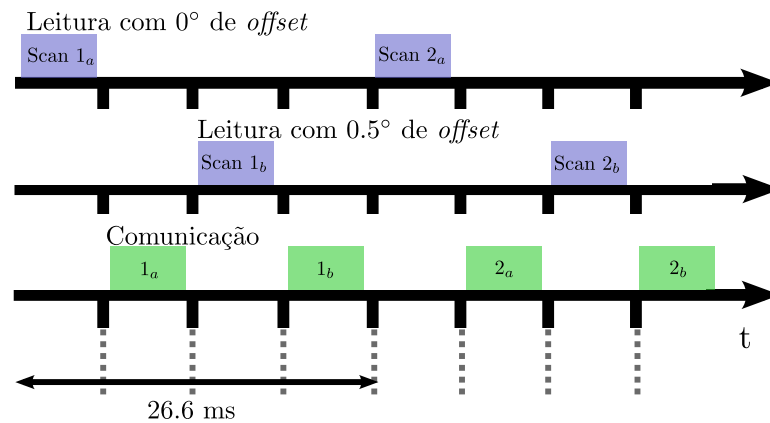


Figura 3.7: Comunicação do Sick LMS em modo de *raw data*.

Na Figura 3.7 vemos o fluxo temporal de um *scan* com uma resolução de 0.5° no modo de funcionamento *raw data*. Os *timestamps* são assim mais precisos, respeitando a sequência de aquisição, e permitindo consequentemente uma melhor compensação de erros sistemáticos.

3.3.2 Laser e Visão

Várias técnicas foram desenvolvidas ao longo dos últimos anos, por forma a aumentar a informação presente nas representações tridimensionais do espaço. Algumas dessas técnicas de modelização focam a sua abordagem na inclusão de cor na informação recolhida pelos sensores *laser*, dando origem à construção de mapas tridimensionais coloridos. Muitas aplicações podem lucrar com este tipo de abordagem, fundamentalmente em situações em que a interpretação humana seja necessária, incluindo estudos arqueológicos, arquitectónicos ou planeamento urbano.

Todas estas técnicas usam uma plataforma estacionária de onde adquirem mapas tridimensionais densos rodando sobre um ou mais eixos a plataforma sensorial. Embora estas técnicas sejam bem sucedidas para determinadas aplicações, elas não são especialmente vocacionadas para utilizações em tempo real, particularmente em plataformas móveis.

Uma das técnicas desenvolvidas para a calibração estrínseca de uma câmara e de um *laser range finder* foi apresentada por Pless e Zhang[28]. Este método consiste na observação de um padrão de xadrez de múltiplas poses, relacionando a imagem e o varrimento *laser* por forma a obter a posição e orientação relativas dos dois sensores.

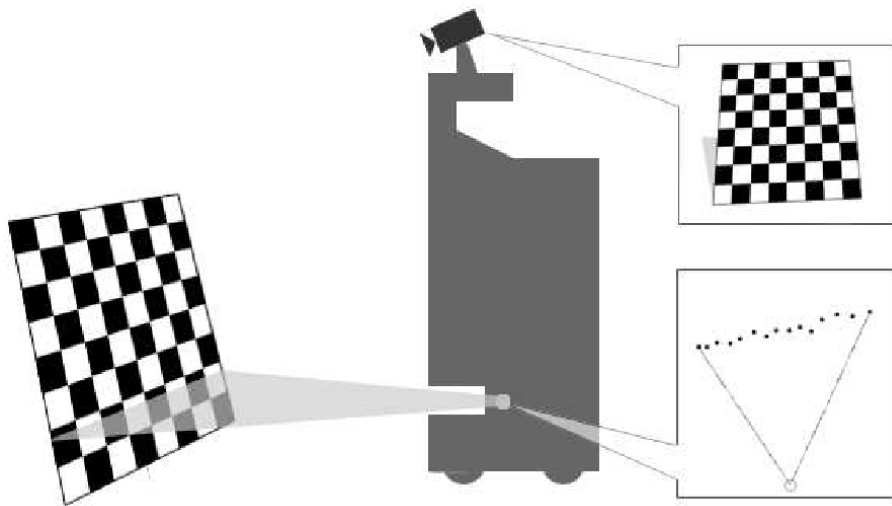


Figura 3.8: Padrão de xadrez observado simultaneamente pelo *laser* e câmara.

O processo de calibração pode ser dividido em três etapas, uma primeira solução linear, seguida de uma optimização não linear e finalmente uma optimização global em que são refinados os parâmetros extrínsecos da câmara. Estes parâmetros extrínsecos ditam a rotação e translação do *laser* em relação à câmara. Sendo então possível mapear os pontos do LRF no referencial da câmara.

$$P^C = P_L^C + C_L^C P^L \quad (3.9)$$

Em seguida é necessário transpor os pontos do referencial da câmara para o referencial da imagem. O primeiro passo é normalizar o ponto do *laser* no referencial da câmara P^C em relação ao referencial da profundidade z no ponto $P^{C'}$ normalizado, de acordo com as equações seguintes:

$$P^C = [X \ Y \ Z]^T \quad (3.10)$$

$$P^{C'} = \left[\frac{X}{Z} \ \frac{Y}{Z} \ 1 \right]^T \quad (3.11)$$

O ponto normalizado é então transformado no referencial da imagem através da matriz de rotação k . Para determinar esta matriz é necessário conhecer os parâmetros intrínsecos da câmara (distância focal $f = [f_1, f_2]$, o ponto principal $p = [p_1, p_2]$ e o factor de torção α) sendo para isso utilizada uma técnica de calibração descrita por *Stoyanov*[29]. Este dados são então utilizados para construir a matriz de rotação K da seguinte forma:

$$K = \begin{bmatrix} f_1 & \alpha f_2 & p_1 \\ 0 & f_2 & p_2 \\ 0 & 0 & 1 \end{bmatrix} \quad (3.12)$$

Com base na matriz de rotação k podemos então expressar os pontos *laser* no referencial da câmara já normalizado no referencial da imagem.

$$P^I = KP^{C'} \quad (3.13)$$

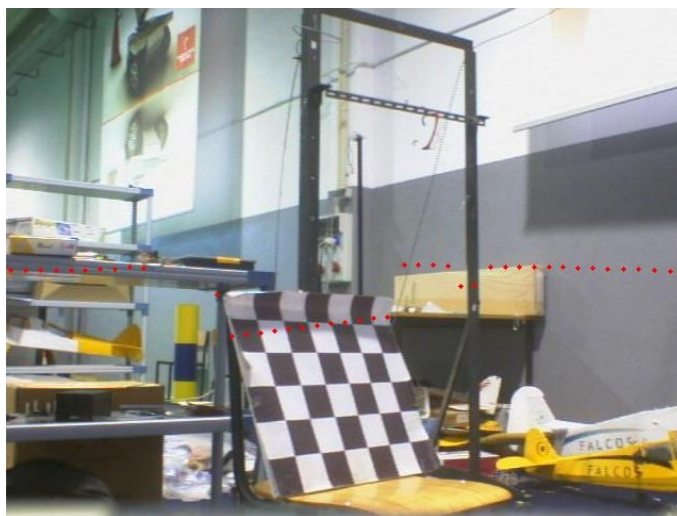


Figura 3.9: Resultado da calibração extrínseca do *laser* e câmara.

Terminado o processo de calibração é possível mapear a *scanline* do *laser* na imagem introduzindo informação relativa à distância dos obstáculos ou estruturas detectadas.

Na Figura 3.9 podemos ver a representados a vermelho os pontos onde incidem os feixes *laser* do LRF. Desta forma podemos extrair a cor do ponto da imagem onde incide o feixe *laser* e construir uma nuvem de pontos com a informação real da cor dos obstáculos.

Na Figura 3.10 podemos ver um exemplo de uma nuvem de pontos com informação. Um conjunto de objectos de dimensões e formas diferentes foi mapeado com o LRF, e à

informação dimensional foi adicionada a informação de cor recolhida com a câmara.

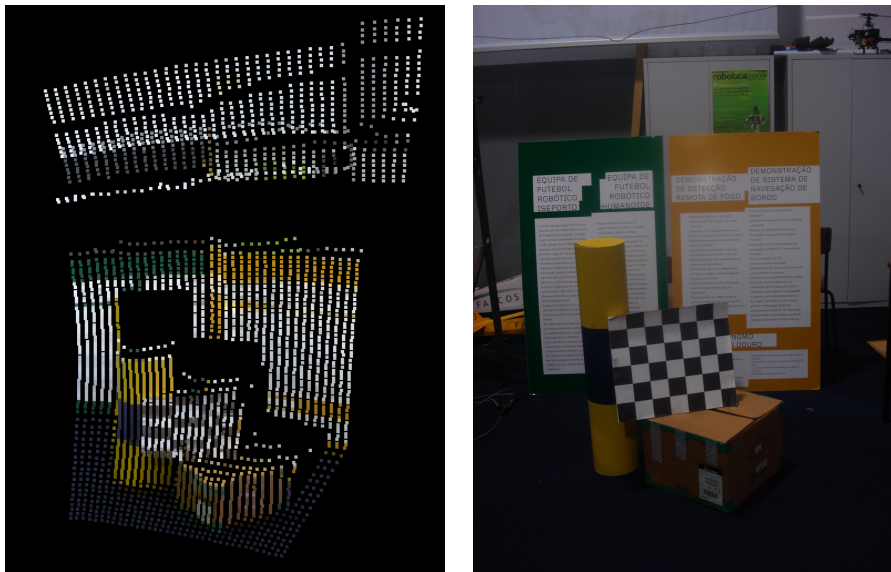


Figura 3.10: Nuvem de pontos de um conjunto de objectos e fotografia dos mesmos.

Capítulo 4

Segmentação

Conteúdo

4.1	Análise quantitativa	32
4.1.1	Determinação de <i>Breakpoints</i>	32
4.1.2	Determinação de Pontos de Inflexão	34
4.1.3	Detecção de Círculos	36
4.1.4	A Terceira Dimensão	36
4.2	Análise Qualitativa	38
4.2.1	Técnica de Segmentação <i>Brenneke</i>	39
4.2.2	Técnica de Segmentação <i>Brenneke</i> Extendida	40

A segmentação é normalmente a primeira etapa do processamento de dados 3D, e define-se pela divisão da cena tridimensional em grupos de pontos com proximidade geométrica ou não, mas que partilham um conjunto de características em comum. Isto permite classificar a nuvem de pontos garantindo assim uma melhor e mais rápida interpretação e identificação de estruturas e objectos.

A segmentação que implementamos assenta em critérios quantitativos e qualitativos, e está dividida em duas fases distintas. Numa primeira etapa, é feita uma distinção entre os pontos que definem os diferentes obstáculos com base nas descontinuidades e na monotonia da sequência de dados provenientes de cada varrimento individual. Numa segunda fase é feita uma segmentação qualitativa dos diferentes conjuntos. Nesta análise qualitativa procura-se distinguir entre os pontos que definem o chão, obstáculos verticais e pontos

suspensos do ambiente de operação do veículo. Os processos que levam à implementação destas duas técnicas são descritos nos dois tópicos seguintes.

4.1 Análise quantitativa

4.1.1 Determinação de *Breakpoints*

Este método de segmentação quantitativa assenta na detecção dos pontos de ruptura ou descontinuidade que são candidatos a pontos limites de uma determinada estrutura ou objecto. Esta detecção assenta na análise da relação geométrica entre pontos consecutivos do mesmo varrimento bidimensional (pontos p_n e p_{n-1} na Figura 4.1). A técnica utilizada para a determinação destes *breakpoints* baseia-se no *Adaptative Breakpoint Detector* (ABD), descrito por Borges e Aldon [30]. Este método consiste numa abordagem muito simples, e que pode ser caracterizada pela seguinte expressão:

$$\text{Se } \|p_n - p_{n-1}\| > D_{max}, \text{ então } k_n^b := \text{Verdade}, \text{ e } k_{n-1}^b := \text{Verdade}, \quad (4.1)$$

em que $\|p_n - p_{n-1}\|$ é a distância euclidiana entre os pontos p_n e p_{n-1} , e k_n^b e k_{n-1}^b são *flags* que quando assumem o valor de verdadeiro, assinalam a existência de uma descontinuidade, como podemos comprovar na Figura 4.1. O D_{max} é o valor máximo da distância euclidiana acima do qual se assume que os dois pontos consecutivos não fazem parte da mesma estrutura.

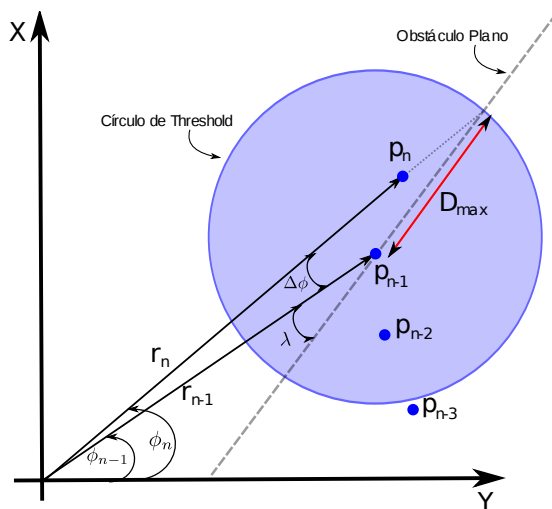


Figura 4.1: Círculo máximo de aceitação.

A principal dificuldade desta técnica é precisamente a estimação deste último parâmetro D_{max} , este depende fundamentalmente da distância relativa entre os dois pontos em análise, da distância de ambos ao sensor, da resolução angular, e do ângulo máximo de aceitação λ .

