



Pedro Miguel Pinto Moreira

# Percepção multi-sensorial baseada em grelhas para robótica móvel

Dissertação submetida em cumprimento parcial dos  
requisitos para obtenção do grau de Mestre em  
Engenharia Electrotécnica e de Computadores  
Fevereiro, 2015



UNIVERSIDADE DE COIMBRA





UNIVERSIDADE DE COIMBRA  
FACULDADE DE CIÊNCIAS E TECNOLOGIA  
DEPARTAMENTO DE ENGENHARIA ELECTROTÉCNICA E DE COMPUTADORES

# Percepção multi-sensorial baseada em grelhas para robótica móvel

Pedro Miguel Pinto Moreira

Coimbra, 2015





# Percepção multi-sensorial baseada em grelhas para robótica móvel

Orientador: Prof. Dr. Urbano José Carreira Nunes  
Co-orientador: Dr. Cristiano Premebida

Júri:

Prof. Dr. Lino José Forte Marques  
Prof. Dr. Nuno Miguel Mendonça da Silva Gonçalves  
Prof(a). Dr(a). Ana Cristina Barata Pires Lopes  
Prof. Dr. Urbano José Carreira Nunes



Pedro Miguel Pinto Moreira

Dissertação submetida para obtenção do grau de Mestre em Engenharia Electrotécnica  
e de Computadores

Departamento de Engenharia Electrotécnica e de Computadores

Faculdade de Ciências e Tecnologia, Universidade de Coimbra

Fevereiro, 2015



# Agradecimentos

Esta dissertação não poderia ser concluída sem a generosidade e o apoio de um número de pessoas e/ou instituições, para as quais eu quero aqui expressar a minha gratidão.

Em primeiro lugar, quero agradecer ao meu orientador Professor Doutor Urbano Nunes e co-orientador Doutor Cristiano Premebida pela disponibilidade e orientação durante o decorrer desta dissertação.

Ao Departamento de Engenharia Electrotécnica e de Computadores (DEEC) e ao Instituto de Sistemas e Robótica (ISR), que me disponibilizaram óptimas condições no local de trabalho, assim como todo o material necessário para a realização desta dissertação.

Aos meus colegas do laboratório, em especial ao Daniel Almeida, Jorge Perdigão, João Sousa, Fábio Faria, Luís Garrote e André Lopes, pelo apoio e ajuda prestado durante o desenvolvimento do trabalho.

Agradeço à Universidade de Coimbra (UC) pelas competências que consegui desenvolver a nível profissional e pessoal.

A todos os meus amigos que de algum modo me ajudaram a alcançar esta etapa significativa. A sua presença com palavras de apoio e motivação foi muito importante.

Estou muito grato aos meus pais por todo o apoio e ajuda que demonstraram durante todo o meu percurso académico e pelo enorme esforço que fizeram até que eu pudesse concluir o mestrado.

Esta dissertação foi realizada no âmbito dos projectos "RECI/EEI-AUT/0181/2012 - AMS-HMI12: Apoio à Mobilidade Suportada por Controlo Partilhado e Interfaces Homem-Máquina Avançados" e "Centro-07-ST24-FEDER-002028:Diagnosis and Assisted Mobility for People with Special Needs" financiados pela Fundação para a Ciência e Tecnologia (FCT), FEDER, e programas QREN e COMPETE.

# *Resumo*

O objectivo principal desta dissertação é desenvolver um sistema de percepção, usando informação de múltiplos sensores, para representação espaço-temporal do ambiente de operação em aplicações de robôs móvel ou veículos autónomos. O espaço sensorial é transformado numa representação local (i.e., centrada no sistema robótico móvel) utilizando o método das grelhas de ocupação na qual os valores da probabilidade de ocupação são actualizados. O método desenvolvido nesta dissertação tem por base os modelos de observação de três tecnologias sensoriais, nomeadamente, sensores de ultra-som, câmara RGB-D e laser. A informação proveniente de cada tecnologia sensorial é utilizada para representar o ambiente numa grelha de ocupação probabilística, ou seja, passam a existir três grelhas de ocupação locais parcelares, grelhas essas que são construídas em tempo real. Devido à complexidade e incerteza inerente ao problema da representação do ambiente (restrito ao espaço sensorial e ao movimento do robô), desenvolveu-se uma solução de percepção na qual a informação multi-sensorial é conjugada numa grelha que incorpora, numa representação comum, as três grelhas construídas parcelarmente. O sistema de percepção é crucial para diversas aplicações robóticas, nomeadamente, para a navegação de robôs móveis em ambientes complexos e/ou constituídos por obstáculos em movimento. Este sistema fornece informação essencial para o robô evitar colisões com os obstáculos, apresentando capacidade para reagir às situações envolvidas no ambiente de operação.

**Palavras-chave:** Robô móvel, Percepção, Grelhas de ocupação, Fusão, Mapeamento.

# *Abstract*

This Master's thesis aims to develop a multi-sensor based perception system for spatiotemporal representation of the operating environment in mobile robots applications or autonomous vehicles. The multimodal sensory space is transformed into a local representation (i.e., centered on the mobile robotic system) using the method of occupancy grids in which the occupation values are estimated according to a probabilistic approach. The framework developed in this thesis is based on the observation models of three sensory technologies, including ultrasound sensors, RGB-D camera and laser scanner. For each sensor, an individual model is used to represent the environment in terms of a probabilistic occupancy grid, that is, our framework depends on three partial local occupancy grids. These grids are built in real time using ROS and C++ code. Due to the complexity and uncertainty inherent to the problem of environment representation of the problem (conditioned on sensory space and robot motion), a solution developed in a such way that multi-sensory information is combined in a final grid that incorporates a common representation, considering the grids built per sensor. Such perception system is crucial for various robotic applications, in particular for navigation of mobile robots in complex environments where moving obstacles are involved, namely humans. Thus, this perception system provides essential information to the robotic system to properly react and make decisions in real-world situations, e.g. collision avoidance, safe navigation and human detection.

**Key words:** Mobile Robot, Perception, Occupancy Grids, Sensor Fusion, Mapping.

# Conteúdo

<b>Agradecimentos</b>	<b>i</b>
<b>Resumo</b>	<b>ii</b>
<b>Abstract</b>	<b>iii</b>
<b>Lista de Figuras</b>	<b>vi</b>
<b>Lista de Tabelas</b>	<b>x</b>
<b>Nomenclatura</b>	<b>xi</b>
<b>1 Introdução</b>	<b>1</b>
1.1 Motivação e Enquadramento . . . . .	1
1.2 Objectivos . . . . .	2
1.3 Implementações e contribuições fundamentais . . . . .	3
<b>2 Revisão de literatura</b>	<b>5</b>
2.1 Sensores para percepção . . . . .	5
2.1.1 Ultra-som ou sonar . . . . .	6
2.1.2 Câmara RGB-D (Kinect) . . . . .	7
2.1.3 Laser (2D) . . . . .	7
2.1.4 Laser 3D* . . . . .	9
2.2 Representação do ambiente . . . . .	9
2.2.1 Representação directa . . . . .	10
2.2.2 Representação baseada em características . . . . .	10
2.2.3 Representação baseada em grelhas . . . . .	11
2.2.3.1 Mapa de ocupação local . . . . .	11
2.2.3.2 Mapa de ocupação global . . . . .	12
2.3 Grelhas de ocupação probabilística . . . . .	12
2.3.1 Inferência Bayesiana . . . . .	13
2.3.2 Aplicação da inferência Bayesiana para grelhas de ocupação . . . . .	13
2.3.3 Fusão entre grelhas de ocupação . . . . .	14
2.3.4 <i>Bayesian Occupancy Filter</i> (BOF) . . . . .	15
2.4 Percepção local para navegação . . . . .	15
2.4.1 Navegação autónoma . . . . .	16
2.4.2 Navegação semi-autónoma . . . . .	17



2.5	Fusão multi-sensorial em mapas de ocupação . . . . .	19
<b>3</b>	<b>Grelhas de ocupação local</b>	<b>20</b>
3.1	Módulo de aquisição de dados . . . . .	20
3.2	Módulo de pré-processamento . . . . .	21
3.2.1	Projecção dos dados do Kinect para 2D . . . . .	21
3.2.2	Projecção dos dados do laser 3D* para 2D . . . . .	23
3.3	Módulo das grelhas de ocupação local . . . . .	24
3.3.1	Métodos . . . . .	25
3.3.2	Grelha usando os sonares . . . . .	27
3.3.3	Grelha usando o Kinect . . . . .	27
3.3.4	Grelha usando o laser 3D* . . . . .	27
<b>4</b>	<b>Fusão das grelhas de ocupação</b>	<b>29</b>
4.1	Módulo de fusão das grelhas de ocupação local . . . . .	29
4.2	Módulo de actualização temporal da grelha local . . . . .	33
<b>5</b>	<b>Implementações</b>	<b>38</b>
5.1	Arquitectura de <i>hardware</i> . . . . .	38
5.2	Arquitectura de <i>software</i> . . . . .	39
5.2.1	Implementação em ROS . . . . .	39
5.2.2	Implementação em Matlab . . . . .	41
<b>6</b>	<b>Resultados experimentais</b>	<b>42</b>
6.1	Análise de resultados . . . . .	42
6.2	Cenários complexos . . . . .	44
6.3	Cenário com integração temporal . . . . .	46
<b>7</b>	<b>Conclusão e Trabalho futuro</b>	<b>50</b>
7.1	Conclusão . . . . .	50
7.2	Trabalho futuro . . . . .	51
	<b>Bibliografia</b>	<b>52</b>
<b>A</b>	<b>Aquisição de dados com sistema mecatrónico</b>	<b>57</b>

# Lista de Figuras

1.1	Alguns dos sistemas robóticos móveis integrados em projectos ligados ao ISR. Estes sistemas têm sido utilizados para desenvolver e testar aplicações de robótica móvel. . . . .	2
1.2	Diagrama ilustrativo do sistema de percepção tratado neste trabalho. As aplicações enquadram-se nas áreas da robótica móvel e veículos inteligentes. . . . .	2
1.3	Estrutura do método para obtenção da grelha de ocupação local 2D. . . . .	4
2.1	(a) Aspecto físico do sonar. (b) Visualização em Rviz dos cones dos três sensores utilizados nesta dissertação. . . . .	6
2.2	(a) Câmara RGB-D Kinect. (b) Visualização em Rviz da nuvem de pontos devolvida pela câmara. . . . .	8
2.3	(a) Sensor laser 2D. (b) Visualização superior em Rviz de um varrimento laser 2D (pontos de cor branco). . . . .	8
2.4	(a) Sistema laser 2D incorporado num servo-motor. (b) Sequência de varrimentos 2D [Chou and Liu, 2013]. (c) Visualização em Rviz da nuvem de pontos devolvida por este sistema. . . . .	9
2.5	Exemplo dos diferentes métodos para representar o ambiente a partir do mesmo conjunto de dados laser 2D [Hahnel, 2005]. . . . .	10
2.6	Visualização, em Rviz, de modelos do ambiente baseados em mapas de ocupação. (a) Grelha de ocupação local 2D. (b) Grelha de ocupação global 2D. . . . .	12
3.1	Estrutura para construção das grelhas de ocupação local 2D com base num conjunto de medidas sensoriais. Cada vez que os dados de alcance dos sonares, a imagem profundidade do Kinect ou a nuvem de pontos do laser 3D* estão disponíveis é construída uma grelha de ocupação local. . . . .	20
3.2	Procedimentos para obtenção do varrimento 2D a partir da imagem de profundidade obtida do Kinect. . . . .	22
3.3	a) Exemplo do método para delinear feixes pelo algoritmo de <i>Bresenham</i> . O círculo vermelho representa o ponto medido pelo sensor (já convertido em coordenadas Cartesianas). A localização do sensor e do ponto medido está identificado pelas células a azul. As células a cinza foram afectadas pela travessia do feixe. As cruzes (x) indicam o centro geométrico da célula. b) Exemplo do procedimento de atribuição de probabilidades. As células mais escuras representam zonas conflituosas, ou seja, durante o processo de delinear feixes, tanto o feixe $k = 1$ como o $k = 2$ atribuíram uma probabilidade para essas células. . . . .	25

3.4	(a) Visualização da imagem RGB obtida a partir do Kinect, representando o cenário real. (b) Visualização no Rviz da grelha de ocupação local 2D resultante dos dados recolhidos dos três sonares, na interacção com o cenário da figura 3.4a.	28
3.5	Visualização no Rviz da grelha de ocupação local 2D resultante dos dados recolhidos do Kinect e do laser 3D*, na interacção com o cenário da figura 3.4a. . . . .	28
4.1	Estrutura para actualização espaço-temporal das grelhas de ocupação local 2D. . . . .	29
4.2	Estrutura para fusão entre as grelhas de ocupação 2D relativas aos sonares, ao Kinect e ao laser (3D). À saída obtém-se uma única grelha resultante da fusão. . . . .	30
4.3	Grelhas de ocupação local 2D construídas através do: a) Kinect; b) laser 3D*; c) método de fusão Kinect e laser 3D*. A imagem RGB da figura 3.2a ilustra o tipo de obstáculo que se encontra a <i>100 cm</i> do robô. . . . .	31
4.4	Grelhas de ocupação local 2D construídas através do: a) método de fusão Kinect e laser 3D*; b) sonares; c) método de fusão global. A imagem RGB da figura 3.2a ilustra o tipo de obstáculo a <i>100 cm</i> do robô. . . . .	32
4.5	Estrutura interna do módulo responsável pela actualização temporal da grelha de ocupação local 2D. . . . .	33
4.6	Direccionalidade da grelha entre dois instantes de tempo consecutivos. . . . .	34
4.7	Deslocamento da grelha entre dois instantes de tempo consecutivos. A grelha $G_t$ é construída com informação da grelha resultante da fusão e a grelha $G_{t-1}$ . Note-se que apenas as células que ficaram sobrepostas têm influência na actualização temporal de $G_t$ . . . . .	35
4.8	Exemplo de uma linha da grelha com deslocamento na direcção +X. . . . .	35
5.1	(a) Nomad Scout utilizado para a implementação prática do trabalho. (b) Modelo do Nomad Scout para visualização no Rviz e respectivo campo de visão vertical de cada um dos sensores. . . . .	38
5.2	(a) Imagem RGB do cenário obtida pelo Kinect. Visualização em Rviz: (b) dos cones de três sonares, da nuvem de pontos (cor branco) do Kinect e da nuvem de pontos do laser 3D*. (c) dos cones de três sonares, dos pontos 2D (cor vermelho) do laser 3D* e dos restantes pontos 2D do Kinect. (d) Grelha de ocupação local 2D construída a partir dos pontos 2D (respectivos a um varrimento 2D) obtidos pelo laser 3D*. . . . .	39
5.3	Nós e tópicos associados ao módulo responsável pela aquisição de dados. . . . .	39
5.4	Nós e tópicos associados ao módulo responsável pelo pré-processamento. . . . .	40
5.5	Nós e tópicos associados ao módulo responsável pela construção das grelhas de ocupação local 2D. . . . .	41

6.1	a) e b) são a imagem RGB da cena e correspondente grelha local 2D construída pelos sonares, respectivamente. Assim como para os pares c), d) e e), f). Esta figura retrata o problema das reflexões especulares dos sonares. . . . .	43
6.2	a) e b) são a imagem RGB da cena e correspondente grelha local 2D construída pelo Kinect, respectivamente. Assim como para os pares c), d) e e), f). O problema dos obstáculos muito próximos e das superfícies em vidro são ilustrados nesta figura. . . . .	44
6.3	a) é a imagem RGB da cena. b) e c) correspondem à grelha local 2D construída pelo laser 3D*. Em b) a grelha é visualizada em Rviz. É ilustrado o problema das superfícies em vidro nesta figura. . . . .	45
6.4	Na imagem a) observa-se o cenário, obtido pela imagem RGB do Kinect. As grelhas b), c), d) e e) são construídas pelo Kinect, pelo laser 3D*, pelos sonares e pelo processo de fusão, respectivamente. . . . .	46
6.5	Na imagem a) observa-se o cenário, obtido pela imagem RGB do Kinect. As grelhas b), c), d) e e) são construídas pelo Kinect, pelo laser 3D*, pelos sonares e pelo processo de fusão, respectivamente. . . . .	46
6.6	Na imagem a) observa-se o cenário, obtido pela imagem RGB do Kinect. As grelhas b), c), d) e e) são construídas pelo Kinect, pelo laser 3D*, pelos sonares e pelo processo de fusão, respectivamente. . . . .	47
6.7	Na imagem a) observa-se o cenário, obtido pela imagem RGB do Kinect. As grelhas b), c), d) e e) são construídas pelo Kinect, pelo laser 3D*, pelos sonares e pelo processo de fusão, respectivamente. . . . .	47
6.8	a) e b) cenário no laboratório de mecatrónica do ISR. c) sucessão de grelhas locais ao longo de um percurso (ilustrativo). O mapa global obtido por Hector SLAM é utilizado como <i>ground truth</i> . . . . .	48
6.9	Resultado do algoritmo proposto para actualização temporal da grelha de ocupação local 2D (implementado em ambiente Matlab). Em comparação com o resultado obtido na figura 6.8c, verificam-se resultados semelhantes e que validam o algoritmo proposto. . . . .	48
6.10	a) e b) cenário no laboratório de mecatrónica do ISR. c) Mapa global obtido por Hector SLAM é utilizado como <i>ground truth</i> . . . . .	49
6.11	Resultado do algoritmo proposto para actualização temporal da grelha de ocupação local 2D (implementado em ambiente Matlab). Em comparação com o resultado obtido na figura 6.10c, verificam-se resultados semelhantes e que validam o algoritmo proposto. Grelha de ocupação global obtida a partir das grelhas locais do: a) Kinect, b) laser 3D* e c) fusão. . . . .	49