A determinação de D_{max} baseia-se num método intuitivo expresso na Figura 4.1, em que é definida uma linha imaginária que passa pelo ponto do varrimento p_{n-1} e que forma um ângulo λ em relação à direcção do varrimento ϕ_{n-1} , permitindo extrapolar o ponto mais distante aceitável para a leitura p_n . De acordo com esta restrição, uma hipotética leitura r_n^h correspondente à distância ao sensor do ponto P_n , está relacionada com a leitura r_{n-1} de acordo com a seguinte equação:

$$r_{n-1} \cdot \sin(\lambda) = r_n^h \cdot \sin(\lambda - \Delta\phi) \quad (4.2)$$

Desenvolvendo matematicamente resulta em:

$$\|p_n^h - p_{n-1}\| = r_{n-1} \cdot \frac{\sin(\Delta\phi)}{\sin(\lambda - \Delta\phi)} \quad (4.3)$$

Poderíamos utilizar $\|p_n^h - p_{n-1}\|$ como o valor limite para o teste de *breakpoint*, contudo, e dado que não é tomado em consideração o ruído associado a r_n , este teste torna-se mais falível para obstáculos mais próximos do sensor. Levando em conta o comportamento estocástico de r_n o valor de D_{max} vem como:

$$D_{max} = \|p_n^h - p_{n-1}\| + 3\sigma_r \quad (4.4)$$

O valor escolhido foi de $\lambda = 10^\circ$, o que garantiu resultados aceitáveis. Este parâmetro foi determinado experimentalmente pela análise dos resultados obtidos, comprovando o valor proposto por Borges e Aldon[30]. Quanto ao parâmetro σ este depende das características do *laser*, e que no caso particular do sensor utilizado é de $\sigma = 15 \text{ mm}$.

Este processo de segmentação permite particionar o varrimento em vários segmentos correspondentes a diferentes alvos, e eliminar os pontos que correspondem ao *time-out*, ou seja, os pontos que são retornados pelo sensor como a distância de alcance máximo, e que na realidade significam a inexistência de obstáculo.

Este método visa encontrar as extremidades da sequência de pontos de um alvo, mas não nos permite, por exemplo, caracteriza-lo, nem inferir acerca da sua monotonia.

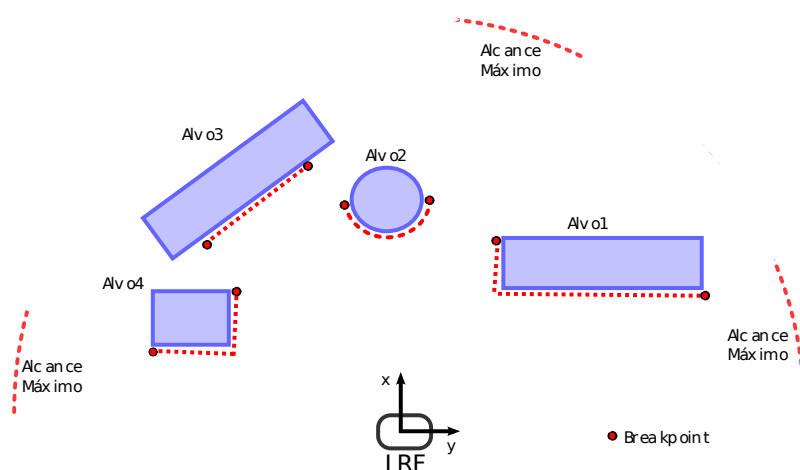


Figura 4.2: Exemplo de um Varrimento Bidimensional.

Como é visível na Figura 4.2, apenas os obstáculos cuja série de pontos se assemelha a um segmento de recta é que podem ser definidos pelos seus pontos delimitadores. Se a geometria da série de pontos se aproximar mais a uma estrutura circular, ou se tiver um ou mais pontos de inflexão, não é suficiente defini-la pelos pontos das extremidades. É necessário utilizar um processo mais complexo por forma a definir e caracteriza-lo correctamente.

4.1.2 Determinação de Pontos de Inflexão

Para encontrar os pontos de inflexão foi adoptado um método que consiste na divisão dos alvos detectados inicialmente na primeira etapa da segmentação. Este método de divisão é muito comum e normalmente utilizado em técnicas de detecção de linhas denominado *split and merge* [31], que muito genericamente se define por dividir os dados em segmentos de recta e em seguida unir aqueles que são colineares. Neste caso apenas foi utilizada a primeira etapa de divisão. A divisão da série de pontos que define o obstáculo, deve ser feita precisamente nos seus pontos de inflexão, se este existirem. Consideremos uma linha l definida pelos pontos 1 e 2 da Figura 4.3, que são os *breakpoints* definidos na primeira fase. Em seguida medimos a distância mais curta entre cada ponto intermédio da sequência e a recta l . O ponto p que apresente a maior distância D é um potencial ponto de inflexão se o valor de D for superior a um valor limite mínimo de aceitação.

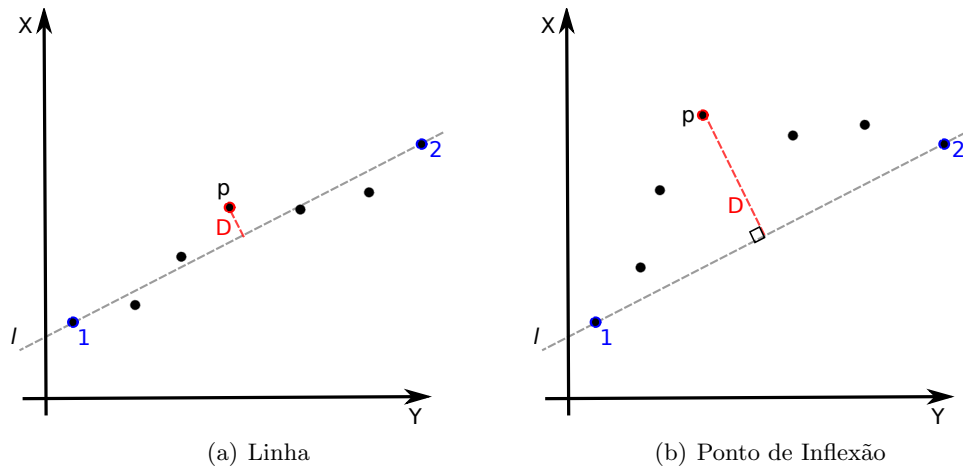


Figura 4.3: Determinação de Pontos de Inflexão.

A distância D é calculada segundo a seguinte expressão:

$$D = \frac{|a \cdot x_p + b \cdot y_p + c|}{\sqrt{a^2 + b^2}} \quad (4.5)$$

em que x_p e y_p são as coordenadas do ponto p e a , b e c são os parâmetros que definem a recta l , e que são dados por:

$$a = \frac{y_2 - y_1}{x_2 - x_1}; \quad b = -1; \quad c = y_1 - a \cdot x_1; \quad (4.6)$$

O valor de referencia para este diferencial foi determinado experimentalmente, e depende do erro estatístico do sensor *laser*, mas também do nível de precisão que se pretende obter para o modelo tridimensional. Um *threshold* muito pequeno dará origem a um modelo mais "pesado", com um numero muito grande de segmentos, por outro lado um *threshold* muito grande dará origem a um modelo mais simplificado e, possivelmente, sem o nível de detalhe desejado. O valor correcto para este parâmetro deverá depender mesmo do ambiente de operação da plataforma robótica, do seu nível de complexidade, e do facto de ser ou não estruturado.

Uma série de pontos que não apresente nenhum ponto de inflexão deverá ser interpretado como um segmento de recta caracterizado pelos dois extremos. Uma sequência de pontos que apresente um ou mais pontos de inflexão será definido por vários segmentos de recta consecutivos, mas não colineares.

4.1.3 Detecção de Círculos

A detecção de círculos em varrimentos bidimensionais utiliza uma técnica chamada *Inscribed Angle Variance* (IAV)[32]. Esta técnica faz uso das propriedades trigonométricas dos arcos de circunferência: cada ponto num arco tem ângulos congruentes (ângulos com valores iguais) em relação aos extremos. Analisemos um exemplo expresso na Figura 4.4, os pontos P_1 e P_4 são dois pontos distintos de um círculo centrado em O , enquanto que P_2 e P_3 são dois pontos do mesmo círculo presentes no arco delimitado por P_1 e P_4 , então:

$$\angle P_1P_2P_4 = \angle P_1P_3P_4 = \frac{\angle P_1OP_4}{2} \quad (4.7)$$

em que os ângulos são medidos no sentido contrário aos ponteiros do relógio. A detecção de círculos é conseguida calculando a média e desvio padrão dos ângulos inscritos. Uma detecção positiva ocorre com com um desvio padrão inferior a 8.6° e os valores médios estão compreendidos entre 90° e 135° . Estes valores foram afinados empiricamente para detectar o máximo de numero de círculos, evitando a ocorrência de falsos positivos. A confiança da análise aumenta com três factores: o número de pontos interiores do arco, um baixo desvio padrão e com a média dos ângulos a rondar os 90° .

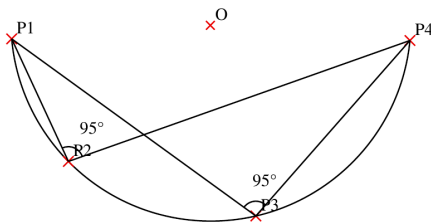


Figura 4.4: Ângulos inscritos num arco de circunferência[32].

4.1.4 A Terceira Dimensão

Até este momento todas as técnicas implementadas apenas permitiram fazer uma análise bidimensional dos dados recolhidos. A etapa seguinte consiste em mapear os segmentos de recta detectados nos vários varrimentos verticais num referencial tridimensional para que seja possível detectar superfícies na cena tridimensional. O princípio fundamental em que se baseia a detecção de superfícies é a de procurar, em varrimentos consecutivos, segmentos de recta complanares e com dimensões muito próximas. Ao inspecionarmos

uma superfície plana com sucessivos varrimentos verticais, o algoritmo de segmentação iria devolver-nos uma sucessão de linhas 2D com dimensões aproximadas à superfície em análise. A tarefa é reconhecer essa superfície na cena 3D e concatenar os segmentos de recta individuais numa única superfície. Este método foi utilizado por *Surmann* [33], e apresentava como principal vantagem um baixo custo de processamento, pois toma partido da sequência de dados, em vez de processar todos os segmentos de recta identificados. O algoritmo de detecção de superfícies pode ser dividido em quatro etapas:

1. Primeiro conjunto de linhas, proveniente do primeiro *scan* vertical, é armazenado.
2. Todas as linhas de cada *scan* vertical são comparadas com as linhas armazenadas. Se duas linhas obedecerem às condições requeridas, então essas duas linhas formam uma superfície.
3. Se não for encontrada nenhuma linha correspondente, esta nova linha poderá fazer parte de uma superfície já existente. Nesse caso compara-se a nova linha com a última linha de cada superfície armazenada, e se houver uma correspondência então a nova linha fica a fazer parte dessa superfície aumentando a sua dimensão.
4. Se por outro lado não for encontrada nenhuma correspondência então a linha é armazenada individualmente no conjunto mencionado no primeiro passo.

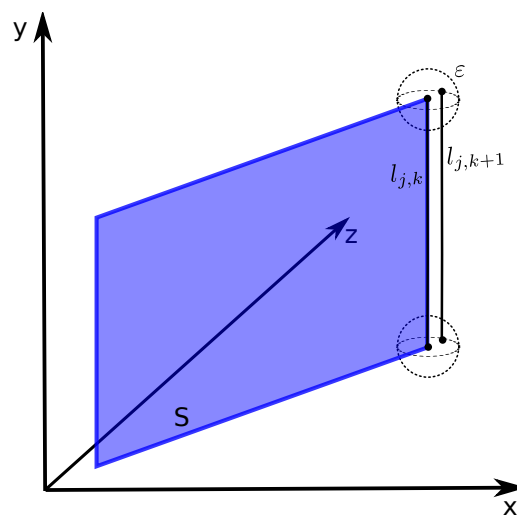


Figura 4.5: Expansão de uma superfície com uma nova linha.

Duas condições têm de ser cumpridas por forma a associarmos duas linhas: por um lado os pontos limites da linha $l_{j,k+1}$ que pretendemos associar deve estar dentro de uma

determinada área ε à volta dos pontos limites da linha de referência $l_{j,k}$. Por outro lado o ângulo formado pelas duas linhas tem de ser inferior a um dado ângulo limite δ .

4.2 Análise Qualitativa

A operação em ambientes não estruturados, tipicamente terrenos naturais com um relevo irregular, e em alguns casos sem um trilho definido, acrescenta mais dificuldade ao problema de mapeamento. A inexistência de superfícies ou estruturas de geometria regular inviabiliza a utilização das técnicas descritas no capítulo 4.1. Não existe um plano do chão perfeitamente definido, o que leva a que os obstáculos se confundam com as irregularidades do terreno. Para contornar este problema é necessário analisar de uma forma mais cuidada a nuvem de pontos, distinguindo os pontos que definem o chão, pontos que pertencem a *landmarks* (obstáculos verticais como por exemplo troncos de árvores, postes ou mesmo paredes), e pontos suspensos (como por exemplo copas de árvores). Este tipo de segmentação, baseado em detecção de obstáculos, foi apresentado por *Talukde* [34] na análise de imagens e posteriormente desenvolvido e adaptado por *Brenneke* [1][17] para a análise de varrimentos bidimensionais de um LRF.



Figura 4.6: Exemplos de cenários *outdoor*.

4.2.1 Técnica de Segmentação *Brenneke*

Na técnica de segmentação expressa por *Brenneke*[1][17] é analisado cada varrimento bidimensional individualmente. A detecção de obstáculos é a base do algoritmo utilizado. Consideremos então um varrimento vertical, adquirido de baixo para cima, de ordem i em que um dado ponto P_{ij} de ordem j desse varrimento, pode ser expresso num sistema de coordenadas cilíndricas da seguinte forma:

$$P_{ij} = (r_{ij}, \varphi_{ij}, z_{ij})^T \quad (4.8)$$

Um ponto de superfície P_{ij} é um ponto suspenso se existir pelo menos um ponto P_{ik} no mesmo varrimento vertical com um ângulo menor e com uma maior distância ao sensor:

$$r_{ik} > r_{ij} + r_t, \quad 0 \leq k < j \quad (4.9)$$

em que r_t é a distância mínima limite. Sendo que o varrimento vertical é sempre efectuado de baixo para cima, dizer-se que um ponto tem ângulo menor é sinónimo de ter um index menor no varrimento.

Um ponto de superfície P_{ij} , representado na Figura 4.7 é um *landmark* se existir pelo menos um ponto P_{ik} no mesmo varrimento vertical que esteja abaixo de P_{ij} , e P_{ij} não seja um ponto suspenso. Sendo que "abaixo" é dado por:

$$\begin{aligned} H_t &< z_{ij} - z_{ik} < H_{max}, \\ \left| \frac{r_{ij} - r_{ik}}{z_{ij} - z_{ik}} \right| &< \tan(\alpha_t), \\ 0 &\leq k < j, \\ H_{max} &= \frac{R_t}{\tan(\alpha_t)}, \end{aligned} \quad (4.10)$$

em que H_t é a altura mínima de um *landmark*, H_{max} é a altura máxima, e α_t é o ângulo máximo de desalinhamento.

Na Figura 4.7 podemos ver uma representação gráfica da segmentação efectuada segundo o algoritmo descrito anteriormente. Embora a maioria dos pontos que não sejam caracterizados como *landmarks* ou suspensos sejam na sua maioria pontos do chão, nada nos garante que todos o sejam. Esta é uma limitação que coloca alguns problemas quando a análise do terreno é um ponto essencial para inferir acerca da possibilidade de percorrer

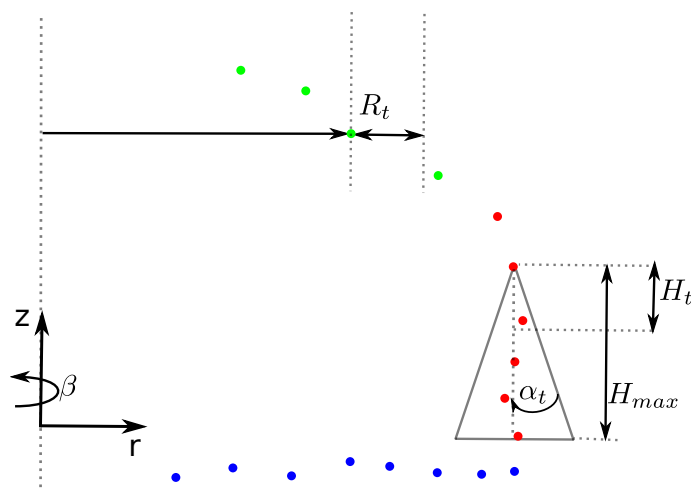


Figura 4.7: Definição de pontos de obstáculo (*landmarks* a vermelho, pontos suspensos a verde, pontos a azul).

ou não determinada área. Outra limitação observada através da análise de resultados experimentais, foi o facto de muito frequentemente não identificar *landmarks* depois de assinalado um ponto suspenso num varrimento vertical. Essas dificuldades são expressas no exemplo de um varrimento verical da Figura 4.8. Como podemos ver, a parede que ocupa uma posição mais distante da *landmark* assinalada, é classificada como um ponto de superfície de chão.

4.2.2 Técnica de Segmentação *Brenneke* Extendida

Como constatamos, a técnica de segmentação qualitativa descrita apresenta algumas limitações na sua aplicação. Para ultrapassar essas limitações desenvolvemos uma nova abordagem baseado no mesmo método de *Brenneke* que seguidamente descrevemos. Imaginemos o cenário da Figura 4.8, neste exemplo (varrimento vertical) podemos observar o chão, um alvo, uma parede interrompida por uma janela, e uma cobertura. Neste cenário são visíveis várias situações que não eram consideradas no algoritmo de detecção de obstáculos apresentado por *Brenneke* [1]. Como podemos ver, o Alvo 1 dá origem a uma zona de oclusão, fazendo com que tenhamos dois segmentos de pontos que caracterizam o chão. A parede aparece como uma segunda *landmark*, acrescentando uma nova dificuldade pelo facto de ser interrompido por uma abertura, e no topo da estrutura temos uma cobertura horizontal.

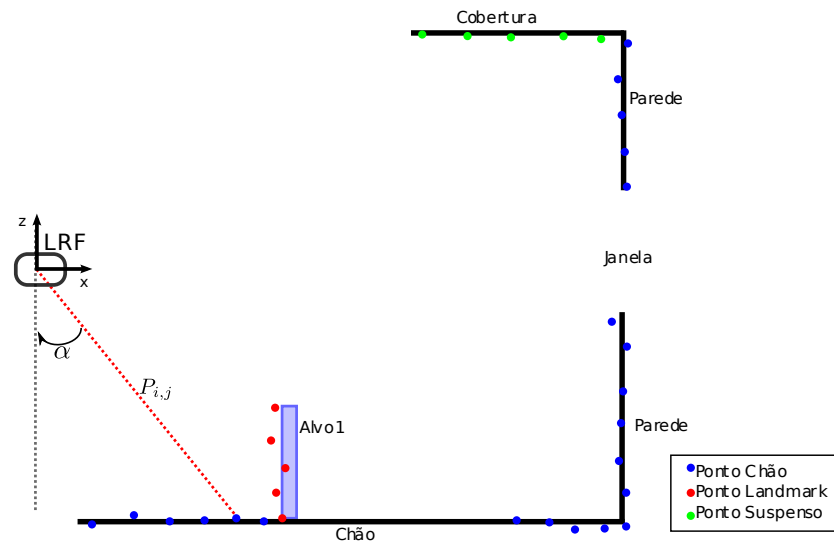


Figura 4.8: Exemplo de um varrimento vertical.

Com vista à resolução dos problemas enumerados foi desenvolvida uma nova técnica, esta caracteriza cada ponto como fazendo parte de um obstáculo, chão, ou estrutura suspensa, tendo em conta a posição do ponto P_{ik} de index k no varrimento vertical i com o ponto anterior desse mesmo varrimento P_{ij} . Para além da posição relativa dos dois pontos, também é tido em conta a caracterização que é feita dos pontos anteriores. Assim cada ponto pode fazer parte de um de três tipos diferentes: Chão, *Landmark* e Suspenso. Sendo que ainda podemos dividir Chão e Suspenso em dois subtipos cada. Se houver um alinhamento horizontal com o ponto de referencia anterior, os pontos Chão são sinalizados como Plano, e os pontos Suspenso são Tecto.

Por forma a ilustrar as ideias fundamentais que são a base desta nova forma de segmentar, podemos ver a Figura 4.9 em que as diferentes áreas coloridas, em torno do ponto P_{ij} , simbolizam as áreas onde o ponto seguinte P_{ik} pode assumir os diferentes tipos já referidos. Na área verde temos a localização de pontos Suspensos (a verde mais escuro a localização de pontos de Tecto), a azul a localização de pontos do chão (a azul mais escuro os pontos e chão plano), e a vermelho a localização de pontos de *landmark*.

Vamos agora analisar as condições necessárias para definir cada ponto segundo os três tipos já enumerados.

Para que um ponto seja considerado como fazendo parte de um *landmark*, a condição a que tem de obedecer é similar aquela que foi descrito por *Brenneke* sem a limitação de o ponto mais baixo não ser suspenso. Assim sendo, um ponto de superfície P_{ij} é um *landmark*

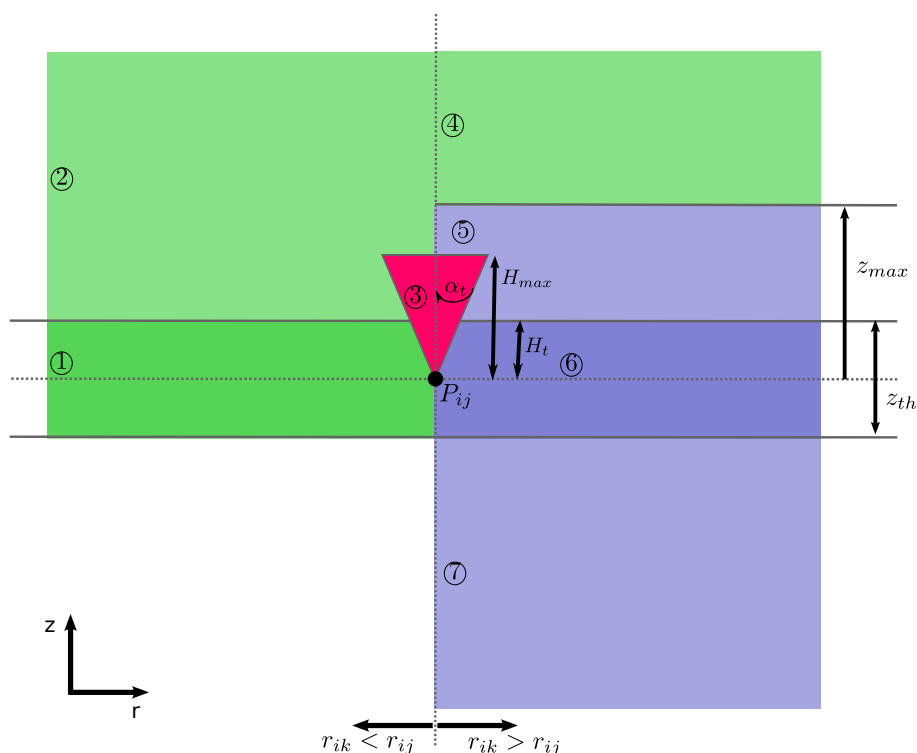


Figura 4.9: Áreas de Aceitação das diferentes classes de pontos.

se existir pelo menos um ponto P_{ik} no mesmo varrimento vertical que esteja abaixo de P_{ij} . Sendo que "abaixo" é dado pela equação 4.11 já referida. Na Figura 4.9 vemos representada a vermelho a zona em que poderemos encontrar possíveis pontos de *landmark*.

Para definir um ponto como sendo do chão, este terá de estar localizado algures na área a azul. Na zona ⑥ o ponto é de uma superfície plana, na zona ⑦ apresenta um declive negativo (um buraco ou descida), na zona ⑤ apresenta um declive positivo (lomba ou subida). Na situação em que o ponto P_{ij} estiver acima de uma altura máxima limite (a altura máxima do veículo) os pontos desta área serão considerados como suspensos. Esta análise, mais fina e cuidada dos pontos que definem o plano de deslocamento da plataforma robótica pode ser muito útil para definir a trajectória em função das dificuldades que o relevo impõe ao longo do percurso. Por vezes o caminho mais curto não é o mais fácil. A procura da maior eficiência energética do *robot*, e as suas limitações de movimentos levam a que seja necessário optar, sempre que possível, pelo percurso que apresente uma menor resistência à passagem. As áreas em questão são definidas pelos parâmetros z_{max} e z_{th} , e o seu valor foi definido pela análise de resultados experimentais.

Condição para um ponto ser classificado como chão:

$$r_{ik} > r_{ij} \cap z_{ik} < z_{max} \quad (4.11)$$

Condição para um ponto ser classificado como chão plano:

$$r_{ik} > r_{ij} \cap |z_{ij} - z_{ik}| < z_{th} \quad (4.12)$$

Os pontos definidos como suspensos poderão estar localizados em qualquer uma das áreas a verde da Figura 4.9. Na área ① temos uma superfície plana suspensa (um tecto), nas restantes áreas a verde, qualquer ponto que esteja lá localizado não tem uma "base de apoio", dado que o ponto anterior ou está muito abaixo, ou mais distante do sensor. À semelhança das áreas que englobam os hipotéticos pontos do chão, também estas áreas são definidas pelos parâmetros z_{max} e z_{th} .

Condição para um ponto ser classificado como suspenso:

$$r_{ik} > r_{ij} \cap z_{ik} > z_{max} \cup r_{ik} < r_{ij} \cap z_{ik} > -\frac{z_{th}}{2} \quad (4.13)$$

Condição para um ponto ser classificado como suspenso plano (tecto):

$$r_{ik} < r_{ij} \cap |z_{ij} - z_{ik}| < z_{th} \quad (4.14)$$

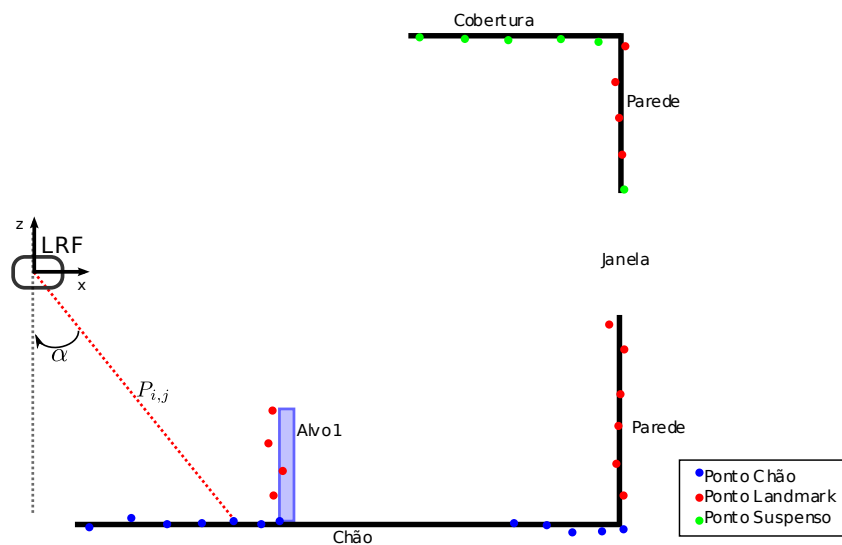


Figura 4.10: Exemplo de um varrimento vertical segmentado.