A.1	Na imagem da esquerda observa-se o ambiente real utilizado para os procedimentos experimentais, assim como, o sistema mecatrónico, os sensores e o módulo de electrónica. Na figura da direita está ilustrado o sistema mecatrónico e os sensores, construído na ferramenta <i>SketchUp</i> . . . . .	57
A.2	Imagens RGB obtidas a partir do Kinect. Apresentam-se o conjunto de cenários reais utilizados para aquisição sistemática de informação sensorial. . . . .	58

# Lista de Tabelas

2.1	Características mais relevantes sobre os sensores utilizados. . . . .	5
2.2	Aplicações para sistemas de navegação recorrendo a mapas de ocupação local. . .	17
2.3	Métodos de fusão multi-sensorial usando a representação em grelhas de ocupação.	19
3.1	Características das grelhas locais 2D construídas. . . . .	24
A.1	Resultados quantitativos (média e desvio padrão) obtidos a partir da aquisição de dados do sonar. . . . .	59
A.2	Resultados quantitativos (média e desvio padrão) obtidos a partir da aquisição de dados do laser (análise para apenas um feixe). . . . .	60

# Nomenclatura

2D/3D	Bidimensional/Tridimensional
ADAS	Advanced Driver Assistance Systems
DATMO	Detection and Tracking of Moving Objects
HMI	Human Machine Interface
ISR	Instituto de Sistemas e Robótica
RGB-D	Red, Green, Blue and Depth
RobChair	Cadeira de rodas robótica (referenciada no ISR)
ROS	Robotics Operating System
RVIZ	Ruby Visualization Tool
SLAM	Simultaneous Localization And Mapping





# Capítulo 1

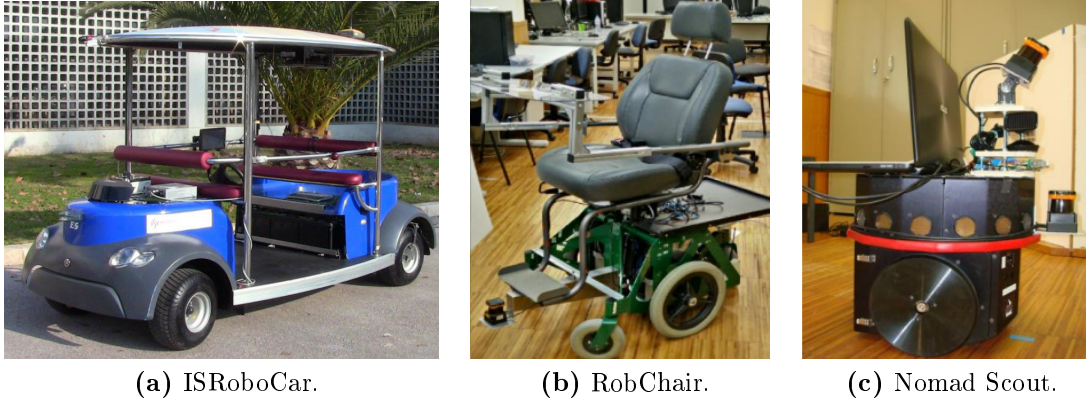
## Introdução

No decorrer deste capítulo apresenta-se uma descrição e fundamentação do problema da percepção multi-sensorial para robótica móvel e/ou veículos inteligentes. As primeiras secções descrevem as motivações, o enquadramento e os principais objectivos desta dissertação. Na última secção apresenta-se um esquema que caracteriza e ilustra, na forma de um diagrama de blocos, os principais tópicos e módulos desta dissertação, descrevendo sucintamente os capítulos existentes neste documento.

### 1.1 Motivação e Enquadramento

Nos últimos anos, a robótica tem sido um dos ramos da engenharia com grandes e importantes avanços tecnológicos; nomeadamente as áreas dos robôs móveis e veículos inteligentes (Figura 1.1) têm sido alvo de recentes evoluções em navegação autónoma, entretanto ainda assim se têm verificado algumas limitações. Problemas como comportamentos incontrolláveis, capacidade de modelar correctamente ambientes dinâmicos e prever antecipadamente possíveis perigos têm prejudicado os robôs móveis e veículos autónomos a tomarem decisões apropriadas, executarem acções e realizar tarefas com base no modelo do ambiente explorado. Na maioria dos casos, esse modelo (que caracteriza um sistema de percepção) tem grande impacto no sucesso ou insucesso das diversas aplicações onde as áreas dos robôs móveis e veículos inteligentes se enquadram, tais como navegação, desvio de obstáculos, DATMO, SLAM, planeamento de trajectórias e localização.

Apesar de muitos dos esforços na investigação, a criação de modelos que melhor representem a informação do ambiente circundante ainda é um problema em aberto e um tópico desafiador. Um exemplo de um novo foco da investigação, nomeadamente na área dos veículos inteligentes, está no desenvolvimento de novas formas de perceber o ambiente circundante para o progresso nos sistemas de assistência de condução avançada (ADAS). Esta tarefa inclui o processamento da incerteza nas medidas sensoriais numa representação genérica do ambiente de modo a que essa representação possa ser usada para diferentes funções de assistência na condução [Nguyen et al., 2012]. Em geral, as representações do ambiente seguem uma abordagem baseada nos mapas de ocupação e/ou baseada em objectos. Esta última faz suposições sobre as propriedades do objecto, por exemplo, a forma e movimento estimado. Em contraste, as representações baseadas no mapa de ocupação têm a capacidade de determinar as informações específicas sobre o espaço livre e obstáculos existentes no ambiente circundante. De um modo

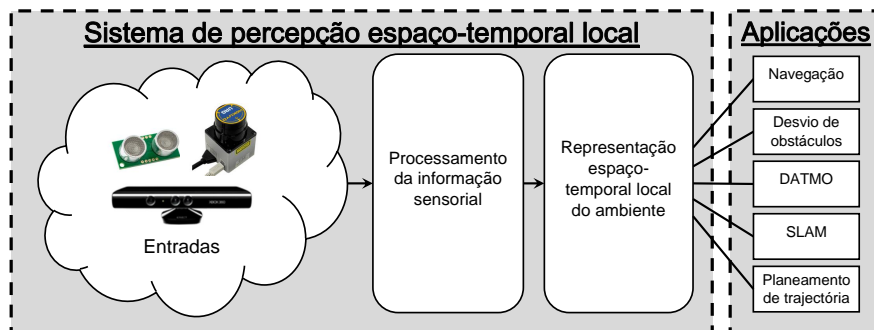


(a) ISRoboCar.

(b) RobChair.

(c) Nomad Scout.

**Figura 1.1:** Alguns dos sistemas robóticos móveis integrados em projectos ligados ao ISR. Estes sistemas têm sido utilizados para desenvolver e testar aplicações de robótica móvel.



**Figura 1.2:** Diagrama ilustrativo do sistema de percepção tratado neste trabalho. As aplicações enquadram-se nas áreas da robótica móvel e veículos inteligentes.

geral, os sistemas de percepção são confrontados com vários problemas quando o ambiente não é estático, por exemplo, situações onde os obstáculos alteraram a sua localização, pessoas que se movimentam e/ou até mesmo outros veículos móveis. Assim, são necessárias técnicas de percepção mais avançadas e robustas para superar esses problemas.

O foco desta dissertação está no desenvolvimento de um sistema de percepção local, usando informação recolhida a partir de múltiplos sensores, capaz de representar a evolução espaço-temporal do ambiente de operação. Este sistema recebe informação do ambiente através de dispositivos sensoriais adequados, tais como sensores de ultra-som, câmara RGB-D e laser. A informação recebida é processada para o espaço de grelhas locais (grelhas parcelares) que representam a informação sobre os estados de ocupação do ambiente circundante (Figura 1.2).

## 1.2 Objectivos

O trabalho realizado nesta dissertação tem por objectivo avaliar o desempenho de diferentes modalidades sensoriais tais como sensores de ultra-som, Kinect e laser, para desenvolver e implementar um sistema de percepção para sistemas robóticos autónomos.

Nesta dissertação é estudada e implementada a abordagem para representação métrica do

ambiente em grelhas de ocupação. Para cada tipo de modalidade (sensores de ultra-som, Kinect e laser) é construída uma grelha de ocupação local (i.e., centrada no sistema robótico), sendo que as grelhas resultantes são responsáveis por modelar, probabilisticamente, os estados de ocupação das regiões do ambiente que se encontram próximas ao sistema robótico. A informação das probabilidades provenientes das grelhas locais são integradas numa única representação em grelha de ocupação. A finalidade deste trabalho é utilizar a grelha resultante desse processo para actualização temporal das probabilidades de ocupação das regiões do ambiente circundante durante a navegação do sistema robótico.

## 1.3 Implementações e contribuições fundamentais

A principal contribuição consistiu no desenvolvimento de um sistema de percepção multi-sensorial, utilizando sensores de ultra-som, Kinect e um laser incorporado num servomotor (denominado neste documento por laser 3D\*), de modo a disponibilizar um modelo do ambiente circundante baseado em grelhas de ocupação local. Antes de se proceder à implementação deste sistema, foi realizado um conjunto de experiências que tinham por objectivo analisar/estudar as modalidades sensoriais apropriadas e conhecer os parâmetros que devem ser atribuídos ao modelo de incerteza do sensor. A fim de tornar este sistema funcional, procedeu-se à elaboração e agregação de diversos módulos até se atingir os objectivos propostos. A figura 1.3 sumaria as implementações e contribuições acerca do trabalho realizado no âmbito desta dissertação.

**Aquisição de dados (Capítulo 3):** Os dados de distância são fornecidos por três modalidades sensoriais com diferentes características e capacidades de percepção. Num determinado instante temporal, o sensor de ultra-som devolve uma leitura, o Kinect uma imagem de profundidade e o laser 3D\* uma nuvem de pontos. Para a aquisição dos dados utilizou-se o *software* ROS.

**Pré-processamento (Capítulo 3):** Projecção da imagem de profundidade do Kinect no espaço 2D contínuo e remoção dos pontos que contêm informação desnecessária (por exemplo, da superfície do chão). Extração dos pontos mais próximos ao centro de coordenadas do sensor, obtendo apenas o contorno dos obstáculos. O resultado final deste pré-processamento é visto como um varrimento bidimensional semelhante a um laser. No caso do laser 3D\* utilizam-se procedimentos semelhantes, mas com a nuvem de pontos.

**Grelhas de ocupação local (Capítulo 3):** Para cada uma das modalidades sensoriais é gerada uma representação do ambiente com base no método das grelhas de ocupação. Deste modo, para cada tipo de modalidade, define-se um modelo probabilístico que regista a probabilidade de ocupação relativa a cada célula da grelha. As grelhas de ocupação local (i.e. centradas no robô móvel) são construídas individualmente e em tempo real (usando ROS).

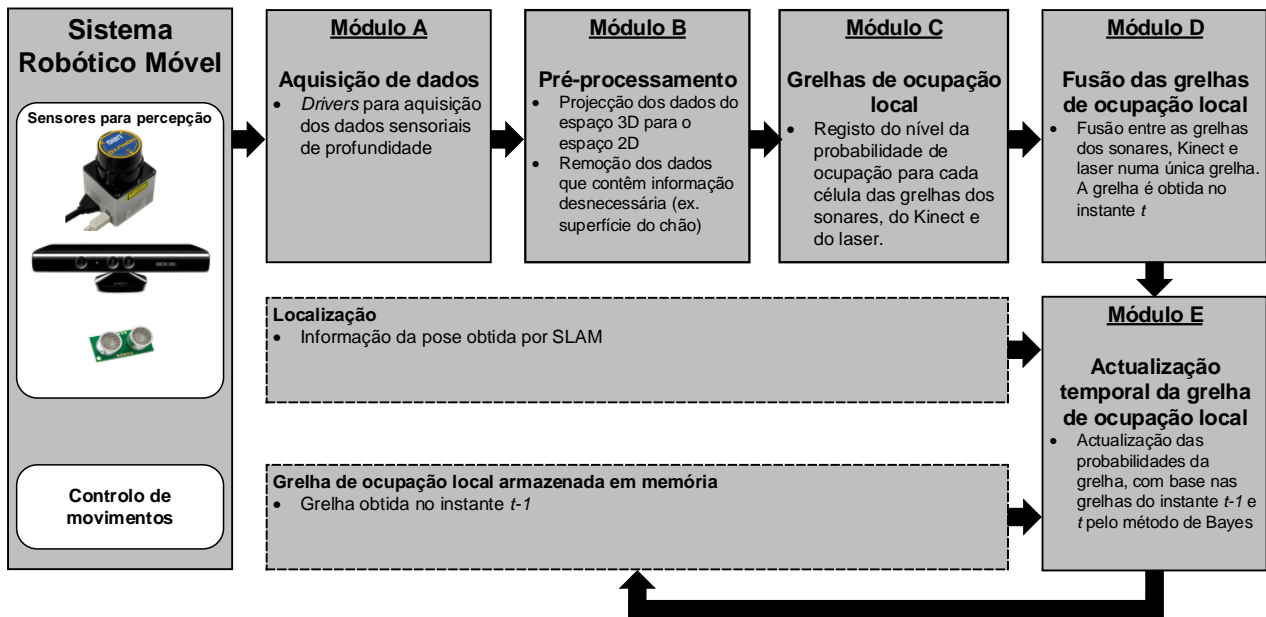


Figura 1.3: Estrutura do método para obtenção da grelha de ocupação local 2D.

**Fusão das grelhas de ocupação local (Capítulo 4):** Por forma a combinar informação a partir de um número de diferentes sensores, desenvolveu-se um método responsável por fundir a informação de ocupação das grelhas construídas individualmente. A grelha resultante da combinação das múltiplas grelhas tem uma representação mais robusta em situações onde alguns dos sensores podem falhar. Este método, testado em Matlab, tem por base a teoria Bayesiana.

**Actualização temporal da grelha local (Capítulo 4):** Com o deslocamento do robô e o surgimento de novas leituras sensoriais no tempo, torna-se necessário reforçar as probabilidades de ocupação na grelha resultante. Para implementar o método de actualização temporal recorre-se à abordagem Bayesiana (implementação realizada em Matlab). A informação do deslocamento do robô, ou seja, da estimativa da sua posição no mundo (*pose*), é fornecida por um processo que corre o algoritmo Hector SLAM [Kohlbrecher et al., 2011].

**Arquitectura de *hardware* e *software* implementado (Capítulo 5):** Para a realização das experiências, com uma aquisição sucessiva de dados, utilizou-se um sistema robótico móvel (NOMAD Scout) composto pelas três modalidades sensoriais utilizadas para percepção. O *software* foi desenvolvido em linguagem C++ no ROS e em linguagem Matlab.

**Análise dos resultados experimentais (Capítulo 6):** O comportamento dos sensores de ultra-som, da câmara RGB-D e do laser, quando expostos a ambientes reais, foi estudado e analisado de acordo com resultados experimentais, que advêm da realização de um conjunto de testes sistemáticos. De modo a avaliar as grelhas de ocupação local construídas, foi considerado um conjunto de cenários em ambientes reais.

# Capítulo 2

## Revisão de literatura

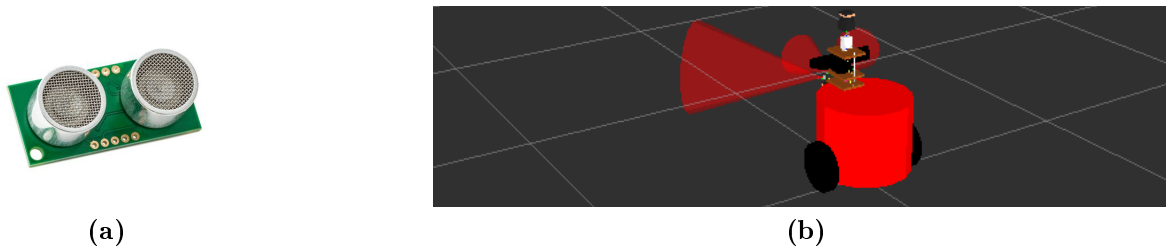
Neste capítulo introduzem-se conceitos relativos ao trabalho apresentado nesta dissertação e uma visão geral sobre o estado da arte, estando dividido em cinco secções: na primeira secção apresentam-se os sensores utilizados para perceber o ambiente; na segunda os tipos de representação do ambiente mais comuns na robótica; na terceira os métodos utilizados para processar a informação em grelhas de ocupação probabilísticas; na quarta os trabalhos descritos na literatura científica acerca da percepção local para aplicações na navegação; por último, os trabalhos desenvolvidos na literatura científica acerca dos métodos para fusão multi-sensorial em mapas de ocupação.

### 2.1 Sensores para percepção

Um requisito fundamental para a construção de um sistema de percepção multi-sensorial está na selecção dos sensores que melhor se adaptem às características ambientais com que o robô será confrontado. Para o desenvolvimento deste sistema propõe-se a utilização de três sensores de ultra-som, uma câmara RGB-D e um laser. Apesar das inúmeras diferenças que estes sensores possuem entre si, como por exemplo na quantidade de dados que adquirem, eles são estudados e analisados segundo uma metodologia comum. São sensores exteroceptivos, isto é, toda a informação é adquirida a partir do ambiente, e além disso, são também sensores activos, uma vez que emitem um sinal de excitação para o espaço exterior, aguardando depois a ocorrência de uma detecção de objectos que possam estar presentes no campo de visão. Desta forma, o robô consegue interagir com o ambiente e poderá alterar o seu comportamento perante o que vai observando.