De acordo com este método, o varrimento vertical que vimos como em exemplo na Figura 4.8 seria segmentado resultando nos três conjuntos de pontos representado em pontos de cor diferente na Figura 4.10. Como podemos ver, temos três segmentos de pontos vermelhos (*landmarks*), sendo que o terceiro segmento está assente num ponto suspenso, ou seja, embora seja uma estrutura vertical que serve eventualmente de referência de localização, está posicionado a uma altura tal que eventualmente não representa um obstáculo ao deslocamento. De referir também que a oclusão causada pelo Alvo 1 não invalida que sejam definidos dois segmentos de pontos do chão.

Capítulo 5

Técnicas de Mapeamento

Conteúdo

5.1	Mapas de Elevação Digital	46
5.1.1	<i>Digital Elevation Map</i>	46
5.1.2	<i>Extended Digital Elevation Map</i>	49
5.1.3	<i>Multi Level Surface Map</i>	50
5.2	Árvores Octais	52
5.3	Mapa de Superfícies e <i>Landmarks</i>	53
5.3.1	Nível Chão	54
5.3.2	Nível <i>Landmark</i>	54
5.3.3	Nível <i>Landmark</i> Suspensa	55
5.3.4	Nível Suspenso	56

As nuvens de pontos que resultam da aquisição de dados e segmentação já referidas nos capítulos anteriores formam um mapa muito detalhado, de elevada resolução, mas difícil de inspeção e processamento. Este tipo de representação é muito exigente em termos computacionais que sem grande simplificações inviabiliza mesmo a sua utilização em aplicações de tempo real.

O elevado nível de detalhe das nuvens de pontos, funciona como um entrave à rápida pesquisa de uma determinada área, ou mesmo ao processamento de todo o conjunto de dados em tempo útil. Num mapeamento vocacionado para a navegação, localização e desvio de obstáculos a janela temporal disponível para o tratamento destes dados é muito

limitada, daí que a opção recaia por representações aproximadas, com menor nível de detalhe mas mais organizado e mais fácil de aceder.

5.1 Mapas de Elevação Digital

Uma das opções para representação simplificadas do ambiente de operação são os mapas de superfície, também referidos como mapas de elevação. Estes mapas são utilizados para representar ambientes de operação *outdoor* ou numa superfície não plana. O espaço é dividido em células com uma determinada área, que define o nível de detalhe do mapa, e em que a altura que cada célula assume, corresponde ao observado no cenário. Serão abordados três tipos de mapas de superfícies: *Digital Elevation Maps* (DEM), *Extended Digital Elevation Maps* (EDEM) e *Multi Level Surface Maps* (MLSM). Na figura 5.1 podemos observar as diferenças no detalhe de cada um deles.

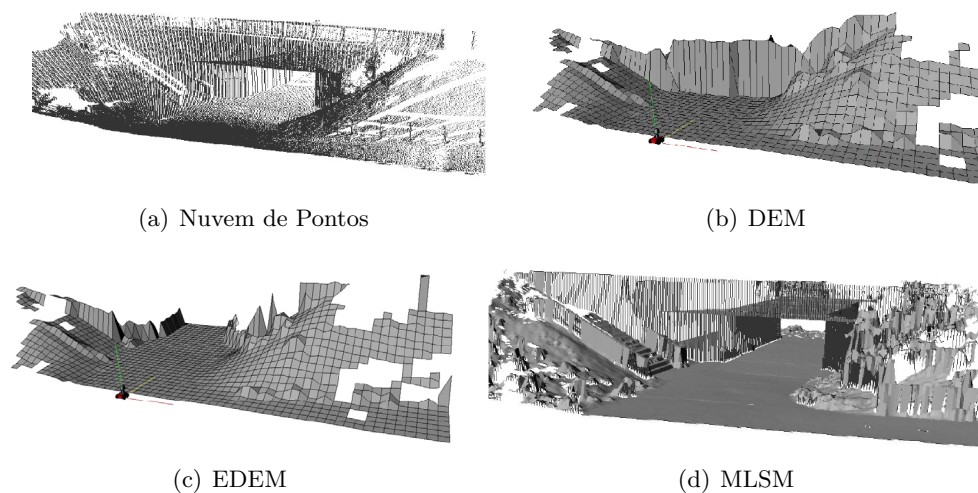


Figura 5.1: Mapas de superfícies gerados com base numa nuvem de pontos [35]

5.1.1 *Digital Elevation Map*

O mapa de elevação digital apresenta-se, face às nuvens de pontos, como uma forma bem mais simples de navegação e planeamento de trajectória numa grelha de 2.5 dimensões, e com uma representação bem mais compacta. A maior desvantagem desta representação é o facto de não precaver a existência de estruturas suspensas como pontes e viadutos ou mesmo tectos em ambientes *indoor*. Este tipo de superfícies aparecem mapeados como grandes obstáculos verticais.

Para construir um mapa de elevação digital é necessário dimensionar a grelha na qual irá ser criado o mapa. A dimensão de cada célula é a resolução do nosso mapa e denomina-se por β , e usualmente tem as dimensões de 50 *cm* ou 25 *cm* de lado, sabendo que quando menor a dimensão da célula maior o detalhe, e conseqüentemente maior será a exigência em termos computacionais. O segundo passo é associar cada ponto $p_n = \{p_{n_x}, p_{n_y}, p_{n_z}\}$ a uma determinada célula $C_{\{i,j\}}$ do mapa. Essa associação é conseguida da seguinte forma:

$$i = \left[\frac{p_{n_x}}{\beta} \right] \quad (5.1)$$

$$j = \left[\frac{p_{n_y}}{\beta} \right] \quad (5.2)$$

O ponto $p_n = \{p_{n_x}, p_{n_y}, p_{n_z}\}$ é referenciado no mapa de elevação digital como sendo $p_k, \{i, j\}$ o ponto de ordem k da célula definida por $C_{\{i,j\}}$.

Terminada a associação dos pontos às células respectivas, passamos á atribuição da altura de cada célula em função dos k pontos que englobam. Existem várias formas para a determinação do valor da célula, pela altura máxima, pela altura média de todos os pontos, pelo número máximo de pontos e por um método chamado Mapa de Elevação Robusto.

5.1.1.1 Altura Máxima

$$m_{ij} = \max_k \{p_{k_z, \{i,j\}}\} \quad (5.3)$$

A célula assume o valor de altura do ponto mais alto. Este método introduz muito ruído no mapa porque nem sempre este valor representa a altura real do obstáculo. Por exemplo, ao observar um alvo de longe introduzimos ruído que não é corrigido mesmo quando esse obstáculo é avistado mais de perto. Figura 5.2(a).

5.1.1.2 Altura Média

$$m_{ij} = \text{med}_k \{p_{k_z, \{i,j\}}\} \quad (5.4)$$

Neste método é atribuído à célula a altura média do conjunto de pontos que esta engloba. Esta não é a forma ideal de elaborar o mapa porque a distribuição dos pontos não depende

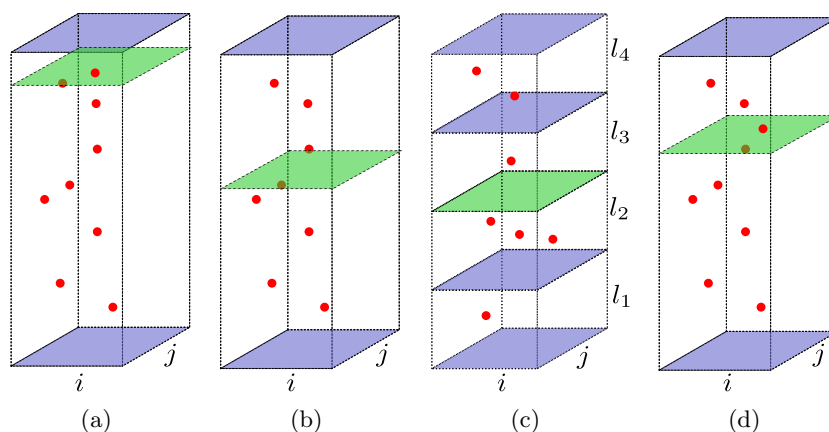


Figura 5.2: Criação de um mapa de elevação.

só do obstáculo em si, mas também do deslocamento da plataforma e do movimento de rotação do sensor *laser*. Além de que, mesmo admitindo uma distribuição uniforme do número de pontos num varrimento vertical, o ponto médio de uma superfície, como por exemplo uma parede, está longe de ser o valor mais correcto para a sua altura. Figura 5.2(b).

5.1.1.3 Maior Número de Pontos

Nesta abordagem, cada célula é dividida em subcélulas de forma análoga ao que tinha sido feita no plano horizontal, mas desta vez em altura:

$$l = \left\lceil \frac{pn_z}{\beta} \right\rceil \quad (5.5)$$

O ponto $p_k, \{i, j\}$ que tinha sido associado à célula $\{i, j\}$ é agora associado a uma subcélula l dessa célula $p_k, \{i, j, l\}$. A subcélula l que tiver o maior numero de pontos será a representativa da altura dessa subregião do mapa. Figura 5.2(c).

5.1.1.4 Mapa de Elevação Robusto

$$p_k = \text{descen sort}_k \{p_{k_z, \{i, j\}}\} \quad (5.6)$$

$$m_{ij} = p[\alpha * k] \quad (5.7)$$

A ideia subjacente a este método é escolher um ponto abaixo do qual existe uma determinada percentagem α , que é um limite mínimo imposto, do número total de pontos k . A célula assume então a altura desse ponto. Figura 5.2(d).

5.1.2 *Extended Digital Elevation Map*

O mapa de elevação estendido é uma evolução do mapa de elevação já apresentado. Neste caso os pontos referentes às estruturas suspensas são extraídos e não considerados para o cálculo da altura da célula do mapa. É então necessário identificar quais as células que contêm estruturas verticais e dessas quais as que contêm espaços livres. Para deter-

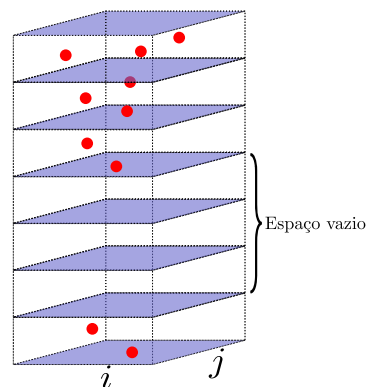


Figura 5.3: Criação de um mapa de elevação estendido.

minar a classe de uma célula, é inicialmente considerada a variância da altura de todos os pontos que preenchem a célula. Se esse valor exceder um limite máximo de aceitação, esse ponto é assinalado como não tendo sido observado de uma perspectiva superior. Em seguida é analisado se o conjunto de pontos de cada célula contém falhas que excedem a altura da plataforma robótica. Isso é conseguido pela subdivisão de cada célula da grade em patamares de altura de dimensão de 10 *cm*. Os intervalos de altura consecutivos que não contiverem nenhum ponto serão então agregados, determinando-se assim a dimensão total do espaçamento vertical.

Os mapas de elevação estendidos são efectivamente uma mais valia face aos mapas de elevação simples, pois permitem identificar zonas atravessáveis que estão por baixo de estruturas verticais. Contudo, essas estruturas verticais são eliminadas do mapa, impossibilitando a sua utilização como referências de localização e posicionamento. Além disso

se pensarmos num cenário urbano, as estruturas como pontes ou viadutos seriam eliminadas da nossa representação sempre que fossem observadas de um ponto de vista inferior, eliminando conseqüentemente eventuais percursos navegáveis.

5.1.3 *Multi Level Surface Map*

O MLSM é um mapa aproximado, que assenta na construção de uma grelha bidimensional à semelhança dos Mapas de Elevação, e que tem como objectivos fundamentais:

- Modelar a incerteza em altura.
- Representar um elevado conjunto de dados.
- Representar vários níveis de altura.
- Ser constantemente actualizado.
- Ser útil para comparação de mapas locais.
- Permitir a junção de vários mapas.

O MLSM é elaborado a partir de uma grelha 2D de tamanho variável em que cada célula $C_{\{i,j\}}$ da grelha armazena uma lista de fragmentos de superfícies FS_{ij}, \dots, FS_{ij} . Neste contexto um fragmento de superfície é definido pelo seu valor médio μ_{ij}^k e pela variância σ_{ij}^k da altura medida da posição da célula C_{ij} no mapa. Cada fragmento de superfície numa célula reflecte a possibilidade de atravessar o ambiente 3D à altura dada pela média μ_{ij}^k , com a incerteza dessa altura sendo dada por σ_{ij}^k . Para além do valor médio e da variância também é armazenado um valor de profundidade d_{ij}^k para cada fragmento. Este valor de profundidade indica que o fragmento pode estar no topo de um obstáculo vertical como um edifício, ponte ou rampa. A profundidade é definida pela diferença entre a altura do fragmento de superfície h_{ij}^k e a altura h_{ij}^k do ponto mais baixo que se considera pertencer ao objecto vertical. Para objecto planos como o chão a profundidade é zero. Na Figura 5.4 podemos ver alguns exemplos de células num MLSM.

O mapa MLSM pode ser gerado de duas maneiras diferentes: ou através de um conjunto de dados 3D, como por exemplo uma nuvem de pontos com variâncias, ou pela junção de dois outros MLSM. Ambas as formas são equivalentes, por exemplo, se um mapa m_1 criado a partir da nuvem de pontos C_1 , e se o mapa m_2 obtido a partir da nuvem C_2 forem unidos num mapa m_3 , o resultado desta junção é idêntico ao mapa gerado com base numa

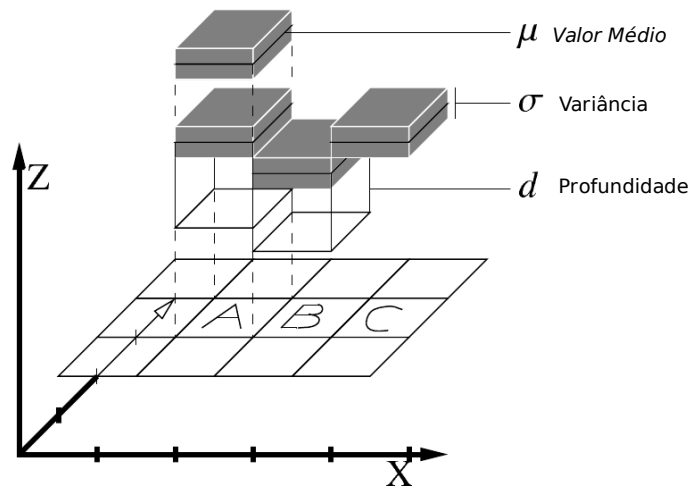


Figura 5.4: Exemplo de diferentes células num mapa MLSM. [35]

nuvem de pontos $C_3 = C_1 \cup C_2$. Para uma nuvem de pontos C com as variâncias $\sigma_1, \dots, \sigma_n$ o MLSM é criado da seguinte forma:

- Cada célula do mapa com index (i, j) recolhe todos os pontos $p = (x, y, z)$ se obedecer à condição: $s.i \leq x \leq s(i + 1)$ e $s.j \leq y \leq s(j + 1)$ em que s é o tamanho da célula do mapa.
- Em cada célula é calculado um conjunto de intervalos de altura a partir dos valores de altura dos pontos recolhidos. Sempre que dois valores de altura consecutivos estiverem mais próximos que um determinado valor limite γ , então eles pertencem ao mesmo intervalo. Isto significa que dois intervalos estão afastados entre si de pelo menos a uma distância γ . Esta distância deverá ser escolhida em função da altura do *robot*, para que seja possível navegar nesse intervalo.
- Os intervalos são classificados como estruturas do tipo horizontal ou vertical. Estas estruturas são distinguidas em função da altura do intervalo. Se ultrapassar um limite de espessura $\tau = 10 \text{ cm}$, é considerado vertical, de outra forma é considerado horizontal.
- Para cada intervalo marcado como vertical é armazenado os valores médio e de variância da leitura mais alta dentro do intervalo. A ideia por de trás disto é que para análise de atravessabilidade apenas o valor máximo é relevante. Adicionalmente

é também armazenada a profundidade d já mencionada antes. Este valor é usado quando comparando dois MLSM.

- Para cada intervalo marcado como horizontal é calculado o valor médio μ e a variância σ com base em todas as observações do intervalo. Isto é feito aplicando a etapa de actualização do filtro de *Kalman* a todas as medições. A profundidade d é colocada a zero.

Terminado o cálculo do valor médio, variância e profundidade, é eliminada a nuvem de pontos. Todos os cálculos posteriores são efectuados com base na informação do mapa gerado, reduzindo substancialmente a memória necessária para processar o MLSM em comparação com o conjunto de dados iniciais, ainda que mantendo uma representação precisa.

5.2 Árvores Octais

Uma árvore octal é uma grelha tridimensional com uma estrutura em árvore composta por um nó de raiz (*octnode*), que tem oito ramos que subdividem igualmente o volume do nó de raiz em espaço que são denominados como *octants* Figura 5.5. Cada ramo funciona também como um *octnode* que recursivamente divide o volume tanto quanto o necessário para representar a mais fina resolução necessária. O número de níveis da árvore octal determina a resolução de cada folha. A mais valia deste tipo de representação é que a árvore não tem de ser completamente instanciada, se uma grande porção da grelha estiver totalmente vazia, ocupada ou desconhecida, então podemos representá-la apenas pelo seu *octnode* truncando todos os ramos seguintes que adoptariam o mesmo valor.

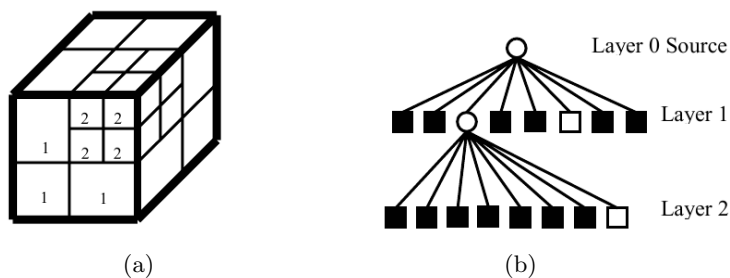


Figura 5.5: Árvore octal. Cada nível divide o volume correspondente em oito partes, mas a árvore não tem de estar completamente expandida. [36]

Tseng e *Wang* [36] apresentaram um processo automático para a extracção de superfícies planas a partir de nuvens de pontos de LIDAR. A abordagem proposta consiste em segmentar os dados segundo uma técnica de *split and merge* tridimensional baseado numa estrutura em árvore octal. No processo de divisão a árvore é dividida de forma a conter todos os pontos da nuvem, no processo de junção os ramos da árvore que contiverem pontos coplanares serão associados.

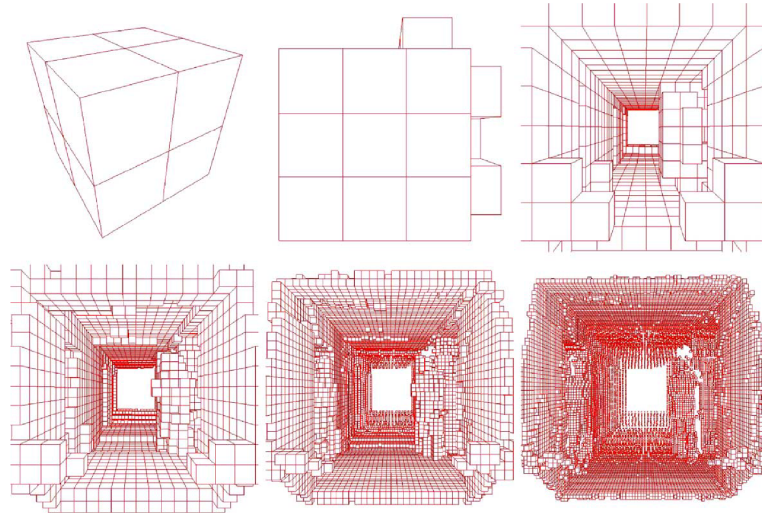


Figura 5.6: Criação de uma grelha de superfícies do nível 1 (em cima à esquerda) ao nível 8 (em baixo à direita) [37].

5.3 Mapa de Superfícies e *Landmarks*

Este é um método de representação original, implementado com o objectivo fundamental de construir um mapa aproximado do ambiente 3D tirando partido da segmentação qualitativa efectuada à nuvem de pontos. O mapa aproximado organizado em grelha, à semelhança dos mapas de superfícies já mencionados, é construído com base não só na altura dos pontos correspondentes de cada célula, mas também na sua classificação segundo as classes de pontos do Chão, pontos de *Landmark*, ou pontos Suspensos. Cada célula, contendo pontos das diferentes classes, pode contemplar vários níveis de superfícies também eles de diferentes categorias e definidos de forma diversa. Na situação em que não exista nenhum ponto associado a uma célula, essa célula é definida como uma zona de oclusão.

Nos outros casos cada célula pode conter um nível de superfície de cada uma de quatro categorias:

- Chão
- *Landmark*
- *Landmark* Suspenso
- Suspenso

A primeira etapa consiste em associar cada um dos pontos que constituem a nuvem de pontos já segmentada à respectiva célula da grelha bidimensional. Esta etapa é em tudo semelhante ao efectuado nos pontos 5.1 e 5.2, conservando também a classe à qual pertence o ponto. Desta forma o ponto $p_n = \{p_{n_x}, p_{n_y}, p_{n_z}, p_{n_{classe}}\}$ de uma nuvem de pontos segmentada é referenciado no Mapa de Superfícies e *Landmarks* como sendo $p_k, \{i, j\}$ o ponto de ordem k da célula definida por $C_{\{i,j\}}$. Depois da atribuição dos pontos correspondentes a cada célula, segue-se a etapa de determinar a existência dos diferentes níveis de superfícies e das suas propriedades.

5.3.1 Nível Chão

O nível de superfície do chão é definido no caso de existir pontos da classe Chão numa célula. Se existirem, o nível de altura desta superfície é determinado pelo cálculo da média aritmética das alturas do total de pontos desta classe. Para uma célula $C_{i,j}$ com k pontos da classe Chão, o altura do nível do chão é dado por.

$$h_{\text{chão}_{ij}} = \text{med}_k \{p_{k_z, \{i,j\}}\} \quad (5.8)$$

5.3.2 Nível *Landmark*

O nível de superfície de *Landmark* é definido pelo ponto da classe *Landmark* que apresente a maior altura se não existir, na mesma célula, nenhum pontos da classe suspenso a uma altura inferior. Esta análise mais simplista é possível devido à forma como é feita a classificação da nuvem de pontos, que nos dá a garantia que um ponto de *landmark* é parte de uma estrutura vertical. A única condição extra é a de não existir um ponto suspenso na base desta estrutura vertical. Assim sendo, podemos assumir que não é possível passar

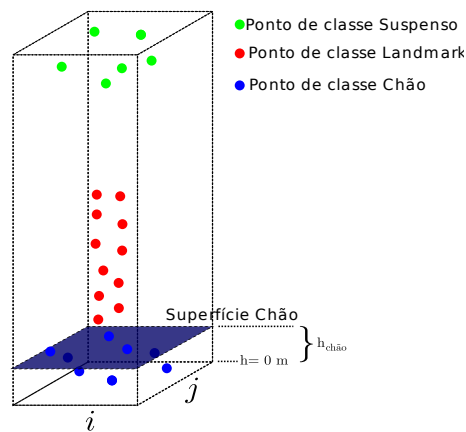


Figura 5.7: Criação do nível do tipo Chão de uma célula.

por baixo da referida estrutura, e que a sua base assenta no nível do chão que está definido para essa célula. Caso o nível do chão não esteja definido a altura inferior da *landmark* é zero. Para uma célula $C_{i,j}$ com k pontos da classe *Landmark* a altura do nível *Landmark* é dada por:

$$h_{landmark_{i,j}} = \max_k \{p_{kz, \{i,j\}}\} \quad (5.9)$$

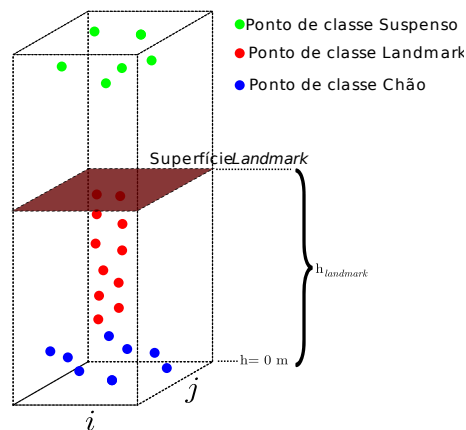


Figura 5.8: Criação do nível do tipo *Landmark* de uma célula.

5.3.3 Nível *Landmark* Suspensa

O nível de superfície de *Landmark* Suspensa é definido por dois parâmetros: altura superior, e altura inferior. Para que uma célula contenha um nível de superfície desta

classe é necessário que existam pontos classificados como *landmarke* que estejam numa posição imediatamente acima a um ou mais pontos classificados como suspensos. Por imediatamente acima, entenda-se que existe um determinado valor limite para determinar a distância entre o ponto suspenso e o ponto de *landmark* mais próximos. Assim, o nível de altura superior do *Landmark* Suspenso corresponde à altura do ponto mais alto da sequência de pontos da classe *landmark*, e o nível de altura inferior é determinado pela altura do ponto mais baixo dessa mesma sequência. Este parâmetro da altura inferior é extremamente importante na análise da atravessabilidade da região delimitada pela célula. Se este valor de altura for superior à altura da plataforma robótica, então podemos referenciar esta *landmark* para efeitos de mapeamento e localização, mas tendo a noção que esta não representa um obstáculo à navegação. Para uma célula $C_{i,j}$ com k pontos da classe *Landmark* a altura do nível *Landmark* Suspenso é dada por:

$$h_{\text{superior}_{ij}} = \max_k \{p_{kz, \{i,j\}}\} \quad (5.10)$$

$$h_{\text{inferior}_{ij}} = \min_k \{p_{kz, \{i,j\}}\} \quad (5.11)$$

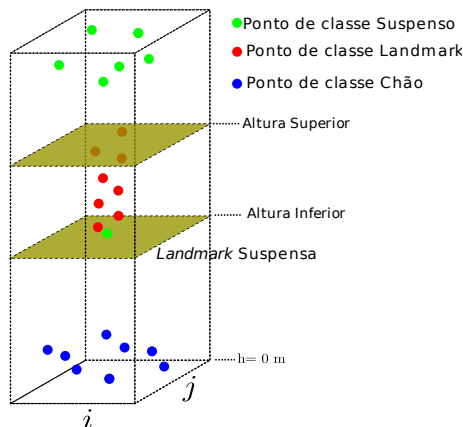


Figura 5.9: Criação do nível do tipo *Landmark* Suspenso de uma célula.

5.3.4 Nível Suspenso

O nível de superfície Suspenso é referenciado numa determinada célula se existirem pontos suspenso compreendidos nessa região. Esta propriedade define-se apenas por um valor de altura, e corresponde à altura do ponto mais baixo da classe Suspenso. A ideia

que está subjacente na escolha do ponto mais baixo é a de garantir que abaixo deste nível não existe nenhum obstáculo, logo se este nível for superior à altura do *robot* então a região desta célula é eventualmente navegável. Para uma célula $C_{i,j}$ com k pontos da classe Suspensa, a altura do nível Suspensa é dada por:

$$h_{suspensa_{ij}} = \min_k \{p_{k_z, \{i,j\}}\} \quad (5.12)$$

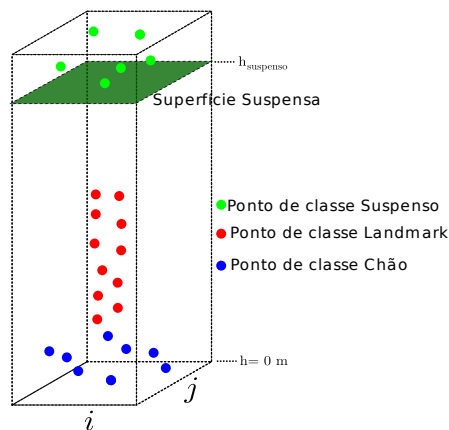


Figura 5.10: Criação do nível de tipo Suspensa de uma célula.