**Tabela 2.1:** Características mais relevantes sobre os sensores utilizados.

Sensor	Sonar	Laser (2D)	Kinect
Taxa de amostragem	$\sim 23 [Hz]$	$40 [Hz]$	$30 [Hz]$
Campo de visão	$\sim 40^\circ$	$270^\circ$	$57^\circ$ horiz / $49^\circ$ vert.
Resolução angular	—	$0.25^\circ$	$0.12^\circ$
Alcance	De $3 [cm]$ a $300 [cm]$	De $10 [cm]$ a $3000 [cm]$	De $60 [cm]$ a $600 [cm]$



**Figura 2.1:** (a) Aspecto físico do sonar. (b) Visualização em Rviz dos cones dos três sensores utilizados nesta dissertação.

### 2.1.1 Ultra-som ou sonar

Os sensores de ultra-som (amplamente reconhecidos como sonares) desde cedo começaram a ter uma vasta utilização no mundo da robótica móvel [Elfes, 1987]. A informação que eles disponibilizam, ainda que limitada, pode ser importante para monitorizar uma determinada área (em forma de cone, como se pode visualizar na figura 2.1b) muito específica do ambiente, com a capacidade de detectar ocupação.

Os sonares são meramente utilizados para medir distâncias, sem contacto, a determinados objectos. Essas medidas de distância são obtidas através do princípio de tempo de voo, que consiste em medir o tempo necessário entre a emissão de sinais ondulatórios em direcção a uma superfície até serem detectados pelo sensor.

Um dos sonares mais utilizados em robôs móveis é o módulo da Polaroid 6500 (por exemplo em [Moita and Nunes, 2001]). Nesta dissertação optou-se por trabalhar com o sensor de ultra-som da Devantech SRF04 (Figura 2.1a), pois este tem algumas vantagens relativamente ao módulo da Polaroid, uma vez que consome menos corrente, tem a capacidade de detectar obstáculos a distâncias mais curtas, tem maior utilidade, um preço acessível e grande versatilidade. Como prova da sua versatilidade, existem vários projectos que integram este tipo de dispositivo sensorial, tais como numa bengala de deficientes visuais, onde o objectivo é essencialmente auxiliar o deficiente visual na detecção de obstáculos que aparecem na via pública. Essa informação é encaminhada para o utilizador através de estímulos vibratórios que consoante determinada frequência indicam a proximidade ao obstáculo [Singh et al., 2010].

O Polaroid 6500 contém apenas um transdutor que está responsável por emitir o sinal e receber o eco. Já o SRF04 dispõe de dois transdutores que estão fisicamente isolados, um responsável por emitir o sinal e o outro por receber o eco. As características mais relevantes do sonar utilizado estão representadas na tabela 2.1.

Devido ao seu funcionamento simples, os sonares conseguiram alcançar uma vasta utilização na robótica devido sobretudo a quatro factores: como devolvem apenas um único valor de distância a cada leitura adequam-se perfeitamente às aplicações em tempo real, conseguem abranger uma grande área de espaço no ambiente, permitem medir distâncias razoáveis com boa precisão e são baratos.

Apesar de todas as suas virtudes, eles ainda possuem várias desvantagens/problemas que sucedem aquando da sua utilização, como por exemplo:

- Divergência do feixe, que fará com que o sonar se torne incapaz de reconhecer passagens estreitas.
- Localização espacial imprecisa acerca do objeto, que faz com que a distância medida não seja necessariamente a distância na direcção em que o sonar está a apontar.
- Erros nas medidas, que ocorrem, sobretudo, devido a reflexões especulares que se manifestam essencialmente em superfícies lisas. Este problema ocorre quando o feixe do sonar incide em tal superfície, que não reflecte o eco directamente para o receptor.
- Onda não reflectida, por exemplo, quando existem superfícies que absorvem a energia das ondas de ultra-som emitidas, fazendo com que não gere qualquer eco detectável pelo receptor.

### 2.1.2 Câmara RGB-D (Kinect)

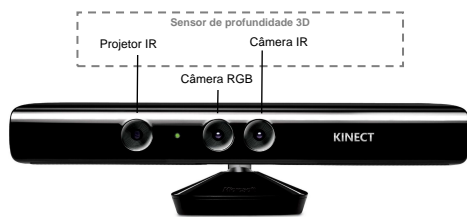
O Kinect é uma tecnologia recente, inicialmente concebida para jogos de consola, que se tem vindo a afirmar como uma escolha viável para diversas aplicações de robôs móveis que operam em ambientes *indoor* [Mojtahedzadeh, 2011]. Este sistema de visão 3D, lançado pela Microsoft em Novembro de 2010, interpreta a informação de um determinado cenário a partir de uma projecção contínua de uma estrutura de luz infravermelha. Para devolver informações de profundidade, este sensor consiste num projector de laser infravermelho (IR) combinado com um sensor CMOS monocromático que captura sucessivas imagens sob condições de luz ambiente [El-laithy et al., 2012]. Contudo, se o sensor for exposto a luz com um brilho intenso (por exemplo luz solar), as leituras de profundidade são afectadas, devolvendo informações erradas. No caso de zonas com ausência de luz (por exemplo durante a noite) as leituras não demonstram ser afectadas. O dispositivo está ainda equipado com uma câmara RGB que não tem influência nas medidas de profundidade, mas que pode ser utilizada para visualização do cenário.

A Figura 2.2a mostra a aparência física deste dispositivo e na figura 2.2b observa-se a nuvem de pontos disponibilizada por este sensor. As características mais relevantes deste dispositivo encontram-se representadas na tabela 2.1.

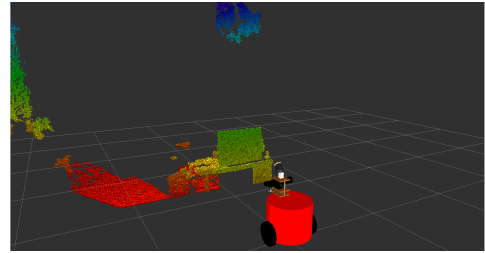
Este sensor disponibiliza uma transmissão em vídeo a uma taxa de 30 imagens por segundo. Ambas as imagens, RGB e profundidade, têm uma resolução de  $640 \times 480$  píxeis.

### 2.1.3 Laser (2D)

Os sensores de varrimento bidimensional laser (designados por laser 2D ou simplesmente laser) são dos sensores mais utilizados para percepção em robótica. No decorrer desta disser-



(a)

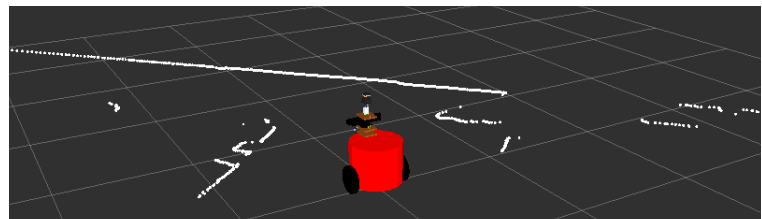


(b)

**Figura 2.2:** (a) Câmera RGB-D Kinect. (b) Visualização em Rviz da nuvem de pontos devolvida pela câmara.



(a)



(b)

**Figura 2.3:** (a) Sensor laser 2D. (b) Visualização superior em Rviz de um varrimento laser 2D (pontos de cor branco).

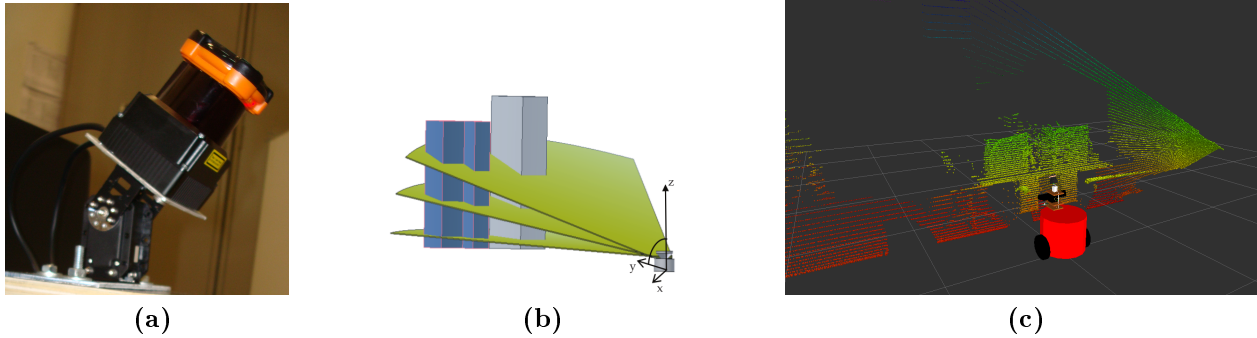
tação utilizou-se o laser Hokuyo UTM-30LX (Figura 2.3a), amplamente integrado em robótica móvel, para resolver questões desafiantes, entre as quais desvio de obstáculos ou mapeamento e localização (SLAM) [Perdigão, 2014]. Tal como os sonares, o sensor laser obtém as medidas de distância através do princípio de tempo de voo, contudo o laser proporciona melhorias significativas devido ao uso da emissão de ondas de luz e não de ultra-sons.

Este dispositivo emite uma onda electromagnética de comprimento de onda dentro do espectro infravermelho. Internamente, utiliza um mecanismo mecânico que causa a rotação de um espelho inclinado de forma a proporcionar a emissão/recepção de um conjunto de vários feixes, tornando assim possível efectuar leituras de distância aos objectos, produzindo um varrimento 2D (figura 2.3b).

Este sensor apresenta algumas vantagens, sendo elas a capacidade de fornecer leituras com boa precisão dentro da sua gama de alcances, torna perceptível a forma dos objectos detectados e monitoriza uma ampla área do ambiente. Porém, também sofre de várias desvantagens ou limitações, uma vez que é dos dispositivos sensoriais mais caros disponíveis no mercado; não detecta alguns materiais, como o vidro ou objectos muito finos (dependendo da distância ao sensor); a poeira, o fumo e/ou o nevoeiro têm interferência directa nas leituras devolvidas.

As características deste dispositivo, que são mais relevantes para este trabalho, encontram-se representadas na tabela 2.1.





**Figura 2.4:** (a) Sistema laser 2D incorporado num servo-motor. (b) Sequência de varrimentos 2D [Chou and Liu, 2013]. (c) Visualização em Rviz da nuvem de pontos devolvida por este sistema.

### 2.1.4 Laser 3D\*

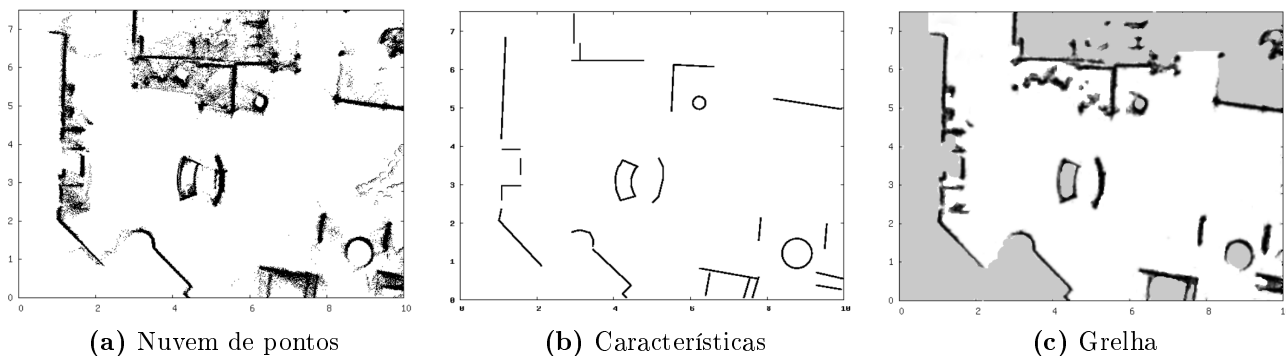
A tridimensionalidade do ambiente de operação do robô é um facto que não deve ser ignorado. A utilização de um sistema sensorial de percepção 2D, nomeadamente um laser 2D, não recolhe a informação mais relevante em cenários complexos. Neste trabalho implementou-se uma solução que concebe um laser 2D incorporado sobre um servo-motor (Figura 2.4a), que executa movimentos de rotação repetitivos entre duas posições extremas definidas, permitindo a acumulação de um conjunto de varrimentos 2D na horizontal (Figura 2.4b). Como resultado dessa acumulação é gerada uma nuvem de pontos 3D (Figura 2.4c).

Este tipo de servo-motor torna-se uma escolha apropriada devido à sua robustez, velocidade variável, simplicidade na interface de comunicação e com ampla utilização na comunidade *opensource*.

Desta forma, para além da câmara RGB-D, proporciona-se um novo sistema sensorial de percepção 3D, designado neste documento por laser 3D\*, que aumentará a complexidade na representação do ambiente com um maior número de dados a serem transmitidos. Este tipo de sistema 3D tem sido uma solução bastante utilizada em sistemas de mapeamento 3D para diversas tarefas em robótica móvel [Wulf and Wagner, 2003, Chou and Liu, 2013].

## 2.2 Representação do ambiente

A escolha para a representação do ambiente é um passo importante quando se tratam problemas de percepção. Em robótica móvel, a representação do ambiente é especificada num modelo, que pode ser descrito como uma representação das relações entre o conjunto de observações sensoriais do robô e o ambiente. Inúmeras formas de representar o ambiente têm sido propostas ao longo dos anos e entre essa enorme variedade existem dois conceitos que ainda prevalecem, sendo eles a representação em mapas métricos e topológicos [Thrun, 1998]. Normalmente, os mapas topológicos são gerados a partir de informações fornecidas pelos mapas métricos. Os métodos



**Figura 2.5:** Exemplo dos diferentes métodos para representar o ambiente a partir do mesmo conjunto de dados laser 2D [Hahnel, 2005].

mais populares para representar mapas métricos do ambiente são: abordagem directa [Lu and Milios, 1997], abordagem baseada em características [Leonard and Durrant-Whyte, 1991] e a abordagem baseada em grelhas [Elfes, 1987].

### 2.2.1 Representação directa

O método de representação directa do ambiente pode ser utilizado quando sensores de profundidade estão à disposição, nomeadamente, sensores laser. Este método utiliza as medidas devolvidas directamente pelo sensor para representar o ambiente, sem extrair quaisquer características acerca dele. Cada medida proveniente de um varrimento laser é um conjunto de pontos, onde esse mesmo conjunto indica a incidência dos feixes nos obstáculos. Este tipo de mapa pode ser construído simplesmente agregando os pontos medidos até conduzir a uma representação do ambiente semelhante a uma nuvem de pontos (ver figura 2.5a). Este método ignora as incertezas e as características do ambiente e requer muitos recursos de memória, revelando-se estes aspectos desvantagens do mesmo.

### 2.2.2 Representação baseada em características

As abordagens baseadas nas características compactam os dados extraídos a partir do ambiente num conjunto de características. As características utilizadas neste tipo de modelo podem ser primitivas geométricas simples tal como pontos, linhas e/ou círculos, ou complexas tal como super quádricas [Pascoal et al., 2014]. O grande desafio consiste na estimação dos parâmetros das primitivas que melhor se ajustam às observações dos sensores. Na figura 2.5b apresenta-se uma imagem de um mapa 2D baseado em características geométricas simples.

Em termos de conteúdo de informação espacial, estes mapas estão limitados a pontos de referência paramétricos ou aos objectos modelados. A representação geométrica não é capaz de representar ambientes bastante complexos nem de reconhecer o espaço livre entre duas características. Como nem tudo no ambiente se caracteriza com primitivas geométricas bem

definidas, este tipo de representação limita-se a aproximar as estruturas naturais dos objectos.

### 2.2.3 Representação baseada em grelhas

Alberto Elfes e Hans Moravec [Elfes, 1987, Moravec and Elfes, 1985, Moravec, 1988] introduziram o conceito da representação baseada em grelhas, provando desde então a sua eficiência na construção de um mapa utilizado para navegação de robôs móveis. Na figura 2.5c observa-se um mapa baseado em grelhas de ocupação 2D. Neste tipo de representação, o ambiente é subdividido num conjunto de células que formam a grelha, sendo que a resolução deste está dependente do tamanho das células. Além desta discretização do espaço em pequenas regiões, a cada célula está associada uma medida probabilística de ocupação. Esta medida tem em conta qualquer número real no intervalo  $[0, 1]$  e descreve um dos dois estados possíveis para a célula: ocupado ou livre. As células com probabilidade de ocupação 0 correspondem às regiões livres, as células com probabilidade 1 às regiões ocupadas e as células com probabilidade 0.5 às regiões ainda desconhecidas. A modelação do mapa em grelhas de ocupação é uma abordagem eficiente para representar a incerteza e para fundir medidas devolvidas por múltiplos sensores. O sucesso dessa solução motivou diversos autores a melhorar e adaptar o modo como representam a informação nas grelhas. Com o passar dos anos, modelar um mapa em grelhas tem-se revelado um modelo métrico muito eficaz para representar o ambiente, uma vez que não faz nenhuma suposição sobre a forma dos elementos que compõem o mesmo. Contudo, [Filliat and Meyer, 2003] esta solução exige o mesmo esforço computacional independentemente da complexidade do ambiente.