Na Figura 5.11 podemos ver um cenário *indoor* representado num Mapa de Superfícies e *Landmarks*. A preto temos as zonas de Oclusão, a cinzento o Chão, a vermelho *Landmarks*, a amarelo as *Landmarks* Suspensas, e a verde os níveis Suspensos.

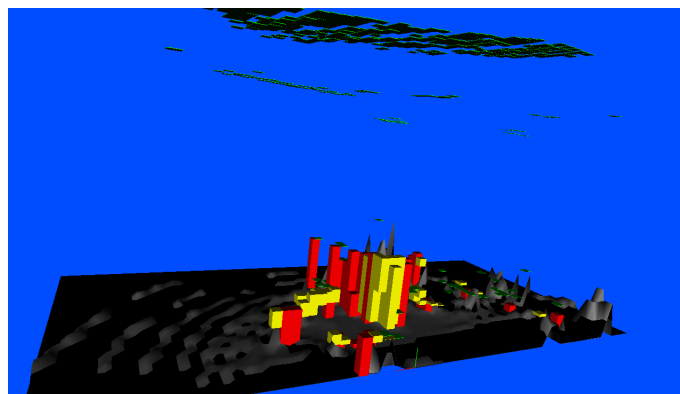


Figura 5.11: Mapa de Superfícies e *Landmarks* de um cenário *indoor*.

Esta página foi intencionalmente deixada em branco.

Capítulo 6

Resultados

Conteúdo

6.1	Nuvens de Pontos	59
6.1.1	Nuvem de Pontos Segmentada	61
6.1.2	Nuvem de Pontos com Informação de Cor	63
6.2	Técnicas de Mapeamento	64
6.2.1	Segmentos de Recta	64
6.2.2	Mapa de Superfícies e de <i>Landmarks</i>	65

Por forma a testar e validar as técnicas implementadas, foi mapeado um cenário constituído por um conjunto de obstáculos de diversas formas irregulares tentando recriar um ambiente não estruturado. Na Figura 6.1 (Cenário 1) podemos ver três fotografias que demonstram a totalidade da área inspeccionada pelo sistema de aquisição. A informação foi recolhida de uma posição estática e referenciada ao veículo. Num cenário de actuação real, seria necessário associar estes dados à informação relativa à navegação, por forma a transpor as representações para o referencial global. À nuvem de pontos resultante da inspecção do cenário, são então aplicadas as diferentes técnicas de segmentação e de simplificação implementadas, resultando num mapa classificado e mais simplificado.

6.1 Nuvens de Pontos

A etapa de aquisição e registo de dados culminou com a criação de uma nuvem de pontos com a informação volumétrica do cenário 1 da Figura 6.1. A nuvem de pontos foi adquirida de uma posição estática com uma abertura angular de 180° tanto na horizontal



Figura 6.1: Sequência de imagens ilustrativas do cenário *indoor* analisado (Cenário 1).

como na vertical, e com uma resolução de 1° nas duas componentes resultando num conjunto de 14 694 pontos. Na figura 6.2 temos uma imagem extraída de uma representação em VRML (*Virtual Reality Modeling Language*) do conjunto de pontos referidos. A re-

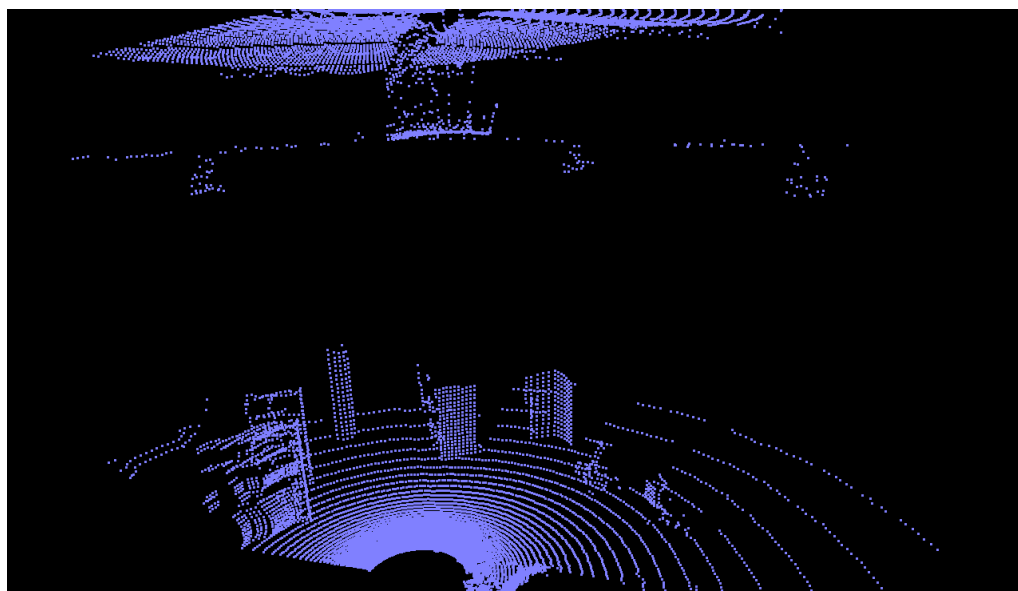


Figura 6.2: Nuvem de pontos referente ao Cenário 1.

presentação gráfica da nuvem de pontos é uma boa forma de validar e inferir, de uma forma intuitiva, sobre o sucesso das etapas de aquisição e registo de dados. Nestas duas etapas, os vários varrimentos são projectados no mesmo referencial dando origem a um mapa coerente e representativo do espaço inspeccionado.

Dado que o LRF utilizado permite efectuar no máximo 75 varrimentos "verticais" de um grau de resolução por segundo, e que são necessários 180 varrimentos verticais para obter uma nuvem de pontos com a resolução de 1 grau na "horizontal", o período que leva à obtenção à obtenção de este conjunto de dados nunca é inferior a 2.4 segundos.

6.1.1 Nuvem de Pontos Segmentada

A nuvem de pontos adquirida referente ao cenário 1 (Figura 6.1) foi processada de acordo com a técnica de segmentação *Brenneke* Extendida desenvolvida ao longo desta dissertação (capítulo 4.2.2). O resultado obtido foi uma nuvem de pontos que acrescenta à informação posicional a classificação resultante da técnica de segmentação enunciada. É então possível distinguir claramente quais os pontos que definem, as *landmarks*, a superfície do chão, e objectos ou estruturas suspensas. A totalidade da nuvem de pontos foi classificada de acordo com a distribuição expressa na Tabela 6.1.1.

Classificação	Número de Pontos
<i>Landmark</i>	1142
Chão	7439
Suspenso	449
Tecto	5664
Total	14694

Tabela 6.1: Classificação da Nuvem de Pontos

A Figura 6.3 foi retirada de um modelo VRML em que a vermelho temos os pontos classificados como *landmark*, a azul os pontos classificados como chão, a amarelo os pontos classificados como pontos suspensos e a verde os pontos suspensos dispostos num plano horizontal (superfícies de tecto).

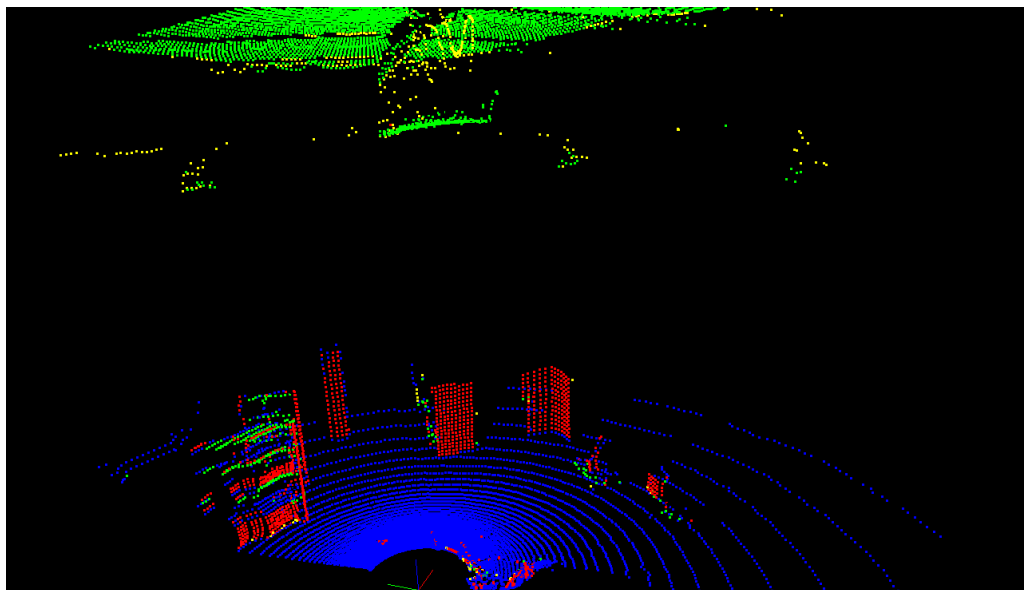


Figura 6.3: Nuvem de pontos segmentada referente ao Cenário 1.

Nas Figuras 6.4, 6.5, e 6.6 que se seguem podemos observar o Cenário 1 de três perspectivas diferentes, comparando com a região correspondente da nuvem de pontos.

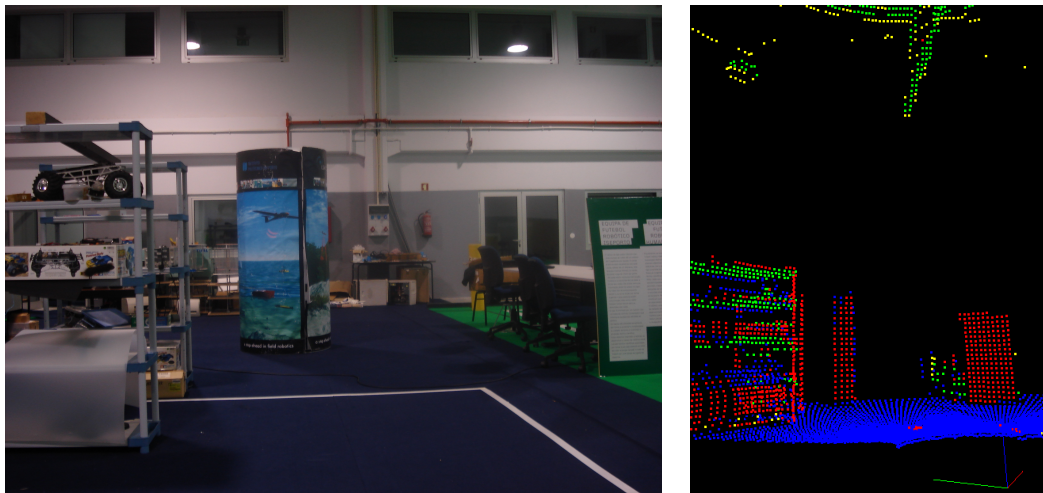


Figura 6.4: Vista parcial (lado esquerdo) do Cenário 1 e nuvem de pontos em pormenor.

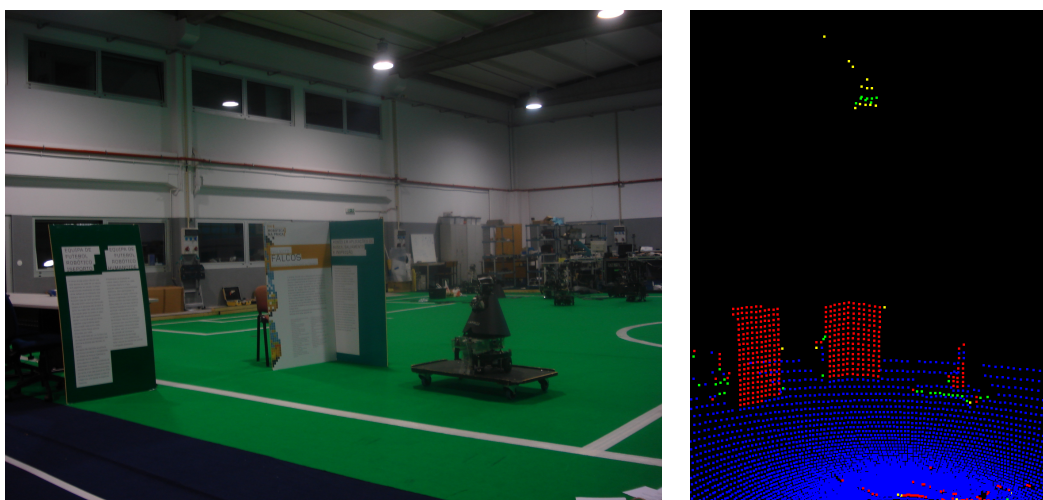


Figura 6.5: Vista parcial (zona central) do Cenário 1 e nuvem de pontos em pormenor.

A representação gráfica resultante deste método é bastante mais intuitiva, e permite efectuar uma pesquisa mais eficiente da informação recolhida. A nuvem de pontos segmentada é mais exigente em termos de armazenamento em memória, dado que, para além das coordenadas, temos agora a classificação de cada ponto quanto ao seu tipo. Contudo, este passo intermédio permite a compressão de dados com base na classificação efectuada.

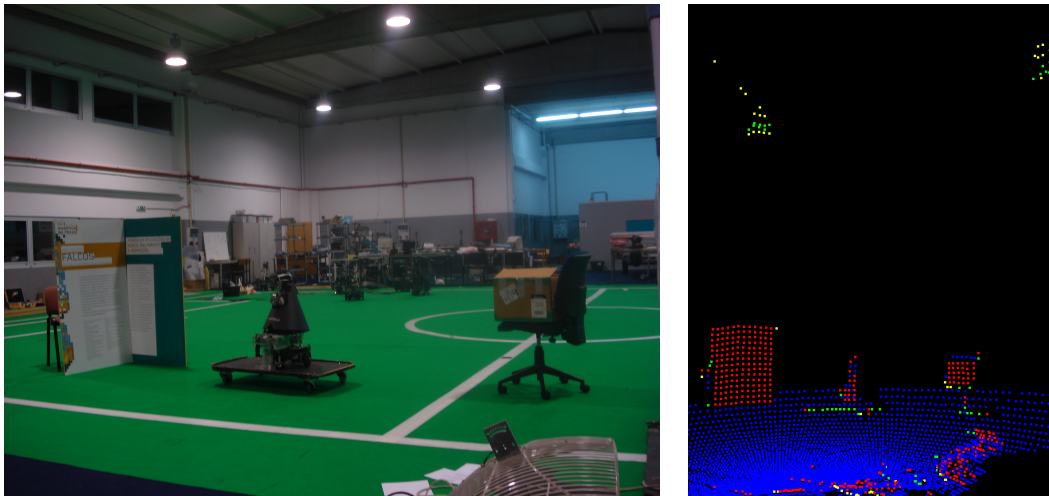


Figura 6.6: Vista parcial (lado direito) do Cenário 1 e nuvem de pontos em pormenor.

6.1.2 Nuvem de Pontos com Informação de Cor

O processo de obtenção dos parâmetros extrínsecos, relativos à calibração do *laser* e câmara, permitiu relacionar a informação volumétrica recolhida com o sistema de mapeamento com a informação de cor recolhida com uma câmara fotográfica convencional. Por forma a validar e avaliar a qualidade deste método foi inspeccionado um cenário constituído por um conjunto de objectos de formas diversas posicionados com é perceptível na Figura 6.7, ao qual chamaremos de Cenário 2.

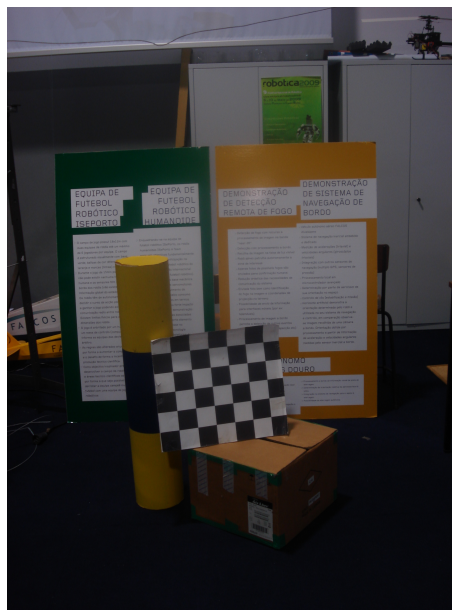


Figura 6.7: Fotografia do conjunto de obstáculos analisado (Cenário 2).

Foram realizadas cerca de 40 leituras verticais com um passo de 1° entre elas, e com uma resolução de 0.5° na vertical, resultando numa densa nuvem de pontos que abrangem o Cenário 2. Na Figura 3.2 podemos analisar essa nuvem de pontos com a informação de cor.

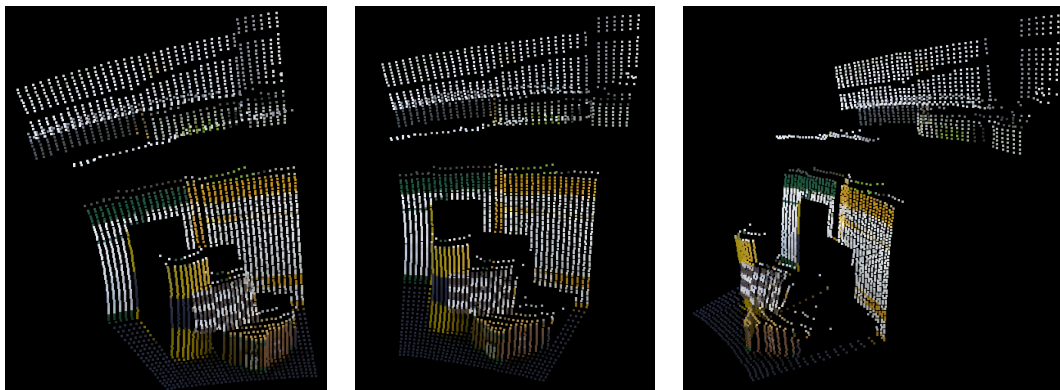


Figura 6.8: Diferentes vistas da nuvem de pontos com informação de cor resultante da análise do Cenário 2

O modelo VRML gerado por este método é mais "pesado" computacionalmente do que a nuvem de pontos simples, pelo facto de ter a informação da cor somada aos dados de posição. Apesar do custo computacional elevado, este modelo pode ser aplicado em sistemas de detecção de textura e/ou objectos de cor e forma conhecidos. A fácil análise intuitiva deste modelo, torna-o também numa boa solução para todas as aplicações em que seja necessária a intervenção humana.

A associação de informação cromática aos pontos referenciados, permite implementar técnicas de detecção de objectos e estruturas, e comprimir o volume de dados.

6.2 Técnicas de Mapeamento

6.2.1 Segmentos de Recta

Aplicando o processo de detecção de segmentos de recta à nuvem de pontos não segmentada relativa ao Cenário 1 (Figura 6.1) é possível simplificar em muito o mapa obtido. A Figura 6.9 apresenta uma vista do modelo tridimensional gerado.

O modelo que resulta deste processo de aproximação é, neste caso em particular, composto por 4 557 segmentos de recta, o que significa uma diminuição considerável do volume de dados quando comparada com a nuvem de pontos inicial. O sucesso da

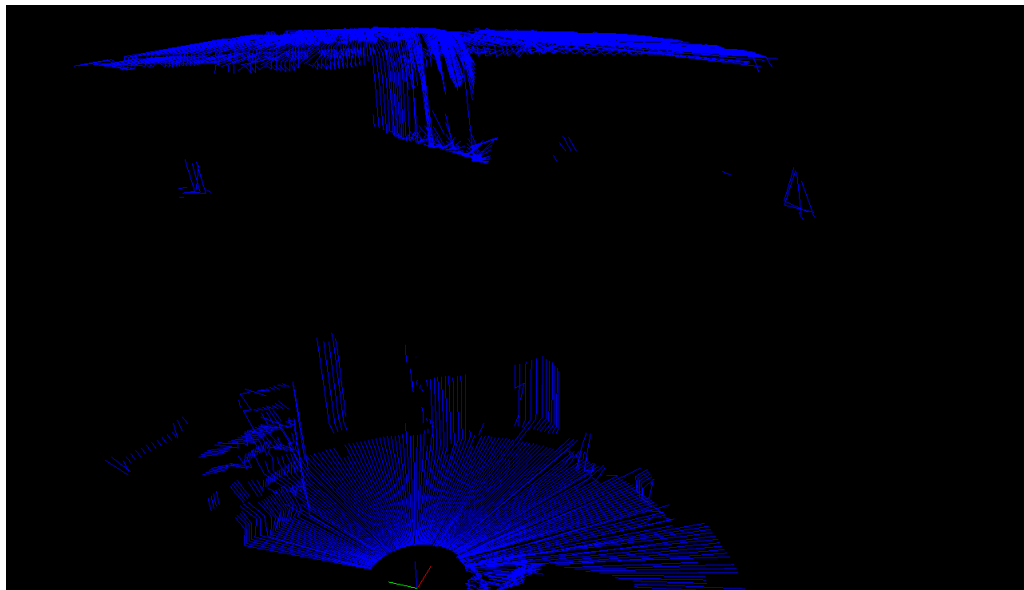


Figura 6.9: Imagem do modelo aproximado obtido por aproximação a segmentos de rectas.

”compressão” de dados depende também do próprio ambiente inspeccionado. Quanto mais regulares foram as estruturas detectadas, menos segmentos serão necessário para o definir. O resultado desta etapa seria o ponto de partida para o processo de associação de linhas coplanares para o registo de planos de superfícies.

6.2.2 Mapa de Superfícies e de *Landmarks*

Tendo como dados de entrada a nuvem de pontos segmentada, descrita e apresentada no subcapítulo 6.1.1, é implementada a técnica de aproximação de Mapa de Superfícies e de *Landmarks* desenvolvida por nós ao longo desta dissertação.

A construção do mapa tridimensional assenta numa grelha bidimensional de 16 metros de largura e de 8 metros de profundidade, com uma resolução de 50 centímetros. O número de ocorrências detectadas estão expressas na Tabela 6.2.2.

Classificação	Número Ocorrências
<i>Landmark</i>	23
<i>Landmarks</i> Suspensa	21
Superfície Suspensa	134
Superfície do Chão	169
Zonas de Oclusão	343

Tabela 6.2: Número de ocorrências das diferentes classificações.

De acordo com o mapa criado foi elaborado um modelo tridimensional em VRML, que permite representar graficamente os resultados obtidos. Na Figura 6.10 podemos ver representadas a vermelho as *Landmarks*, a amarelo as *Landmarks* Suspensas, a verde as superfícies Suspensas, a cinzento o Chão e a preto as zonas de oclusão.

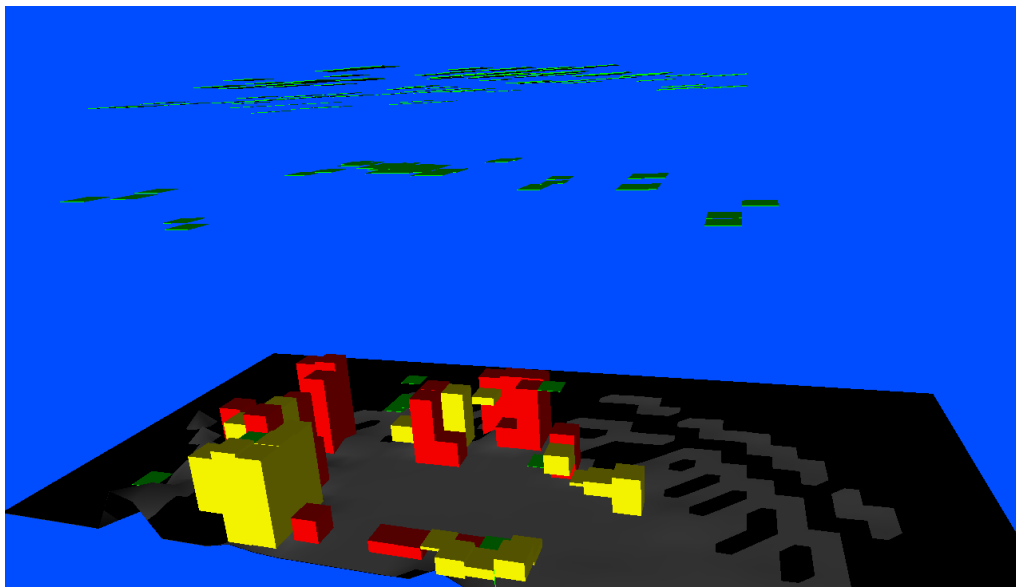


Figura 6.10: Imagem do mapa de superfícies e *landmarks* representativo do cenário 1.

Como é visível, este modelo 3D aproximado, perdeu claramente definição quando comparado com a nuvem de pontos inicial. Apesar disso, o essencial do cenário de operação continua preservado. As diferentes estruturas presentes no cenário não são agora identificáveis, mas o volume ocupado por estas é assinalado como um eventual obstáculo. No caso de estruturas verticais, como o vão de uma porta num ambiente *indoor* ou uma ponte num ambiente *outdoor*, é assinalada a sua altura inferior, permitindo traçar eventuais trajectórias no percurso livre que se encontra por baixo destas.

Capítulo 7

Conclusões e Trabalho Futuro

Conteúdo

7.1	Conclusões	67
7.2	Trabalho Futuro	68

7.1 Conclusões

Nesta dissertação foram apresentados os problemas e desafios para os quais a modelação tridimensional por parte de um LADAR se mostrou ser uma solução adequada. Também foi conseguido com sucesso avaliar diferentes sistemas de mapeamento, e inferir o quão eficientes estes são para o problema de mapeamento em tempo real por uma plataforma robótica móvel.

Foi desenvolvido um sistema capaz de adquirir informação volumétrica e cromática do espaço envolvente, capaz de ser utilizada na detecção de obstáculos e na navegação de veículos autónomos. O conjunto de dados adquiridos num ponto de observação são registados no referencial do veículo de uma forma coerente. O registo efectuado permite a rápida leitura, a inspecção e a realização das tarefas necessárias para a elaboração de mapas tridimensionais.

As técnicas de segmentação desenvolvidas e implementadas (em particular a nossa versão que resulta da extensão do algoritmo proposto por Brenneke) permitiram a identificação de objectos e estruturas facilitando a construção de mapas, que podem auxiliar as técnicas de planeamento de trajectórias, e desvio de obstáculos. Comprovamos também a capacidade de compressão de dados com base na segmentação efectuada.

Os mapas de superfícies elaborados, com diferentes técnicas, provaram ser uma boa forma de simplificação de dados. A perda de detalhe neste tipo de modelo não é um problema crítico, desde que isso não se traduza na omissão de obstáculos ou de qualquer tipo de entrave ao deslocamento. O método de mapeamento mostrou ser eficaz, pela utilização da nuvem de pontos já segmentada para a sua elaboração resultando numa representação útil para a navegação de um veículo autónomo.