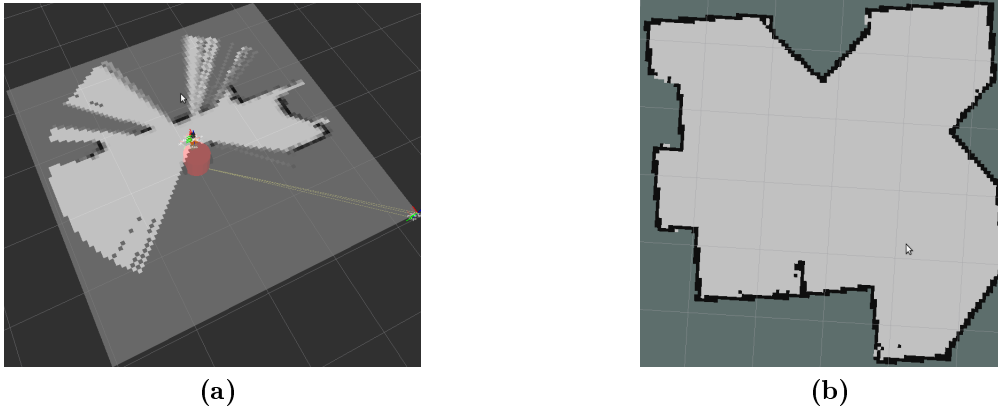
Uma das maiores vantagens da abordagem baseada em grelhas sobre as outras duas é que ela tem em conta as características dos sensores, o que permite modelar explicitamente o espaço livre, fornecendo, por exemplo, conhecimento útil para aplicações de navegação.

Neste trabalho, este tipo de representação do ambiente é proposto com uma intenção de mapa local.

#### 2.2.3.1 Mapa de ocupação local

O mapa de ocupação local (como por exemplo, figura 2.6a) refere-se à criação em tempo real de grelhas de ocupação local de alta resolução da zona que se encontra próxima ao robô móvel. A informação concedida por este tipo de grelhas torna-se útil em situações em que são exigidas manobras complexas em ambientes reais.

Com a existência destas grelhas locais para situações de emergência, o desempenho do sistema robótico está a ser melhorado. Tipicamente, existe um raio de segurança em torno do robô por forma a assegurar que ele não se aproxime dos obstáculos. Se uma grelha local está a ser construída, torna-se possível operar muito mais perto de obstáculos. Esta é uma situação bastante útil em ambientes *indoor*, onde este tipo de conceito poderá permitir ao robô deslocar-se através de locais apertados, como por exemplo, atravessar uma porta.



**Figura 2.6:** Visualização, em Rviz, de modelos do ambiente baseados em mapas de ocupação. (a) Grelha de ocupação local 2D. (b) Grelha de ocupação global 2D.

### 2.2.3.2 Mapa de ocupação global

Comparativamente ao mapa de ocupação local, um mapa de ocupação global (como por exemplo, figura 2.6b) deve ter menor resolução. Na prática, se assim não fosse, seria bastante exigente para o robô móvel criar e actualizar um mapa muito preciso. O principal objectivo de um mapa global é permitir o planeamento eficaz em longas distâncias. Neste caso, é mais importante que o sistema de planeamento determine com precisão se o robô pode chegar a um ponto distante, do que fornecer detalhes de alta resolução acerca dos obstáculos que surgiram ao longo do percurso. Os pequenos detalhes do ambiente não são importantes, desde que eles não afectem as operações no planeamento de uma trajectória. Por exemplo, a largura de uma porta não é relevante, desde que ela esteja bem representada no mapa global e o robô reconheça que pode atravessá-la correctamente.

## 2.3 Grelhas de ocupação probabilística

Na representação baseada em grelhas, a cada região do ambiente (definido como célula) está associado uma estimativa da probabilidade de ocupação. O método Bayesiano permite a actualização espaço-temporal dessas estimativas probabilísticas que caracterizam o estado de ocupação de cada célula. Para além deste método, convém enunciar outras técnicas que também permitem estimar as probabilidades de ocupação. Em [Hoseinnezhad et al., 2002] formulam três diferentes métodos para realizar a integração e actualização da informação sensorial numa grelha, que são baseados na abordagem Bayesiana, lógica difusa e *Dempster-Shafer*. Os autores concluíram que a abordagem Bayesiana concretiza mapas mais precisos.

### 2.3.1 Inferência Bayesiana

A inferência Bayesiana é um método de inferência probabilística em que as observações (eventos) são utilizadas para actualizar ou inferir a probabilidade de que uma hipótese possa ser verdade. Nesta inferência, a incerteza do evento é expressa em termos de uma distribuição probabilística. O método de inferência Bayesiana é formulado, de um modo específico, tendo por base o teorema de Bayes. Considerando os eventos  $A$  e  $B$  e as respectivas probabilidades  $P(A)$  e  $P(B)$ , esse teorema (ou regra) é dado por:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \quad (2.1)$$

onde  $P(A|B)$  representa a probabilidade condicional do evento  $A$  dada a ocorrência do evento  $B$ . A utilização deste formalismo numa estrutura em grelhas foi inicialmente proposto em [Elfes, 1989]. A grelha de ocupação é uma representação espacial estocástica em que cada célula  $C$  é caracterizada por uma variável de estado de ocupação  $s(C)$ . Esse estado é definido entre duas possibilidades exclusivas, ocupado  $s(C) = O$  ou livre  $s(C) = E$ . Cada célula  $C$  contém uma probabilidade de estar ocupada  $P(O)$  e estar livre  $P(E)$ . Assumindo que todas as células são independentes entre si e satisfazendo um dos axiomas da teoria da probabilidade  $P(O) + P(E) = 1$ .

### 2.3.2 Aplicação da inferência Bayesiana para grelhas de ocupação

Considere-se a existência de um modelo em grelhas de ocupação para representar um determinado ambiente constituído por obstáculos estáticos. Para cada célula  $C$  é atribuída uma probabilidade de ocupação condicional  $P(O|Y_{1:n})$ , onde  $Y_{1:n} = \{Y_1, Y_2, \dots, Y_n\}$  representa o conjunto de  $n$  medidas  $Y$ .

De acordo com o teorema de Bayes sabe-se que

$$P(O|Y_n, Y_{1:n-1}) = \frac{p(Y_n|O, Y_{1:n-1})P(O|Y_{1:n-1})}{p(Y_n|Y_{1:n-1})} \quad (2.2)$$

onde  $p(Y_n|O, Y_{1:n-1})$  representa a probabilidade de ocupação da célula  $C$  tendo em conta o modelo do sensor.

Se os erros de medida são independentes entre si no tempo

$$p(Y_n|O, Y_{1:n-1}) = p(Y_n|O) \quad (2.3)$$

Usando a marginalização do denominador obtém-se

$$p(Y_n|Y_{1:n-1}) = p(Y_n, O|Y_{1:n-1}) + p(Y_n, E|Y_{1:n-1}) \quad (2.4)$$

Substituindo (2.3) em (2.4), resulta que

$$p(Y_n|Y_{1:n-1}) = p(Y_n|O)P(O|Y_{1:n-1}) + p(Y_n|E)P(E|Y_{1:n-1}) \quad (2.5)$$

E assim (2.2) pode reescrever-se na forma

$$P(O|Y_{1:n}) = \frac{p(Y_n|O)P(O|Y_{1:n-1})}{p(Y_n|O)P(O|Y_{1:n-1}) + p(Y_n|E)P(E|Y_{1:n-1})} \quad (2.6)$$

onde  $Y_n$  indica uma nova leitura para actualizar a grelha. Este método estima recursivamente a probabilidade *a posteriori*  $P(O|Y_{1:n})$ , a partir do conhecimento da probabilidade *a priori*  $P(O|Y_{1:n-1})$  e da probabilidade condicional  $p(Y_n|O)$ .

Na abordagem considerada em [Elfes, 1989], o modelo de ruído da medida  $p(Y_n|z_n)$  corresponde à probabilidade entre o valor  $Y$  medido pelo sensor conhecendo o valor real  $z_n$ . Para calcular o termo  $p(Y_n|O)$  de (2.6), a partir de  $p(Y_n|z_n)$  é utilizado o teorema de *Kolmogorov* [Elfes, 1989, Berger, 1985] sobre as densidades de probabilidade condicional da medida para todas as configurações de grelha possíveis. O objectivo é identificar as células pertencentes ao estado ocupado. A cada nova leitura devolvida pelo sensor, todas as células dentro do seu campo de visão são actualizadas de acordo com esta abordagem.

Inicialmente não existe qualquer conhecimento acerca da probabilidade de ocupação das células, ou seja, a cada célula  $C$  estão associadas as probabilidades *a priori*  $P(O|Y_{1:n-1}) = P(E|Y_{1:n-1}) = 0.5$ . Relembre-se que, em qualquer instante de tempo, a soma dessas probabilidades é sempre unitária.

### 2.3.3 Fusão entre grelhas de ocupação

Para fundir directamente duas grelhas independentes (por exemplo), propõe-se a utilização de (2.7), a partir do conhecimento obtido em (2.6)

$$P_{1 \odot 2} = \frac{P_1 P_2}{P_1 P_2 + (1 - P_1)(1 - P_2)} \quad (2.7)$$

onde  $\odot$  significa o operador de combinação,  $P_1 = P(O|Y_n^1)$  e  $P_2 = P(O|Y_n^2)$  representam as probabilidades de ocupação da grelha  $G^1$  construída pela medida  $Y_n^1$  e da grelha  $G^2$  a partir de  $Y_n^2$ , respectivamente.

Usando a regra de Bayes e considerando a independência das medidas é possível obter que

$$P(O|Y_n^1, Y_n^2) = \frac{P(O|Y_n^1)P(O|Y_n^2)}{P(O|Y_n^1)P(O|Y_n^2) + P(E|Y_n^1)P(E|Y_n^2)} \quad (2.8)$$

A integração temporal da grelha é realizada entre as probabilidades  $P_{t-1}^G$  da grelha  $G$  conhecida no instante  $t - 1$  e as probabilidades  $P_t^{new}$  da grelha  $G^{new}$  obtida no instante  $t$ . Do mesmo modo, expressa-se que

$$P_t^G = P_{t-1}^G \odot P_t^{G^{new}} = \frac{P_{t-1}^G P_t^{G^{new}}}{P_{t-1}^G P_t^{G^{new}} + (1 - P_{t-1}^G)(1 - P_t^{G^{new}})} \quad (2.9)$$

### 2.3.4 Bayesian Occupancy Filter (BOF)

O filtro de ocupação Bayesiano, designado por BOF (*Bayesian Occupancy Filter*) [Coué et al., 2006, Tay et al., 2007], consiste numa técnica que combina dois métodos clássicos: grelhas de ocupação 2D para representar o ambiente e um filtro estocástico que permite ter em conta a dinâmica do ambiente que está a ser modelado. Esta abordagem estima recursivamente as distribuições de probabilidade de cada célula utilizando a observação sensorial. Análogo a todos os algoritmos de filtros estocásticos, o BOF também tem dois estágios para obter a distribuição de probabilidade posterior: predição e estimação. No estágio de predição é calculada uma previsão *a priori* do estado da célula usando um modelo definido que não tem em conta a observação actual do sensor. Em seguida, no estágio de estimação, a distribuição posterior do estado é calculada usando a distribuição *a priori* e a observação actual do sensor. Esta abordagem permite que o robô móvel modele o ambiente dinâmico e acompanhe a evolução temporal do seu ambiente.

O BOF demonstra-se uma ferramenta poderosa que prevê a evolução dos estados de ocupação das células da grelha mesmo que nenhuma medida esteja disponível, num curto período de tempo. Em contra partida, este método ainda tem alguns inconvenientes. Em [Nègre et al., 2014] apresentam uma nova representação para o BOF, que descreve o ambiente através de uma combinação de ocupação estática e dinâmica.

## 2.4 Percepção local para navegação

O desenvolvimento de um sistema de percepção espaço-temporal local é um elemento chave para melhorar o desempenho na navegação do robô móvel. A informação do ambiente, recorrendo a dados recolhidos a partir de múltiplos sensores, passa a estar representado num modelo, que pode ser usado para resolver dois problemas principais: detecção/seguimento dos obstáculos e determinação do espaço livre. Desta forma, o modelo construído poderá auxiliar o robô móvel a executar as suas tarefas ou acções de forma segura.

### Detecção e seguimento obstáculos

A percepção local deve ser capaz de detectar e caracterizar os obstáculos presentes no ambiente de operação do robô móvel. Esta acção pode ser dividida em duas tarefas elementares: a detecção e a localização de objectos no ambiente e estimativa do movimento.

A detecção está responsável por identificar o número de objectos presentes no ambiente. No caso de existirem, deve-se complementar com a localização dos obstáculos detectados de modo a perceber os seus possíveis comportamentos. Estas duas tarefas são cruciais para todos

os sistemas de percepção e têm sido resolvidas por sistemas do tipo DATMO. Esta é uma metodologia tradicional que, com base nas medidas sensoriais, modela objectos em movimento através da estimação do estado do objecto em movimento e o modelo de movimento correspondente [Petrovskaya et al., 2011].

Finalmente, uma vez que os objectos são detectados e acompanhados, eles ainda podem ser classificados ou identificados, o que permite aumentar o nível de conhecimento sobre o tipo de objecto. Em [Li, 2013], apresentam um sistema de percepção que utiliza uma câmara stereo para detectar e reconhecer objectos dinâmicos em torno de um veículo em movimento. As informações do movimento e reconhecimento dos obstáculos são integradas em grelhas de ocupação.

### Determinação do espaço livre

De modo dual à detecção dos obstáculos, a percepção é útil para determinar as regiões livres, ou seja, as regiões em que o robô móvel se pode mover sem risco de colisão. Estas regiões são geralmente determinadas de maneira implícita, ou seja, deduzidas durante os procedimentos de detecção e seguimento de obstáculos. No campo dos robôs móveis ou veículos inteligentes muitos dos algoritmos de planeamento de trajectórias baseiam-se numa representação explícita acerca do espaço navegável. Por conseguinte, é necessário identificar e caracterizar o espaço de acordo com a integração de erros nas medidas, limitações dos sensores e conhecimentos prévios para calcular o espaço navegável.

No âmbito desta dissertação, um modelo do ambiente que se tem demonstrado eficaz e consistente para resolver os dois problemas enunciados é o mapa de ocupação. Na tabela 2.2 são apresentados alguns dos trabalhos desenvolvidos ao nível da percepção modelada neste tipo de mapas. Estes trabalhos, para além de apresentarem soluções para os problemas enunciados, têm aplicações práticas em tarefas de navegação autónoma ou semi-autónoma de robôs móveis ou veículos inteligentes. O desenvolvimento de sistemas robóticos móveis que apresentem elevado grau de autonomia, durante a sua navegação em ambientes complexos, exige técnicas de modelação do ambiente avançadas.

#### 2.4.1 Navegação autónoma

A implantação de sistemas robóticos autónomos em ambientes reais depende fortemente dos mapas de ocupação criados a partir do ambiente. Um ambiente pode ser complexo, dinâmico, constituído por obstáculos vulneráveis, como pessoas, e/ou em qualquer outro tipo de condições. A exigência do robô estar equipado com dispositivos sensoriais avançados, capazes de originar progressos que se verifiquem nos mapas do ambiente, é cada vez maior, o que levará a novas e melhores soluções de navegação autónoma, especialmente durante a execução de tarefas como planeamento de trajectórias.

Qualquer sistema robótico móvel autónomo deve ter a capacidade de desempenhar as operações no ambiente sem a necessidade de um operador externo. Todos os processos de tomadas



**Tabela 2.2:** Aplicações para sistemas de navegação recorrendo a mapas de ocupação local.

Instituição	Objectivo	Métodos utilizados	Sensores
Universidad de Zaragoza [Montesano et al., 2008]	Modelar os elementos estáticos e dinâmicos em ambientes <i>indoor</i> . Posteriormente, os modelos do ambiente são utilizados para planeamento de trajectórias de uma cadeira de rodas autónoma em cenários dinâmicos e desconhecidos.	Recorrem a uma grelha de ocupação binária local para modelar os espaços livres ou ocupados do ambiente. Definem um filtro de Kalman estendido (EKF) para fazer seguimento aos obstáculos que se movem.	Laser.
Case Western Reserve University [Murarka and Kuipers, 2009]	Percepcionar e classificar os riscos que impossibilitam a navegação autónoma de uma cadeira de rodas, tais como, escadas, rampas acentuadas ou saliências.	Utilizam uma grelha de ocupação 3D actualizada probabilisticamente pelo método de Bayes. A grelha 3D é segmentada por forma a identificar os diversos perigos. Criação de uma grelha local 2D com a devida informação acerca dos resultados da segmentação.	Laser e câmara de visão <i>stereo</i> .
Ulm University [Weiss et al., 2007]	Construir uma grelha de ocupação com o propósito de detectar os limites da estrada e determinar o corredor de espaço livre para condução de veículos em ambientes urbanos e auto-estradas.	Propõem a utilização de um novo modelo para o sensor, onde a actualização da grelha é realizada através do teorema de Bayes. A abordagem em grelhas de ocupação local permite separar os obstáculos estáticos ou dinâmicos.	Laser.
Otto-von-Guericke-Universität Magdeburg [Nguyen et al., 2012]	Desenvolver um sistema de protecção de pedestres para ambientes urbanos, assistindo activamente o condutor para prevenir possíveis colisões.	Utilizam um modelo do sensor apropriado e mapeam as medidas numa grelha de ocupação. Recorrem à teoria de <i>Dempster-Shafer</i> para actualizar a informação na grelha. Através de conjuntos ( <i>clusters</i> ) de células da grelha pretendem detectar o obstáculo e segui-lo, por forma a actualizar a grelha de forma eficaz.	Câmara de visão <i>stereo</i> .
Tech. Univ. München [Weiherer et al., 2013]	Desenvolver uma arquitectura para um sistema de percepção em ambientes altamente dinâmicos, com aplicação em ADAS.	Combinam uma representação num mapa de intervalo 2D e um modelo com representação baseado no objecto. Demonstram como as probabilidades de ocupação podem ser atribuídas e actualizadas neste mapa segundo o teorema de Bayes. Mapa de intervalo 2D é um novo conceito que pretende substituir as conhecidas grelhas.	Laser.

de decisão estão totalmente dependentes do robô. A navegação autónoma continua a ser um objectivo em que a investigação se tem concentrado e que está cada vez mais presente (por exemplo, o veículo autónomo desenvolvido pela Google). Contudo, o conceito de navegação semi-autónoma continua ainda bastante activo, no qual os mapas de ocupação do ambiente também se têm demonstrado como uma solução importante para resolver os vários problemas existentes neste tipo de navegação.