7.2 Trabalho Futuro

A validação das diferentes técnicas de aquisição, registo, segmentação, e mapeamento dos dados adquiridos através de um LRF, abre portas para a integração deste sistema nas plataformas robóticas móveis dos projectos LINCE e TIGRE. Contudo será ainda necessária a caracterização da latência do sistema de mapeamento implementado, para poder inferir acerca da qualidade dos mapas gerados. A georeferênciação, e processamento em tempo real são também etapas importantes a cumprir para a integração na plataforma.

A distinção clara entre zonas atravessáveis, e os diferentes obstáculos (definidos pela sua posição e volume) torna esta metodologia ideal para fornecer informação às técnicas de desvio de obstáculos, e planeamento de trajectórias. Contudo, a resolução fixa da grelha bidimensional dos mapas de elevação pode ser uma limitação significativa. O desenvolvimento de novos métodos que permitam uma resolução variável, que evolua proporcionalmente em função da proximidade com a plataforma, ou mesmo o abandono de uma abordagem em grelha para uma mapa de representação contínua, apresentam-se como caminhos viáveis a seguir.

Desenvolver novas técnicas de mapeamento tridimensional que resultem da fusão de informação proveniente de sistemas de visão, LADAR's e sistemas inerciais. A inclusão da informação relativa à intensidade do sinal *laser* recebido à informação geométrica, apresenta-se também como uma forma de obter melhores resultados e uma melhor caracterização do mapa.

A capacidade destas plataformas em operar em cenários hostis ou inacessíveis à presença e intervenção humana, como por exemplo túneis, minas ou cenários de catástrofe, possibilita a inspecção e conseqüente elaboração de mapas essenciais para a tomada de decisões que podem valer vidas humanas.

A procura de soluções para os problemas simultâneos de localização e mapeamento em ambientes *outdoor* é uma válida aplicação para os modelos tridimensionais simplificados. A necessidade de rápida resposta, e em tempo real, das técnicas de SLAM, exige da etapa de mapeamento um resultado simplificado, e de rápida análise como são exemplo os mapas de elevação digital e mapas de superfícies.

Esta página foi intencionalmente deixada em branco.

Referências

- [1] Christian Brenneke, Oliver Wulf, and Bernardo Wagner. Using 3D Laser Range Data for SLAM in Outdoor Environments. *Intelligent Robots and Systems, IROS*, Outubro 2003.
- [2] David Ferguson Sebastian Thrun, Dirk Haehnel, Michael Montemerlo, Rudolph Triebel, Wolfram Burgard, Christopher Baker, Zachary Omohundro, Scott Thayer, and William "Red" L. Whittaker. A System for Volumetric Robotic Mapping of Abandoned Mines. In *Proceedings of the 2003 IEEE International Conference on Robotics and Automation (ICRA '03)*, volume 3, pages 4270 – 4275, May 2003.
- [3] Thrun S., Burgard W., and Fox D. A Real-Time Algorithm for Mobile Robot Mapping With Applications to Multi-Robot and 3D Mapping. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, San Francisco, CA, 2000. IEEE.
- [4] Andreas Nüchter. *3D Robotic Mapping. The Simultaneous Localization and Mapping Problem with Six Degrees of Freedom.*, volume 52 of *Springer Tracts in Advanced Robotics*. 2009.
- [5] S. Thrun. Robotic Mapping: A Survey. In G. Lakemeyer and B. Nebel, editors, *Exploring Artificial Intelligence in the New Millenium*. Morgan Kaufmann, 2002. to appear.
- [6] Carlos Almeida, Tiago Franco, Hugo Ferreira, Alfredo Martins, Ricardo Santos, José Almeida, João Carvalho, and Eduardo Silva. Collision detection developments on USV ROAZ II. *Undersea Defence Technology Conference*, 2009.
- [7] Alfredo Martins, Hugo Ferreira, Carlos Almeida, Hugo Silva, José Miguel Almeida, and Eduardo Silva. ROAZ and ROAZ II Autonomous Surface Vehicle Design and Implementation. *International Lifesaving Congress*, 2007.
- [8] Alfredo Martins, José Almeida, André Figueiredo, Filipe Santos, Domingos Bento, Hugo Silva, and Eduardo Silva. Small fixed wing autonomous aerial vehicle for forest management applications. *7th Conference on Mobile Robots and Competitions*, 2007.
- [9] Alfredo Martins, José Almeida, Carlos Almeida, André Figueiredo, Filipe Santos, Domingos Bento, Hugo Silva, and Eduardo Silva. Forest fire detection with a small fixed wing autonomous aerial vehicle. *International Autonomous Vehicles Conference*, 2007.
- [10] H. M. Silva, J. M. Almeida, L. Lima, A. Martins, E. P. Silva, and A. Patacho. Real-Time Vision System for Mobile Robotics. *Robótica Control, Automação, Instrumentação*, 2007.

- [11] Laboratório de Sistemas Autónomos (LSA) Instituto Superior de Engenharia do Porto (ISEP). Projecto LINCE, 2009. http://lsa.isep.ipp.pt/pages/lsa_lince.html.
- [12] Ivo Costa. Sistema de Controlo de um Veículo Autónomo Terrestre. Master's thesis, Instituto Superior de Engenharia do Porto (ISEP), Novembro 2009. Laboratório de Sistemas Autónomos (LSA).
- [13] RIEGL. RIEGL Laser Measurement System, 2009. <http://www.riegl.com/>.
- [14] Velodyne. High Definition Lidar, 2009. <http://www.velodyne.com/lidar/>.
- [15] Sebastian Thrun, Michael Montemerlo, Hendrik Dahlkamp, David Stavens, Andrei Aron, James Diebel, Philip Fong, John Gale, Morgan Halpenny, Gabriel Hoffmann, Kenny Lau, Celia Oakley, Mark Palatucci, Vaughan Pratt, Pascal Stang, Sven Strohband, Cedric Dupont, Lars-Erik Jendrossek, Christian Koelen, Charles Markey, Carlo Rummel, Joe van Niekerk, Eric Jensen, Philippe Alessandrini, Gary Bradski, Bob Davies, Scott Ettinger, Adrian Kaehler, Ara Nefian, and Pamela Mahoney. Stanley The Robot that Won the DARPA Grand Challenge. 23(1):661–692, June 2006.
- [16] David Ferguson, Aaron Christopher Morris, Dirk Haehnel, Christopher Baker, Zachary Omohundro, Carlos Reverte, Scott Thayer, William (Red) L. Whittaker, Chuck Whittaker, Wolfram Burgard, and Sebastian Thrun. An Autonomous Robotic System for Mapping Abandoned Mines. In *Advances in Neural Information Processing Systems (NIPS 03)*, 2003.
- [17] Oliver Wulf, Christian Brenneke, and Bernardo Wagner. Colored 2D Maps for Robot Navigation with 3D Sensor Data. *Intelligent Robots and Systems, 2004. (IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on*, 2004.
- [18] Hartmut Surmann, Andreas Nuchter, Kai Lingemann, and Joachim Hertzberg. 6D SLAM - Preliminary Report on Closing The Loop in Six Dimensions. In *Proceedings of the 5th IFAC Symposium on Intelligent Autonomous Vehicles, Lisbon, 2004*.
- [19] Kurt3D at ELROB 2006, 2009.
- [20] CMU's robotic mine mapping project.
- [21] James Underwood, Steve Scheduling, and Fabio Ramos. Real-Time Map Building with Uncertainty using Colour Camera and Scanning Laser. In *Proceedings of the 2007 Australian Conference on Robotics and Automation, 2007 Presented at 2007 Australian Conference on Robotics and Automation, Brisbane, Australia, 2007*.
- [22] James Patrick Underwood. *Reliable and Safe Autonomy for Ground Vehicles in Unstructured Environments*. PhD thesis, ARC Centre of Excellence in Autonomous Systems, The University of Sydney, 2008.
- [23] James Patrick Underwood, Andrew John Hill, and Steve Scheduling. Calibration of range sensor pose on mobile platforms. In *IROS*, pages 3866–3871. IEEE, 2007.
- [24] IBEO Lux. Ibeo Lux, 2009.
- [25] Sick. *LMS200/211/221/291 Laser Measurement Systems - Technical Description*, 2006.

- [26] O. Wulf B. Wagner. Fast 3D Scanning Methods for Laser Measurements Systems. *International Conference on Control Systems and Computer Science (CSCS)*, 1, 2003.
- [27] Martial Herbert and Eric Krotkov. 3D measurements from imaging laser radars: how good are they? *Image Vision Comput.*, 10(3):170–178, 1992.
- [28] Qilong Zhang and Robert Pless. Extrinsic Calibration of a Camera and Laser Range Finder. In *Proc. IEEE International Conference on Intelligent Robots and Systems (IROS)*, pages 2301–2306, 2004.
- [29] D. Stoyanov. Camera Calibration Toolbox for Matlab, 2009.
- [30] G.A. Borges and M.-J. Aldon. Line Extraction in 2D Range Images for Mobile Robotics. *Journal of Intelligent and Robotic Systems*, 40:267–297, 2004.
- [31] G.A. Borges and M.-J. Aldon. A split-and-merge segmentation algorithm for line extraction in 2D range images. In *Pattern Recognition, 2000. Proceedings. 15th International Conference on*, volume 1, pages 441–444 vol.1, 2000.
- [32] João Xavier, Marco Pacheco, Daniel Castro, António E. B. Ruano, and Urbano Nunes. Fast Line, Arc-Circle and Leg Detection from Laser Scan Data in a Player Driver. In *ICRA*, pages 3930–3935. IEEE, 2005.
- [33] Hartmut Surmann, Kai Lingemann, Andreas Nüchter, and Joachim Hertzberg. A 3D laser range finder for autonomous mobile robots. *32nd International Symposium on Robots*, pages 153–158, 2001.
- [34] A. Talukder, R. Manduchi, A. Rankin, and L. Matthies. Fast and Reliable Obstacle Detection and Segmentation for Cross-country Navigation. pages 610–618, 2002.
- [35] Rudolph Triebel, Patrick Pfaff, and Wolfram Burgard. Multi-Level Surface Maps for Outdoor Terrain Mapping and Loop Closing. In *IROS*, pages 2276–2282. IEEE, 2006.
- [36] Yi-Hsing Tseng and Miao Wang. Automatic plane extraction from LIDAR data based on octree splitting and merging segmentation. 5:3281– 3284, 2005.
- [37] Hartmut Surmann, Andreas Nüchter, and Joachim Hertzberg. An autonomous mobile robot with a 3D laser range finder for 3D exploration and digitalization of indoor environments. *Robotics and Autonomous Systems*, 45(3-4):181–198, 2003.
- [38] Y. Liu, R. Emery, D. Chakrabarti, W. Burgard, and S. Thrun. Using EM to Learn 3D Models with Mobile Robots. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2001.
- [39] D.M. Cole and P.M. Newman. Using laser range data for 3D SLAM in outdoor environments. In *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, pages 1556–1563, May 2006.
- [40] Maurice Muller, Hartmut Surmann, Kai Pervolz, and Stefan May. The Accuracy of 6D SLAM using the AIS 3D Laser Scanner. In *Multisensor Fusion and Integration for Intelligent Systems, 2006 IEEE International Conference on*, pages 389–394, Sept. 2006.

- [41] D. Hahnel, W. Burgard, and S. Thrun. Learning Compact 3D Models of Indoor and Outdoor Environments with a Mobile Robot. *Robotics and Autonomous Systems*, 2002. to appear.
- [42] D. Holz, C. Lorken, and H. Surmann. Continuous 3D sensing for navigation and SLAM in cluttered and dynamic environments. In *Information Fusion, 2008 11th International Conference on*, pages 1–7, 30 2008–July 3 2008.
- [43] P. Pfaff, R. Triebel, and W. Burgard. An Efficient Extension to Elevation Maps for Outdoor Terrain Mapping and Loop Closing. In *International Journal of Robotics Research (IJRR) - Special Issue of International Conference on Field and Service Robotics (FSR)*, 2005.
- [44] A. Nuchter, K. Lingemann, J. Hertzberg, and H. Surmann. Accurate object localization in 3D laser range scans. In *Advanced Robotics, 2005. ICAR '05. Proceedings., 12th International Conference on*, pages 665–672, July 2005.
- [45] Paul J. Besl and Neil D. McKay. A Method for Registration of 3-D Shapes. *IEEE Trans. Pattern Anal. Mach. Intell.*, 14(2):239–256, 1992.
- [46] M. Hentschel, O. Wulf, and B. Wagner. A GPS and laser-based localization for urban and non-urban outdoor environments. In *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, pages 149–154, Sept. 2008.
- [47] Carl Wellington, Aaron Courville, and Anthony (Tony) Stentz. A Generative Model of Terrain for Autonomous Navigation in Vegetation. *The International Journal of Robotics Research*, 25(12):1287 – 1304, December 2006.
- [48] V.T. Nguyen, A. Martinelli, N. Tomatis, and R. Siegwart. A Comparison of Line Extraction Algorithms using 2D Laser Rangefinder for Indoor Mobile Robotics. In *None*, 2005.
- [49] Andreas Nuchter, Kai Lingemann, and Joachim Hertzberg. 6D SLAM with Cached kd-tree Search. In Sándor Fekete, Rudolf Fleischer, Rolf Klein, and Alejandro Lopez-Ortiz, editors, *Robot Navigation*, number 06421 in Dagstuhl Seminar Proceedings, Dagstuhl, Germany, 2007. Internationales Begegnungs- und Forschungszentrum für Informatik (IBFI), Schloss Dagstuhl, Germany.
- [50] Andreas Nuchter, Hartmut Surmann, and Joachim Hertzberg. Automatic Classification of Objects in 3D Laser Range Scans. In *In Proc. 8th Conf. on Intelligent Autonomous Systems*.