## 2.4.2 Navegação semi-autónoma

Neste caso, um operador humano já pode ter algum tipo de influência sobre o comportamento da máquina perante diversas situações que o ambiente apresente. Neste tipo de navega-

ação, o robô pode funcionar como um assistente que vai garantir a não colisão com os obstáculos. Nos últimos anos, os dois grandes sistemas que incorporam navegação semi-autônoma, que têm sido âmbito de investigação exaustiva, são: os sistemas de suporte na mobilidade e os sistemas de assistência na condução (ADAS).

### **Sistemas de suporte na mobilidade (aplicações *indoor*)**

Os robôs, compreendidos como sistemas de suporte na mobilidade, estão referenciados na área da robótica orientada à assistência do ser humano e visam melhorar de forma substancial a independência e qualidade de vida das pessoas com limitações motoras graves, interagindo com as mesmas e fornecendo-lhes a assistência necessária em tarefas específicas [Pires and Nunes, 2002, Lopes et al., 2013]. Este tipo de robôs, designadamente cadeiras de rodas robotizadas, podem integrar um tipo de controlador colaborativo [Fong, 2001] que faça a gestão da informação proveniente de comandos do utilizador (através de uma HMI) e de um sistema de segurança activo (através da informação proveniente do sistema de percepção local) durante a sua navegação. Esta abordagem permite que os seres humanos e a máquina possam trabalhar em conjunto, sendo esta uma característica importante que permite que o robô opere de forma eficaz e segura, tendo em conta as suas necessidades. Em particular, um controlador colaborativo concebido para aplicações em sistemas de suporte na mobilidade foi desenvolvido no ISR [Lopes, 2012].

### **Sistemas de assistência na condução (ADAS) (aplicações *outdoor*)**

Recentemente, devido a razões de segurança nas estradas, os sistemas de assistência na condução (designados por ADAS) têm obtido uma grande atenção por parte da comunidade de investigação e da indústria. O desenvolvimento e aplicação deste tipo de sistemas iniciou-se no ano 1999 (segundo [Nguyen et al., 2012]), nomeadamente com o sistema *Adaptive Cruise Control* que detecta automaticamente o veículo na zona frontal que se encontra em movimento e ajusta a própria velocidade para manter uma distância de segurança adequada. Actualmente, existe uma ampla variedade destes sistemas, tais como *Side Assist*, que observa objectos em ângulos mortos para ajudar o condutor durante mudanças na faixa de rodagem, ou *Park Assist*, que presta assistência quase completa durante as manobras de estacionamento.

Os ADAS têm sido uma solução que proporciona maior conforto e segurança para os condutores de veículos automóveis. Desde a sua existência no mercado que estes sistemas têm contribuído para a redução de acidentes automóveis através da prevenção de possíveis colisões. Contrariamente ao sistema de suporte na mobilidade, utilizado em aplicações para ambientes *indoor*, o sistema de percepção local desenvolvido para ADAS apresenta-se com outro tipo de desafios, tais como lidar com ambientes *outdoor* altamente dinâmicos (por exemplo, outros veículos automóveis que se deslocam a altas velocidades), considerar uma maior área em torno do veículo e enfrentar alterações climáticas rigorosas.

**Tabela 2.3:** Métodos de fusão multi-sensorial usando a representação em grelhas de ocupação.

Instituição	Método	Descrição	Sensores
Czech Technical University in Prague [Stepan et al., 2005]	Recorre à fórmula bayesiana.	Cada sensor constrói a sua grelha independentemente. A combinação das grelhas 2D tem em conta que o sensor câmara é mais preciso que os sonares.	Sonares e câmara monocular.
Universität Bremen [Pathak et al., 2007]	Uso de uma fórmula designada por <i>Superbayesian Independent Opinion Pool</i> .	Criação de um modelo tridimensional do sensor utilizado para fundir informação da nuvem de pontos. A fusão é realizada num mapa de ocupação local cartesiano comum.	Laser e câmara de visão <i>stereo</i> .
University of Technology of Compiègne [Moras et al., 2011]	Utilização da teoria de <i>Dempster-Shafer</i> .	É utilizada uma abordagem credível para modelar a informação sensorial e para fundir as informações conflituosas num mapa global.	Dois sensores Laser.
INRIA Grenoble [Adarve et al., 2012]	Método designado por <i>Linear Opinion Pool</i> .	A ideia é desempenhar o processo de fusão como uma soma pesada das observações sensoriais. Cada sensor devolve duas quantidades: a sua estimativa acerca da ocupação da célula e um peso que indica a confiança dessa estimativa.	Laser e câmara de visão <i>stereo</i> .

## 2.5 Fusão multi-sensorial em mapas de ocupação

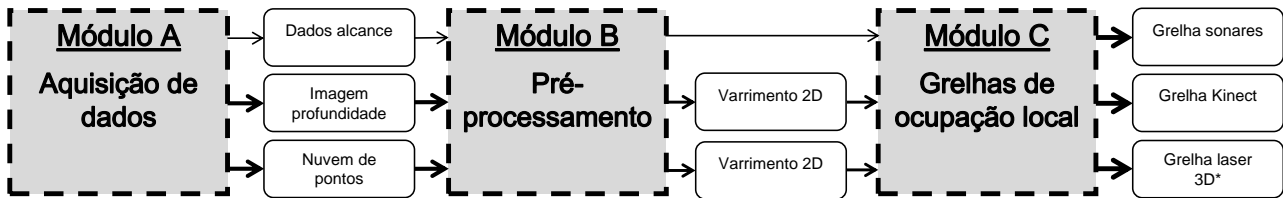
A partir de um conjunto de sensores exteroceptivos incorporados num robô móvel ou veículo é possível combinar a informação proveniente de todos esses sensores, com o intuito de atingir uma representação mais robusta acerca do ambiente. Reconhecendo que todos os sensores apresentam limitações, haverá situações onde alguns deles, por algum motivo, poderão eventualmente falhar nas suas capacidades de observação. A principal expectativa ao fundir informação entre múltiplos sensores é aumentar o desempenho nas capacidades do sistema de percepção, explorando os recursos individuais de cada uma das fontes sensoriais até à obtenção de um único modelo acerca do ambiente.

Alguns métodos de fusão multi-sensorial em mapas de ocupação têm sido propostos na literatura e estão apresentados na tabela 2.3.

# Capítulo 3

## Grelhas de ocupação local

No capítulo 1, mais precisamente na secção 1.3, foi apresentada uma visão geral sobre a estrutura do método para obtenção da grelha de ocupação local 2D. Neste capítulo, realiza-se uma análise mais aprofundada sobre os módulos A, B e C, examinando o funcionamento interno e metodologias elaboradas em cada um deles. A figura 3.1 demonstra o fluxo de informação à entrada e à saída de cada módulo. O módulo C fornece ao sistema robótico móvel três grelhas de ocupação local 2D construídas em tempo real.



**Figura 3.1:** Estrutura para construção das grelhas de ocupação local 2D com base num conjunto de medidas sensoriais. Cada vez que os dados de alcance dos sonares, a imagem profundidade do Kinect ou a nuvem de pontos do laser 3D\* estão disponíveis é construída uma grelha de ocupação local.

A localização do robô no mundo não é necessária para os desenvolvimentos apresentados neste capítulo. Caso o robô móvel se desloque, cada vez que uma nova leitura estiver disponível no tempo, é gerada uma nova grelha em torno dele. Não é contemplado nenhum método de integração temporal nem nenhum método para fundir a informação proveniente das grelhas contruídas isoladamente. Essas questões serão abordadas no próximo capítulo.

Este capítulo está organizado da seguinte forma: na secção 3.1 apresentam-se as modalidades utilizadas e o tipo de informação que disponibilizam; na secção 3.2 descrevem-se as técnicas responsáveis por converter os dados para o espaço 2D; por fim, na secção 3.3, são apresentados os métodos para construção de grelhas de ocupação local 2D.

### 3.1 Módulo de aquisição de dados

A maioria dos ambientes *indoor*, tais como escritórios ou cozinhas, apresentam situações complexas que podem provocar a colisão durante a navegação do robô móvel. Para perceber de forma eficaz o ambiente, evitando que este tipo de problema ocorra, propõe-se a utilização de múltiplos sensores por forma que a confiança do robô em tomar decisões seguras é reforçada.

A aquisição da informação sensorial foi realizada utilizando um computador com o *software* ROS e a quantidade de dados extraídos varia consoante o sensor utilizado:

- O sonar proporciona um único dado a cada leitura efectuada. Utilizando um anel constituído por três sonares, em cada instante temporal passam a estar disponíveis três dados. O sonar tem um campo de visão de  $40^\circ$ . O anel foi concebido de maneira a não sobrepôr os campos de visão dos sonares, preenchendo um campo de visão de  $120^\circ$ .
- O Kinect devolve uma imagem de profundidade com dimensão  $640 \times 480$  píxeis, considerada como uma estrutura de dados matricial. Cada elemento dessa matriz tem uma correspondência na distância medida entre o sistema referencial do sensor e os obstáculos observados. Horizontalmente o campo de visão deste sensor é de  $57^\circ$  e verticalmente de  $49^\circ$ .
- O sensor laser 3D\* disponibiliza uma nuvem de pontos que resultou da sucessiva acumulação de varrimentos 2D em diferentes posicionamentos angulares do servo-motor. Horizontalmente o seu campo de visão é de  $270^\circ$  (respectivo ao laser) e verticalmente o pré-definido no servo-motor.

Após o pré-processamento, a construção das três grelhas de ocupação local 2D utilizam a informação: dos três dados devolvidos pelos sonares; da estrutura matricial do Kinect e da nuvem de pontos do laser 3D\*.

## 3.2 Módulo de pré-processamento

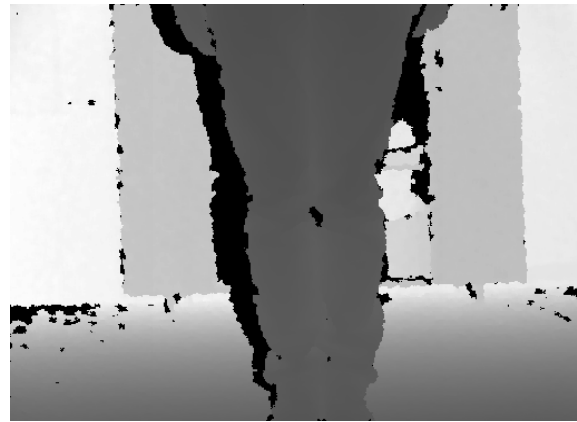
A existência deste pré-processamento é justificada pelo facto da modelação do ambiente ser 2D. Apesar de existirem diversas motivações que têm levado a investigação a modelar o ambiente em 3D, no campo da robótica móvel e/ou veículos, ainda existem fortes razões para continuar com modelos 2D, nomeadamente, pelo facto de o robô ser, frequentemente, utilizado para navegar em ambientes planares. Ainda assim, seria necessário uma maior quantidade de memória e capacidade de processamento para os modelos 3D. A modelação 2D é computacionalmente menos dispendiosa e um modelo adequado para que o robô desempenhe as suas tarefas em segurança, por exemplo, durante o planeamento de trajetórias.

### 3.2.1 Projecção dos dados do Kinect para 2D

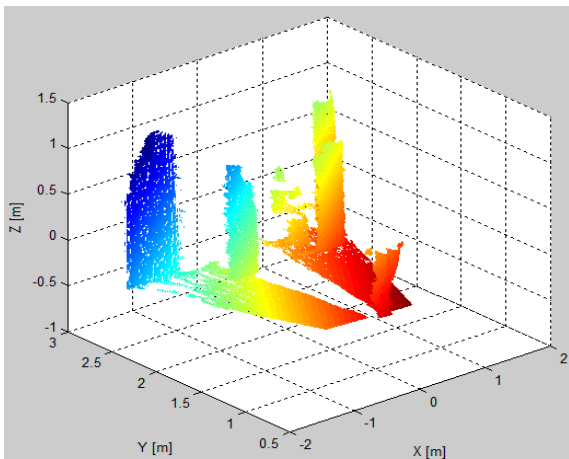
A matriz ( $I^{depth}$ ) que representa os dados correspondentes à imagem de profundidade do Kinect é projectada num vector de dados (designado por varrimento 2D). O varrimento 2D contém informação da distância aos obstáculos que se encontram mais próximos do sistema referencial do Kinect. Essa distância é identificada por um feixe de amplitude  $z_k$  e um ângulo direccional  $\theta_k$ , obviamente limitado ao campo de visão horizontal do Kinect, com  $k = 1 \dots n$ .



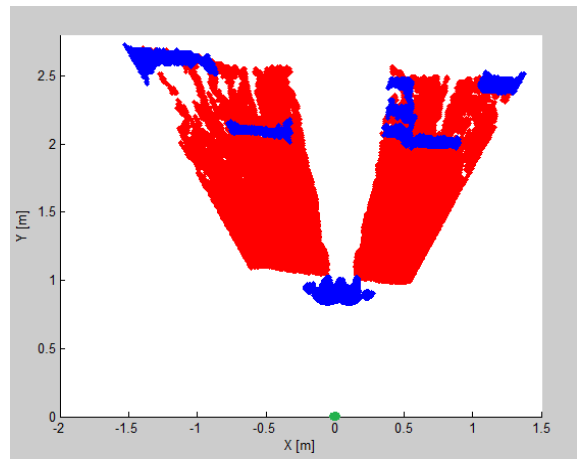
(a) Imagem RGB obtida do Kinect.



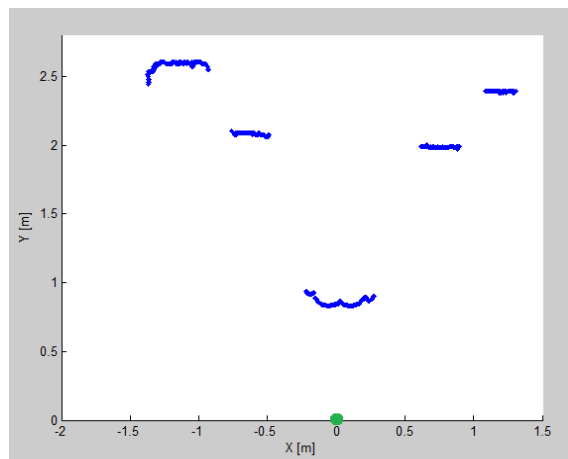
(b) Imagem profundidade obtida do Kinect.



(c) Nuvem de pontos no plano 3D.



(d) Projecção da nuvem de pontos no plano 2D.



(e) Extração do contorno dos obstáculos.

**Figura 3.2:** Procedimentos para obtenção do varrimento 2D a partir da imagem de profundidade obtida do Kinect.

A figura 3.2a mostra uma imagem RGB obtida a partir do Kinect. Esta imagem é exibida apenas com o propósito de mostrar a visão deste sensor, facilitando assim o processo de comparação com a imagem de profundidade da figura 3.2b. Quanto mais escuros forem os píxeis, mais próximos se encontram os obstáculos. Por outro lado, os píxeis de cor branco representam dados inválidos, que ocorrem pelo facto de os obstáculos estarem fora da gama de alcances do Kinect ou por uma série de factores intrínsecos a este sensor.

Considere-se a seguinte sequência de passos até se obter um varrimento 2D com  $n$  pontos:

1. Cálculo da nuvem de pontos 3D para cada elemento da matriz  $I^{depth}$ . Para cada elemento está associado um ponto no espaço de coordenadas Cartesianas  $(X, Y, Z)$ , resultando assim num conjunto de 307200 pontos. A figura 3.2c ilustra o resultado deste cálculo com base na imagem de profundidade da figura 3.2b. Neste caso, considera-se que o centro de coordenadas do sistema referencial do Kinect está localizado em  $(X = 0, Y = 0, Z = 0)$ .
2. Projecção da nuvem de pontos no espaço 2D. A figura 3.2d demonstra o resultado da projecção no plano do sensor para  $Z = 0$ . Em seguida, os pontos que possuem qualquer altura acima (pontos azuis) ou abaixo (pontos vermelhos) de uma altura mínima (definida por um *threshold*) devem ser identificados.
3. Remoção dos pontos vermelhos do conjunto e extracção o contorno dos obstáculos a partir dos restantes pontos. Para cada ponto com a mesma componente em  $X$  selecciona-se a distância mínima para todas as suas componentes em  $Y$ , resultando assim num varrimento 2D de  $n = 480$  pontos. A figura 3.2e demonstra o varrimento 2D esperado.

A técnica utilizada no passo 2 permite a remoção dos pontos que contêm informação desnecessária (por exemplo, da superfície do chão). Esta é uma técnica simples, rápida e eficiente e pode ser aplicável neste trabalho, uma vez que o sensor está fixo relativamente ao sistema referencial do robô e a superfície do chão é assumida como planar. O valor do *threshold* é estabelecido de acordo com a altura entre o sistema de coordenadas do sensor e a superfície do chão. Desta forma, essa superfície não surge como um obstáculo para o robô.

### 3.2.2 Projecção dos dados do laser 3D\* para 2D

A solução aqui proposta segue procedimentos semelhantes aos exercidos na projecção dos dados do Kinect para 2D. Contudo não existe uma imagem profundidade discretizada do ambiente. A informação da profundidade está representada numa nuvem de pontos, onde cada ponto é identificado pela respectiva coordenada Cartesiana  $(X, Y, Z)$ .

A operação realizada nesta etapa tem por objectivo continuar a manter a informação mais relevante sobre os obstáculos mais próximos ao sistema referencial do sensor. Do resultado desta operação surge um varrimento 2D (vector de dados). O ângulo máximo ( $angle_{max}$ ), o ângulo mínimo ( $angle_{min}$ ) e a distância angular entre dois feixes consecutivos ( $\Delta angle$ ) são as propriedades que definem esse varrimento.

Considere-se a seguinte sequência de passos até se obter o varrimento 2D (vector):

1. Determinação dos  $n$  pontos que definem a dimensão do vector através da expressão:  $\lceil \frac{angle_{max} - angle_{min}}{\Delta angle} \rceil$ . O número de feixes é igual ao número de pontos.
2. Para cada ponto  $(X, Y, Z)$ , calcular a distância euclidiana,  $distance = \sqrt{X^2 + Y^2}$ , e o ângulo,  $angle = atan2(\frac{Y}{X})$ , relativamente ao sistema de coordenadas do sensor. Apenas os pontos que estão dentro das propriedades definidas para o vector e de coordenada  $Z > threshold$  são armazenados. Neste caso, a eliminação da informação desnecessária (por exemplo, da superfície do chão) segue a técnica aplicada na projecção dos dados da Kinect para 2D.
3. Para cada ponto  $(X, Y, Z)$  armazenado no passo 2, determinar o índice que identifica a sua posição no vector através do cálculo:  $index = \lceil \frac{angle - angle_{min}}{\Delta angle} \rceil$ . Se o valor de profundidade já armazenado em  $index$  for superior a  $distance$ , o menor valor de profundidade passa a ser dado por  $distance$ .

### 3.3 Módulo das grelhas de ocupação local

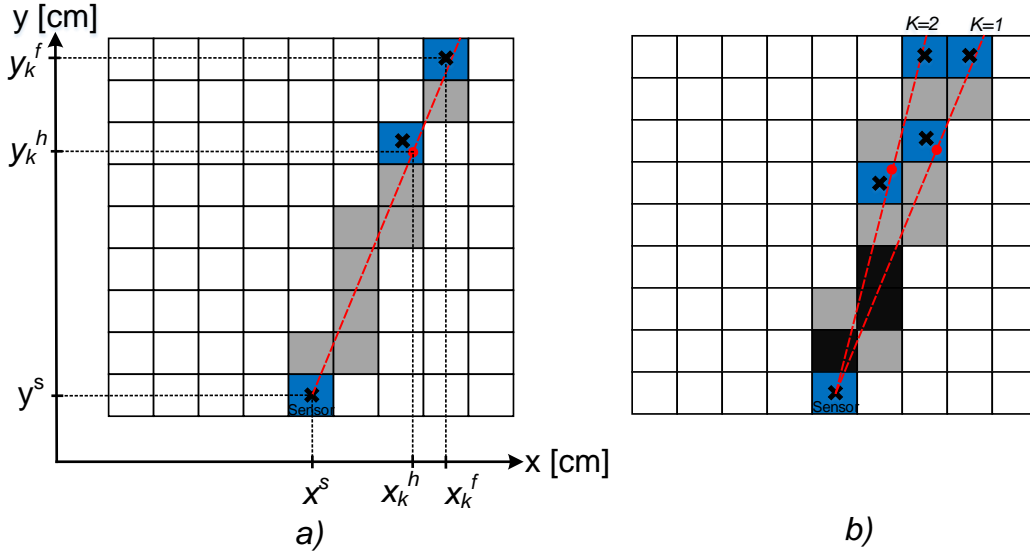
Nesta secção, descreve-se as técnicas utilizadas para construção das grelhas de ocupação local 2D, supondo que em cada instante de tempo  $t$  as leituras de distância  $Z_t$  devolvidas pelos sensores estão disponíveis. A grelha  $G$  é definida como um conjunto de células  $C$  uniformemente distribuídas numa área definida por  $W$  e consiste numa matriz com  $m^G$  linhas e  $n^G$  colunas, em que cada elemento é identificado pelo índice  $i$  das linhas e pelo índice  $j$  das colunas. Os elementos representam as células e estas têm dimensões iguais de resolução  $\Delta r$ . Cada célula  $C$  é caracterizada por uma variável de estado de ocupação  $s(C)$ , que é definido entre duas possibilidades exclusivas, ocupado  $s(C) = O$  ou livre  $s(C) = E$ .

N.º linhas $m^G$	N.º colunas $n^G$	Dimensão $x$	Dimensão $y$	Resolução $\Delta r$	N.º células $C$
81	81	405 cm	405 cm	5 cm	6561

**Tabela 3.1:** Características das grelhas locais 2D construídas.

Os métodos que determinam as probabilidades de ocupação de cada célula são semelhantes para todos os sensores utilizados, sendo distinguidos apenas pelo campo de visão, pelos parâmetros para o modelo e pela quantidade de informação disponível do sensor. Após a compreensão desses métodos, apresenta-se uma grelha de ocupação local 2D (com as características da tabela 3.1), que representa o ambiente em torno do robô móvel, construída com dados do sonar, Kinect e laser 3D\*, respectivamente. Para condições iniciais e em caso de falta de informação sensorial, o estado de ocupação de determinada célula é assumido como desconhecido.





**Figura 3.3:** a) Exemplo do método para delinear feixes pelo algoritmo de *Bresenham*. O círculo vermelho representa o ponto medido pelo sensor (já convertido em coordenadas Cartesianas). A localização do sensor e do ponto medido está identificada pelas células a azul. As células a cinza foram afectadas pela travessia do feixe. As cruces (x) indicam o centro geométrico da célula. b) Exemplo do procedimento de atribuição de probabilidades. As células mais escuras representam zonas conflituosas, ou seja, durante o processo de delinear feixes, tanto o feixe  $k = 1$  como o  $k = 2$  atribuíram uma probabilidade para essas células.

### 3.3.1 Métodos

#### Delinear feixes

Como descrito na secção 2.4, num determinado instante de tempo, as medidas devolvidas pelo Kinect e pelo laser 3D\* passaram a ser vistas como um varrimento 2D com  $n$  pontos, em que a cada um dos pontos está associado um feixe de amplitude  $z_k$  e um ângulo  $\theta_k$  relativamente ao centro de coordenadas do sensor, com  $k = 1 \dots n$ . Por questões de simplicidade e rapidez, para a criação da grelha dos sonares, considera-se que o seu campo de visão em forma de cone é visto também como um conjunto de  $n$  feixes com amplitudes iguais e de distâncias angulares  $\Delta\theta$  entre si. Todo este conjunto de pontos, descritos em coordenadas polares, pode ser rapidamente transformado num sistema de coordenadas Cartesiano de acordo com:

$$\begin{cases} x_k^h &= z_k \cos \theta_k \\ y_k^h &= z_k \sin \theta_k \end{cases} \quad \begin{cases} x_k^f &= z_{máx} \cos \theta_k \\ y_k^f &= z_{máx} \sin \theta_k \end{cases} \quad (3.1)$$

onde  $(x_k^h, y_k^h)$  são as coordenadas do ponto  $k$  na grelha,  $z_{máx}$  é uma amplitude relativamente longe do feixe e  $(x_k^f, y_k^f)$  são as coordenadas do ponto relativo a esse feixe, perante o conhecimento das coordenadas Cartesianas dos pontos, pretende-se determinar quais são os índices  $(i^h, j^h)$  e  $(i^f, j^f)$  da célula correspondente, através de:

$$\begin{cases} i^h &= \lceil \frac{x_k^h}{\Delta r} \rceil \\ j^h &= \lceil \frac{y_k^h}{\Delta r} \rceil \end{cases} \quad \begin{cases} i^f &= \lceil \frac{x_k^f}{\Delta r} \rceil \\ j^f &= \lceil \frac{y_k^f}{\Delta r} \rceil \end{cases} \quad (3.2)$$

onde  $\lceil \cdot \rceil$  significa o arredondamento para o valor inteiro superior mais próximo e  $\Delta r$  representa a resolução da grelha, posteriormente, utilizou-se o algoritmo de *Bresenham* [Bresenham, 1965] para delinear rectas imaginárias na grelha. Essas rectas são vistas como o conjunto de feixes. O algoritmo apenas precisa de conhecer as coordenadas Cartesianas  $(x^s, y^s)$  dos índices  $(i^s, j^s)$  da célula inicial (onde está posicionado o sensor) e  $(x^f, y^f)$  dos índices  $(i^f, j^f)$  da célula final (na direcionalidade da incidência no obstáculo) para determinar quais as coordenadas  $(x_l, y_l)$  das  $m$  células que estão afectadas pela travessia do feixe que liga as células inicial e final, onde  $l = 1 \dots m$ . A figura 3.3a ilustra os procedimentos deste método.

### Atribuição das probabilidades

Para cada uma das  $m$  células afectadas pelos  $n$  feixes é atribuída uma probabilidade de ocupação. Para realizar o processo de atribuição de probabilidades utilizou-se o modelo de incerteza Gaussiano. Considere-se, como exemplo, que para um dado feixe conhecem-se todas as coordenadas Cartesianas pelo método de delinear feixes. Posto isso, recorre-se à fórmula enunciada em [Adarve et al., 2012], dada por:

$$P(z|O_{i,j}) = \begin{cases} g(z, z^h) = \lambda \exp\left(-\frac{(z-z^h)^2}{2\sigma^2}\right), & \text{para } z \in [0, z^h] \\ \max\{0.5, g(z, z^h)\}, & \text{para } z \in [z^h, z_{máx}] \end{cases} \quad (3.3)$$

onde  $\lambda$  é um factor de escala,  $\sigma^2$  é a variância que caracteriza a incerteza do sensor,  $z$  e  $z^h$  são distâncias euclidianas dadas por:

$$z = \sqrt{(x_l - x^s)^2 + (y_l - y^s)^2} \quad z^h = \sqrt{(x_1^h - x^s)^2 + (y_1^h - y^s)^2} \quad (3.4)$$

Assim, para além de identificar a célula de incidência do ponto, é possível propagar probabilidades sobre as células que o feixe atravessou até chegar a esse ponto. Para os  $n$  feixes calculam-se as probabilidades de modo semelhante.

O problema sucede quando dois ou mais feixes identificam a mesma célula, durante o processo de delinear feixes e atribuição de probabilidades. Essa zona pode ser considerada como conflituosa, uma vez que cada um dos feixes pode assumir uma probabilidade diferente. A figura 3.3b ilustra uma ocorrência dessa situação. A resolução de tal problema baseou-se no cálculo da média das probabilidades que cada feixe assume nessa célula. Para isso, utilizaram-se duas matrizes auxiliares com dimensões semelhantes à grelha de ocupação, designadas por *HitGrid* e *SumGrid*. A *HitGrid* é uma matriz que armazena o número de feixes que passam em determinada célula  $(i, j)$  e *SumGrid* é um somador das probabilidades que cada um dos

feixes assume em determinada célula  $(i, j)$ . No final, as probabilidades da grelha de ocupação são dadas por:

$$P(z|O_{i,j}) = \frac{SumGrid}{HitGrid} \quad (3.5)$$

A aplicação prática destes métodos na construção de grelhas de ocupação local 2D é verificada nas próximas secções deste capítulo. As grelhas estão centradas no sistema referencial do robô.

### 3.3.2 Grelha usando os sonares

Em cada instante de tempo, cada um dos três sonares utilizados devolve um dado de distância. Para a criação da grelha dos sonares, o seu campo de visão é transformado num conjunto de 160 feixes/pontos ( $n$ ) com amplitudes atribuídas pela leitura da distância. No entanto, a distância angular ( $\Delta\theta$ ) entre dois feixes consecutivos é de  $0.25^\circ$ . A figura 3.4b ilustra uma grelha de ocupação local 2D construída segundo estas condições, para o cenário representado na figura 3.4a.

### 3.3.3 Grelha usando o Kinect

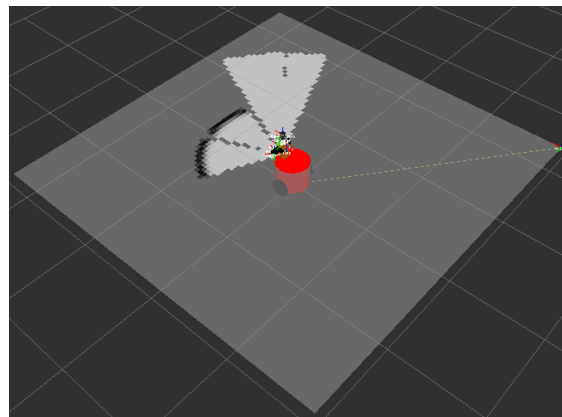
Em cada instante de tempo, o sensor Kinect devolve um varrimento bidimensional de 480 feixes/pontos ( $n = 480$ ). A distância angular ( $\Delta\theta$ ) entre dois feixes consecutivos é de aproximadamente  $0.12^\circ$ . A figura 3.5a ilustra uma grelha de ocupação local 2D construída segundo essas condições, para o cenário representado na figura 3.4a. Também é possível observar o conjunto de pontos projectados no plano do sensor obtidos a partir dos métodos apresentados na secção 3.2.

### 3.3.4 Grelha usando o laser 3D\*

Neste caso, em cada instante de tempo, o sensor laser 3D\* devolve um varrimento bidimensional de 1080 pontos ( $n = 1080$ ). A distância angular ( $\Delta\theta$ ) entre dois feixes consecutivos é de  $0.25^\circ$ . A figura 3.5b ilustra uma grelha de ocupação local 2D construída segundo essas condições, para o cenário representado na figura 3.4a. Também é possível observar o conjunto de pontos projectados no plano do sensor obtidos a partir dos métodos apresentados na secção 3.2.

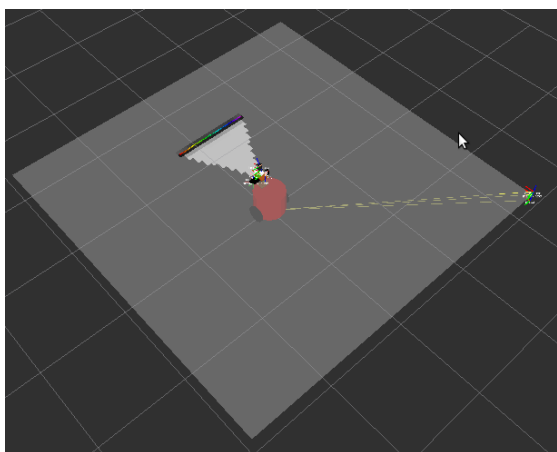


(a) Imagem RGB obtida pelo Kinect.

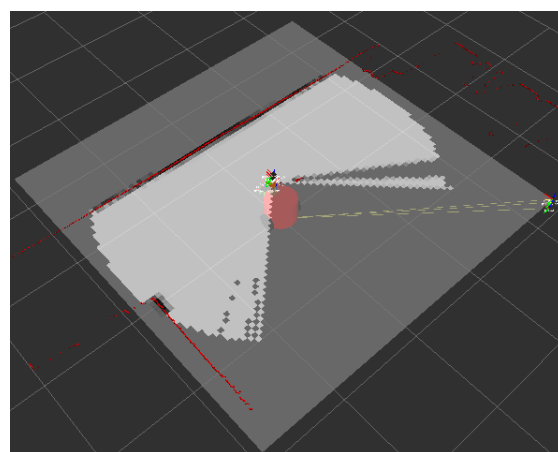


(b) Grelha de ocupação local 2D utilizando os dados dos sonares.

**Figura 3.4:** (a) Visualização da imagem RGB obtida a partir do Kinect, representando o cenário real. (b) Visualização no Rviz da grelha de ocupação local 2D resultante dos dados recolhidos dos três sonares, na interacção com o cenário da figura 3.4a.



(a) Grelha de ocupação local 2D utilizando os dados do Kinect projectados no plano 2D.



(b) Grelha de ocupação local 2D utilizando os dados do laser 3D\* projectados no plano 2D.

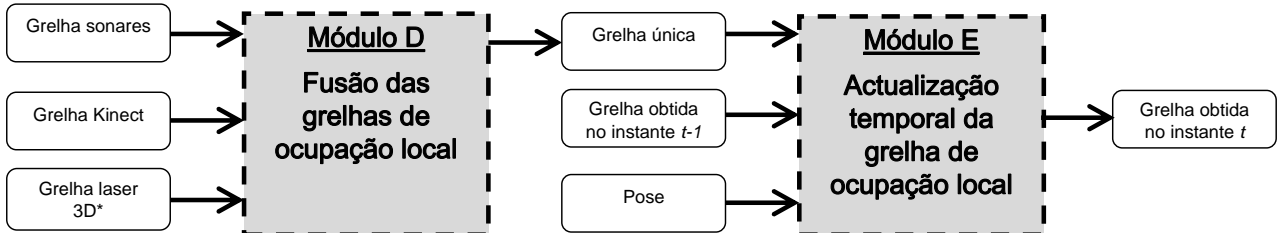
**Figura 3.5:** Visualização no Rviz da grelha de ocupação local 2D resultante dos dados recolhidos do Kinect e do laser 3D\*, na interacção com o cenário da figura 3.4a.

# Capítulo 4

## Fusão das grelhas de ocupação

Neste capítulo, apresentam-se detalhadamente os métodos utilizados para realizar a actualização espaço-temporal da grelha de ocupação local 2D. A actualização espacial refere-se à fusão entre as grelhas de ocupação local 2D construídas, independentemente, pelos sonares, pelo Kinect e pelo laser 3D\*. Em contrapartida, a actualização temporal indica a fusão entre a grelha de ocupação local 2D, resultante da actualização espacial, e a grelha obtida no instante de tempo  $t - 1$ .

Na figura 4.1 observa-se o fluxo de informação à entrada e à saída de cada módulo. No módulo D, as três grelhas de entrada estão disponíveis num tempo suficientemente próximo, para garantir que os sensores observam o ambiente nas mesmas condições. O módulo E fornece a grelha obtida após a actualização espaço-temporal, tendo em conta o deslocamento do robô. A obtenção de uma boa estimativa acerca da localização do robô no mundo (*pose*) é extremamente importante para o processo de actualização temporal.



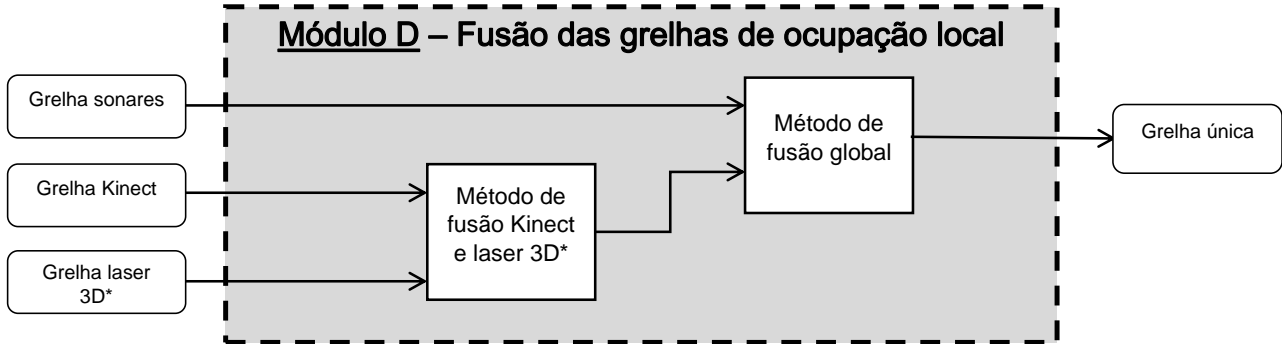
**Figura 4.1:** Estrutura para actualização espaço-temporal das grelhas de ocupação local 2D.

Este capítulo encontra-se dividido por duas secções, sendo que na primeira, secção 4.1, apresenta-se o método utilizado para fundir as grelhas de ocupação local 2D e na segunda, secção 4.2, descreve-se a técnica responsável por realizar a actualização temporal da grelha de ocupação local 2D.

### 4.1 Módulo de fusão das grelhas de ocupação local

É possível argumentar que se um laser 3D\* ou uma câmara Kinect está presente num sistema robótico móvel, nenhum outro sensor seria necessário para modelar correctamente o ambiente circundante, o que não é totalmente verdade. Em condições presentes num ambiente *outdoor*, ou mesmo *indoor*, só um desses sensores não seria suficiente para produzir uma solução válida e robusta para o problema da representação local em grelhas. Num conjunto de situações, tais

como, locais escuros, objectos demasiado próximos, resposta lenta dos sensores, nevoeiro, poeira ou fumo e portas/janelas/paredes de vidro pode-se produzir grelhas inválidas para alguns dos sensores utilizados. Essa conclusão advém da análise descrita na secção 6.1.



**Figura 4.2:** Estrutura para fusão entre as grelhas de ocupação 2D relativas aos sonares, ao Kinect e ao laser (3D). À saída obtém-se uma única grelha resultante da fusão.

Nesta secção apresentam-se os métodos utilizados para conjugar toda a informação sensorial numa única representação em grelha de ocupação local 2D. O principal objectivo ao realizar a fusão entre as grelhas construídas a partir dos sonares, Kinect e laser 3D\* é de alcançar um melhor desempenho, explorando as capacidades de combinar estas fontes sensoriais em situações onde algumas delas possam falhar.

A figura 4.2 especifica em pormenor como o processo de fusão em grelhas de ocupação local 2D é realizado. Inicialmente é realizada uma fusão entre as grelhas construídas a partir do Kinect e laser 3D\*. A grelha resultante é utilizada como entrada para fusão com a grelha construída a partir dos sonares. Recorreu-se às técnicas de fusão referenciado no capítulo 2, adaptando-as para a solução de fusão proposta.

Considere-se que as grelhas de ocupação local 2D construídas pelos ultra-som, Kinect e laser 3D\* são denotadas por  $G^U$ ,  $G^K$  e  $G^L$ , respectivamente. A probabilidade de cada célula  $C$  estar ocupada, a partir de um conjunto de observações  $Y_N$  para  $N = 1 \dots n$ , é dada por  $P(O|Y_N)$ .

### Método de fusão Kinect e laser 3D\*

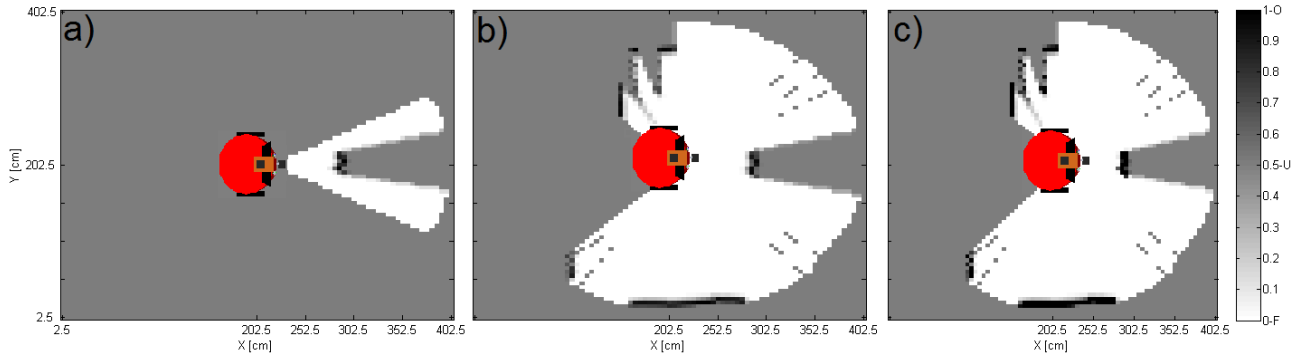
Neste método, as probabilidades de ocupação das respectivas grelhas  $G^K$  e  $G^L$  são integradas numa única grelha (definida por  $G^{K,L}$ ). As probabilidades de cada célula das grelhas são avaliadas independentemente antes de se realizar essa integração. Após a modificação desses valores probabilísticos, aplica-se a regra de Bayes para obter a grelha resultante da fusão entre o Kinect e o laser 3D\*. Estes procedimentos estão sequenciados no algoritmo 4.1.

Como este método de fusão é concretizado a partir do conhecimento das grelhas do Kinect e do laser 3D\*, ele não apresenta a capacidade de distinguir a probabilidade obtida a partir de cada tipo de sensor. Assim, a informação das grelhas é tratada de igual forma, independentemente da certeza nas leituras do sensor utilizado. Na figura 4.3 verifica-se o resultado da fusão entre as grelhas contruídas pelo Kinect e pelo laser 3D\* para o cenário da figura 3.2a. Nota-se claramente que algumas das probabilidades de ocupação foram reforçadas.

**Algoritmo 4.1** Método de fusão entre as grelhas de ocupação local 2D construídas a partir do Kinect e do laser 3D\*.

```

1: Input:  $P^K(O|Y_n)$  //Probabilidades de ocupação das células de  $G^K$ 
2:  $P^L(O|Y_n)$  //Probabilidades de ocupação das células de  $G^L$ 
3:  $T_0$  //threshold
4: Output:  $P^{K,L}(O|Y_n)$  //Probabilidades de ocupação das células de  $G^{K,L}$ 
5:
6: for all células  $C$  do
7: //Linearização das probabilidades de ocupação de  $G^K$ 
8: if  $P^K(O|Y_n) > T_0$  then
9:  $P^{K_{aux}}(O|Y_n) \leftarrow 1$ 
10: else if  $P^K(O|Y_n) \leq T_0$  and  $P^K(O|Y_n) > 0.5$  then
11:  $P^{K_{aux}}(O|Y_n) \leftarrow \frac{P^K(O|Y_n)+T_0-1}{2T_0-1}$ 
12: else
13:  $P^{K_{aux}}(O|Y_n) \leftarrow P^K(O|Y_n)$ 
14: end if
15: //Linearização das probabilidades de ocupação de  $G^L$ 
16: if  $P^L(O|Y_n) > T_0$  then
17:  $P^{L_{aux}}(O|Y_n) \leftarrow 1$ 
18: else if  $P^L(O|Y_n) \leq T_0$  and  $P^L(O|Y_n) > 0.5$  then
19:  $P^{L_{aux}}(O|Y_n) \leftarrow \frac{P^L(O|Y_n)+T_0-1}{2T_0-1}$ 
20: else
21:  $P^{L_{aux}}(O|Y_n) \leftarrow P^L(O|Y_n)$ 
22: end if
23: //Aplicação da regra de Bayes
24:  $P^{K,L}(O|Y_n) \leftarrow \frac{P^{K_{aux}}(O|Y_n)P^{L_{aux}}(O|Y_n)}{P^{K_{aux}}(O|Y_n)P^{L_{aux}}(O|Y_n)+(1-P^{K_{aux}}(O|Y_n))(1-P^{L_{aux}}(O|Y_n))}$ 
25: end for
    
```



**Figura 4.3:** Grelhas de ocupação local 2D construídas através do: a) Kinect; b) laser 3D\*; c) método de fusão Kinect e laser 3D\*. A imagem RGB da figura 3.2a ilustra o tipo de obstáculo que se encontra a  $100\text{ cm}$  do robô.

### Método de fusão global

Neste método admite-se que os sensores que estão a ser utilizados para fusão possuem diferentes características, nomeadamente a precisão. Obviamente que a validade deste pressuposto

**Algoritmo 4.2** Método de fusão entre a grelha de ocupação local 2D resultante da fusão do Kinect e do laser 3D\* com a grelha construída a partir dos sonares (fusão global).

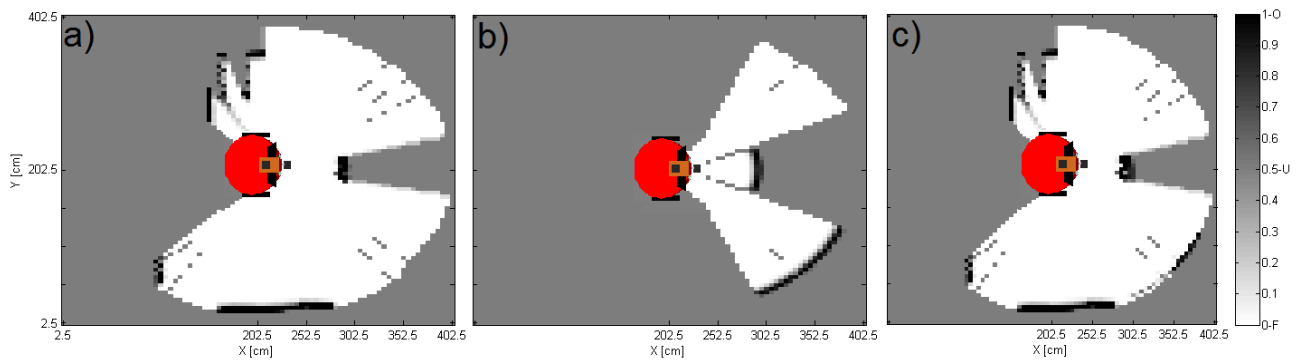
```

1: Input:  $P^U(O|Y_n)$  //Probabilidades de ocupação das células de  $G^U$ 
2:  $P^{K,L}(O|Y_n)$  //Probabilidades de ocupação das células de  $G^{K,L}$ 
3:  $T_1$  //threshold
4: Output:  $P^{K,L,U}(O|Y_n)$  //Probabilidades de ocupação das células de  $G^{K,L,U}$ 
5:
6: for all células  $C$  do
7: //Determinação da maior probabilidade numa máscara definida em  $G^U$  (1ª etapa)
8:  $P^{U^{max}}(O|Y_n) = \max_{\substack{i-1 \leq i \leq i+1 \\ j-1 \leq j \leq j+1}} P^U(O_{i,j}|Y_n)$ 
9: //Determinação das probabilidades associadas a  $G^{U,K,L}$  (2ª etapa)
10: if  $P^{U^{max}}(O|Y_n) > T_1$  then
11: //Aplicação da regra de Bayes
12:  $P^{K,L,U}(O|Y_n) \leftarrow \frac{P^{K,L}(O|Y_n)P^U(O|Y_n)}{P^{K,L}(O|Y_n)P^U(O|Y_n) + (1-P^{K,L}(O|Y_n))(1-P^U(O|Y_n))}$ 
13: else
14: //As medidas dos sensores menos precisos (sonares) são descartadas
15:  $P^{K,L,U}(O|Y_n) \leftarrow P^{K,L}(O|Y_n)$ 
16: end if
17: end for

```

é assegurada, uma vez que o Kinect e o laser 3D\* proporcionam dados mais precisos acerca da distância aos obstáculos em comparação com os sonares. Para a realização deste método, a grelha resultante do processo de fusão do Kinect com o laser 3D\* ( $G^{K,L}$ ) e a grelha dos sonares ( $G^U$ ) são criadas individualmente. A sua implementação está dividida em duas etapas, onde o algoritmo 4.2 especifica a sequência de instruções.

Na figura 4.4 demonstra-se um exemplo da fusão entre a grelha resultante deste procedimento de fusão, Neste caso, o sonar central reforçou as probabilidades de ocupação de algumas das células ocupadas pelo obstáculo detectado.



**Figura 4.4:** Grelhas de ocupação local 2D construídas através do: a) método de fusão Kinect e laser 3D\*; b) sonares; c) método de fusão global. A imagem RGB da figura 3.2a ilustra o tipo de obstáculo a 100 cm do robô.



## 4.2 Módulo de actualização temporal da grelha local

Esta secção explica os diferentes passos para actualização temporal da grelha de ocupação local 2D. Esta actualização tem em conta as novas medidas sensoriais que vão surgindo ao longo do tempo e do deslocamento do robô. Para cada célula  $C$  da grelha de ocupação local 2D obtida no instante  $t$  ( $G^t$ ) está associada uma probabilidade. A probabilidade dessa célula estar ocupada é dada por:

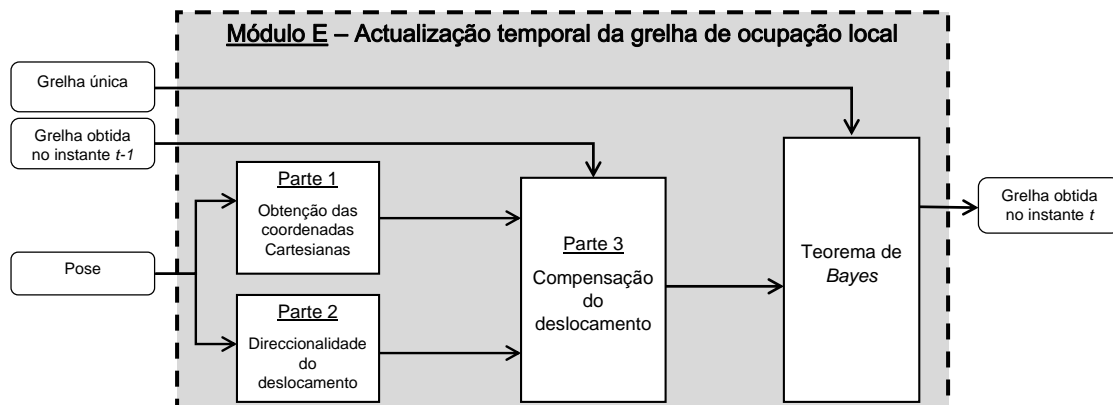
$$P^{G^t} = \frac{P^{G^{t-1}} P^{K,L,U}}{P^{G^{t-1}} P^{K,L,U} + (1 - P^{G^{t-1}}) (1 - P^{K,L,U})} \quad (4.1)$$

onde  $P^{K,L,U}$  indica as probabilidades associadas à grelha de ocupação local 2D resultante da fusão ( $G^{K,L,U}$ ) e  $P^{G^{t-1}}$  as probabilidades associadas à grelha de ocupação local 2D obtida no instante  $t - 1$  ( $G^{t-1}$ ). Inicialmente, para cada célula  $C$  da grelha  $G^{t-1}$  está associada uma probabilidade de 0.5. Deve-se frisar que, para qualquer instante de tempo  $t$ , o número de células  $C$  é constante, ou seja, o número de linhas e colunas das grelhas não se altera.

Para obter as posições  $(x_{t-1}, y_{t-1}, \theta_{t-1})$  e  $(x_t, y_t, \theta_t)$ , entre dois instantes de tempo consecutivos  $t - 1$  e  $t$ , recorreu-se ao método Hector SLAM. Este algoritmo fornece os dados estimados acerca das sucessivas posições do robô no mapa (baseado no trabalho implementado na RobChair [Perdigão, 2014]). Supondo que entre  $t$  e  $t - 1$  o robô não se deslocou, (4.1) é aplicável directamente. Logo, para cada célula da grelha  $G^{t-1}$  existe uma correspondência espacial com a célula da grelha  $G^t$ .

No caso em que o robô apenas rodou sobre ele próprio, (4.1) também é aplicável. Considera-se apenas que as células abrangidas pelo campo de visão do sensor foram alteradas durante a construção das grelhas de ocupação 2D. Como a grelha apenas se desloca com movimentos de translacção do robô, a ocorrência de erros de discretização é eliminada ([Weiss et al., 2007]).

Finalmente, o módulo da figura 4.5 é apresentado com o propósito de solucionar o problema da compensação da grelha de ocupação local 2D em relação aos movimentos de translacção do robô.

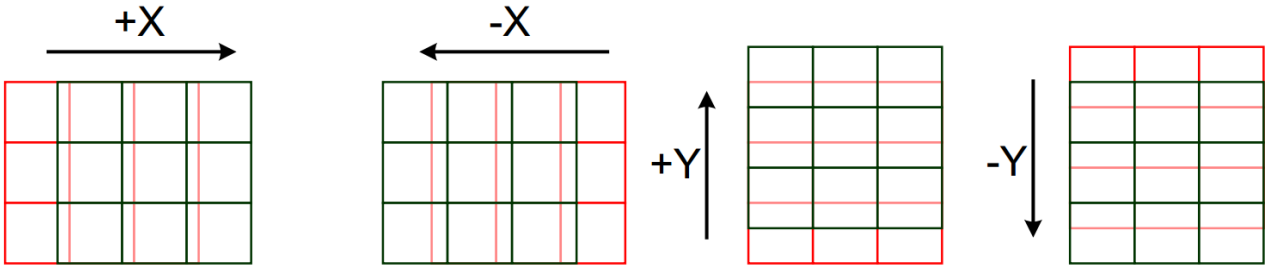


**Figura 4.5:** Estrutura interna do módulo responsável pela actualização temporal da grelha de ocupação local 2D.

A grelha obtida no instante  $t - 1$  é modificada ( $G_{t-1}^{mod}$ ) para tornar a aplicação de (4.1) válida. Nessa equação, as probabilidades de  $P^{G_{t-1}}$  são substituídas pelas probabilidades  $P^{G_{t-1}^{mod}}$ . A obtenção das probabilidades associadas à grelha  $G_{t-1}^{mod}$  é descrita em três partes: decisão do movimento; obtenção das coordenadas Cartesianas e compensação do movimento.

### Parte 1 - Direcionalidade do deslocamento

A primeira parte deste módulo consiste em determinar qual a direcção do deslocamento de translacção do robô entre dois instantes de tempo consecutivos. A figura 4.6 demonstra as possíveis direcções da grelha de ocupação local 2D.



**Figura 4.6:** Direcionalidade da grelha entre dois instantes de tempo consecutivos.

O deslocamento incremental na direcção  $x$  ( $dx$ ) e  $y$  ( $dy$ ) é obtido por:

$$dx = \begin{cases} +X & , se \Delta x > 0 \\ -X & , se \Delta x < 0 \end{cases} \quad dy = \begin{cases} +Y & , se \Delta y > 0 \\ -Y & , se \Delta y < 0 \end{cases} \quad (4.2)$$

onde  $\Delta x = x_t - x_{t-1}$  e  $\Delta y = y_t - y_{t-1}$ .

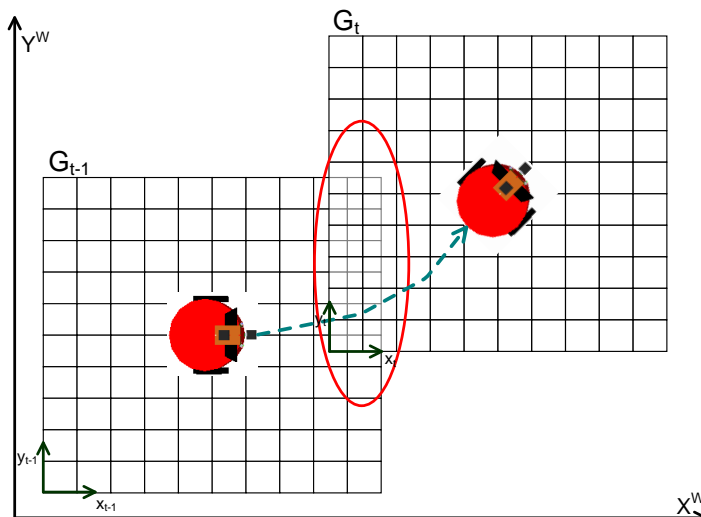
### Parte 2 - Obtenção das coordenadas Cartesianas

A figura 4.7 ilustra um exemplo de duas grelhas de ocupação local 2D obtidas no instante de tempo  $t - 1$  e  $t$ . Considere-se que  $G_{t-1}$  e  $G_t$  representam essas grelhas, respectivamente. Conhecida a direcionalidade do deslocamento da grelha, o próximo passo consiste em determinar as equações que permitem obter as coordenadas Cartesianas ( $x$ ,  $y$ ) dos centros geométricos das células entre esses instantes de tempo.

As coordenadas Cartesianas dos centros geométricos de cada uma das grelhas são dadas por:

$$\begin{cases} C^x & = x + \frac{\Delta r}{2} + \Delta r \cdot j \\ C^y & = y + \Delta r \cdot dimx - \frac{\Delta r}{2} - \Delta r \cdot i \end{cases} \quad (4.3)$$

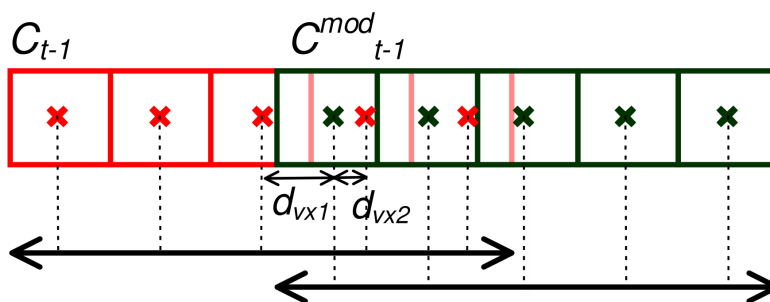
onde  $i$  e  $j$  são os índices das linhas e colunas, respectivamente,  $\Delta r$  é a resolução e  $dimx$  é o número de linhas. A aplicação desta equação para a grelha  $G_{t-1}$  com as posições  $(x_{t-1}, y_{t-1})$  resulta no conjunto de vectores  $(C_{t-1}^x, C_{t-1}^y)$ , e para  $G_t$  com as posições  $(x_t, y_t)$  no conjunto de vectores  $(C_t^x, C_t^y)$ .



**Figura 4.7:** Deslocamento da grade entre dois instantes de tempo consecutivos. A grade  $G_t$  é construída com informação da grade resultante da fusão e a grade  $G_{t-1}$ . Note-se que apenas as células que ficaram sobrepostas têm influência na actualização temporal de  $G_t$ .

### Parte 3 - Compensação do deslocamento

Considere-se o exemplo da figura 4.8 em que é necessário compensar o deslocamento da grade de ocupação local na direcção  $+X$ . Supondo que essa figura representa um vector linha dessa grade, o objectivo é determinar as probabilidades  $P^{C_{t-1}^{mod}}$ . Aplicando o método proposto, para cada célula desse vector ( $C_{t-1}^{mod}$ ), determina-se os índices do vector dos seus vizinhos mais próximos em relação ao vector  $C_{t-1}$ . Volta-se a frisar que as dimensões das grades de ocupação local têm dimensões constantes ao longo do tempo. O primeiro elemento (célula) do vector  $C_{t-1}^{mod}$  contém dois vizinhos em relação a  $C_{t-1}$ . Determinadas as suas coordenadas Cartesianas, calcula-se a distância entre a primeira célula e os vizinhos (dadas por  $d_{vx1}$  e  $d_{vx2}$ ). Após esse conhecimento, considera-se que a probabilidade associada à primeira célula de  $C_{t-1}^{mod}$  é dada pela probabilidade associada à célula do vizinho que está mais próximo. O raciocínio para as restantes células é semelhante, à excepção das células que não têm vizinhos próximos. Nesse caso, o vector é preenchido com probabilidade de 0.5. Neste exemplo, as três últimas células do vector  $C_{t-1}^{mod}$  vão assumir esse valor probabilístico.



**Figura 4.8:** Exemplo de uma linha da grade com deslocamento na direcção  $+X$ .

O método implementado no algoritmo 4.3 tem o objectivo de determinar a grelha modificada ( $G_{t-1}^{mod}$ ) a partir da grelha obtida no instante ( $G_{t-1}$ ). Dessa forma, a grelha  $G_{t-1}^{mod}$  contém informação acerca das probabilidades  $P^{G_{t-1}^{mod}}$ , compensando o deslocamento do robô. Como é possível reparar, este método funciona como um filtro de distância mínima entre os centros geométricos das células que se sobrepõem. As probabilidades  $P^{G_{t-1}^{mod}}$  das restantes células da grelha  $G_{t-1}^{mod}$  são adicionadas com valor 0.5 (estado de ocupação desconhecido). Sendo que a região quadrática (grelha) é armazenada na memória do computador com as mesmas dimensões, quando a grelha se desloca, as células na direcção oposta ao movimento são removidas. Já no caso em que não existe distância mínima, isto é, a distância aos dois vizinhos é igual, decidiu-se aplicar o operador de máximo.

Também é possível verificar, pelo algoritmo 4.3, que a operação sobre a grelha  $G_{t-1}$  compensa o deslocamento na direcção em  $X$  e, posteriormente, o deslocamento na direcção em  $Y$ . As principais funções deste algoritmo são:

- **GRELHA AUXILIAR:** função que implementa uma regra de atribuição de probabilidades de modo a compensar o deslocamento da grelha na direcção  $X$ . Esta função atribui as probabilidades de ocupação a cada célula tendo por base a menor distância entre o seu centro geométrico e o das células vizinhas.
- **GRELHA MODIFICADA:** função que implementa uma regra de atribuição de probabilidades que compensam o deslocamento da grelha na direcção  $Y$ , após a compensação na direcção  $X$ . Esta função também atribui as probabilidades de ocupação a cada célula tendo por base a menor distância entre o seu centro geométrico e o das células vizinhas. As probabilidades atribuídas, dadas por  $P^{G_{t-1}^{mod}}$ , substituem  $P^{G_{t-1}}$  em (4.1).

**Algoritmo 4.3** Método que determina as probabilidades da grelha modificada, tendo em conta o deslocamento do robô.

---

```

1: Input:  $(C_{t-1}^x, C_{t-1}^y)$  //Vector com as coordenadas Cartesianas de  $G_{t-1}$ 
2:           $(C_t^x, C_t^y)$  //Vector com as coordenadas Cartesianas de  $G_t$ 
3:           $dx, dy$  //Deslocamento incremental na direção x e y, respectivamente
4:           $P^{G_{t-1}}$  //Probabilidade de cada célula de  $G_{t-1}$  estar ocupada
5: Output:  $P^{G_{t-1}^{mod}}$  //Probabilidade de cada célula de  $G_{t-1}^{mod}$  estar ocupada
6:
7: for all  $i$  do
8:   for all  $j$  do
9:      $j_{vx1} = \lfloor \frac{C_t^x(i) - C_{t-1}^x(1)}{\Delta r} \rfloor$  //Índices coluna vizinhos mais próximos
10:     $j_{vx2} = \lceil \frac{C_t^x(i) - C_{t-1}^x(1)}{\Delta r} \rceil$ 
11:    if  $j_{vx1} \leq \text{length}(C_t^x)$  and  $j_{vx2} \leq \text{length}(C_t^x)$  then
12:       $d_{vx1} \leftarrow |C_t^x(i) - C_{t-1}^x(j_{vx1})|$  // Distância aos vizinhos mais próximos
13:       $d_{vx2} \leftarrow |C_t^x(i) - C_{t-1}^x(j_{vx2})|$ 
14:       $P^{G_{t-1}^{aux}} \leftarrow \mathbf{GRELHAUXILIAR}(d_{vx1}, d_{vx2}, P^{G_{t-1}}, dx)$  //Regra de atribuição
15:    else if  $j_{vx1} \leq \text{length}(C_t^x)$  and  $j_{vx2} > \text{length}(C_t^x)$  then
16:       $d_{vx1} \leftarrow |C_t^x(i) - C_{t-1}^x(j_{vx1})|$ 
17:       $P^{G_{t-1}^{aux}} \leftarrow \mathbf{GRELHAUXILIAR}(d_{vx1}, -, P^{G_{t-1}}, dx)$ 
18:    else
19:       $P^{G_{t-1}^{aux}} \leftarrow 0.5$ 
20:    end if
21:     $i_{vy1} = \lfloor \frac{C_t^y(i) - C_{t-1}^y(\text{length}(C_{t-1}^y))}{\Delta r} \rfloor$  //Índices linha vizinhos mais próximos
22:     $i_{vy2} = \lceil \frac{C_t^y(i) - C_{t-1}^y(\text{length}(C_{t-1}^y))}{\Delta r} \rceil$ 
23:    if  $i_{vy1} \leq \text{length}(C_t^y)$  and  $i_{vy2} \leq \text{length}(C_t^y)$  then
24:       $d_{vy1} \leftarrow |C_t^y(i) - C_{t-1}^y(i_{vy1})|$  //Distância até aos vizinhos mais próximos
25:       $d_{vy2} \leftarrow |C_t^y(i) - C_{t-1}^y(i_{vy2})|$ 
26:       $P^{G_{t-1}^{mod}} \leftarrow \mathbf{GRELHAMODIFICADA}(d_{vy1}, d_{vy2}, P^{G_{t-1}^{aux}}, dy)$  //Regra
27:    else if  $i_{vy1} \leq \text{length}(C_t^y)$  and  $i_{vy2} > \text{length}(C_t^y)$  then
28:       $d_{vy1} \leftarrow |C_t^y(i) - C_{t-1}^y(i_{vy1})|$ 
29:       $P^{G_{t-1}^{mod}} \leftarrow \mathbf{GRELHAMODIFICADA}(d_{vy1}, -, P^{G_{t-1}^{aux}}, dy)$ 
30:    else
31:       $P^{G_{t-1}^{mod}} \leftarrow 0.5$ 
32:    end if
33:   end for
34: end for

```

---

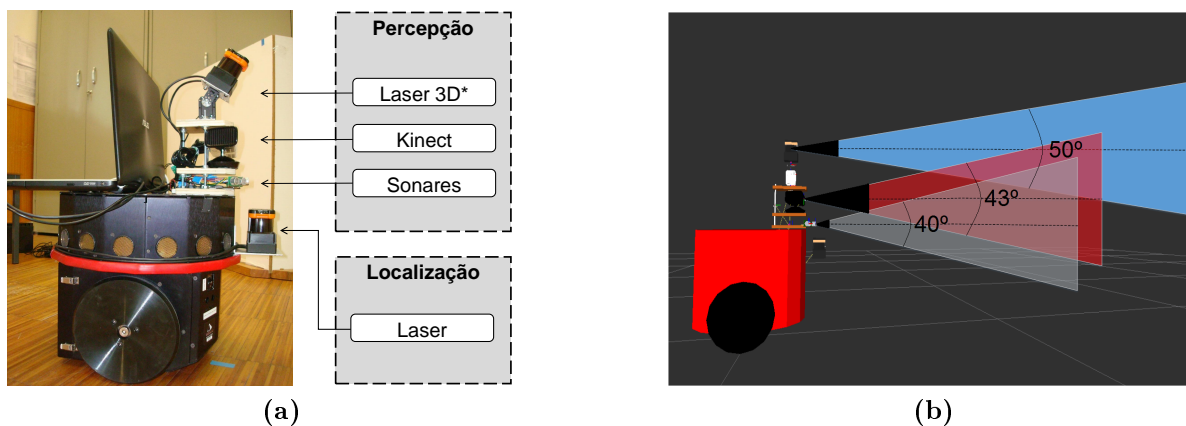
# Capítulo 5

## Implementações

Neste capítulo descrevem-se as arquitecturas de *hardware* e *software* necessárias para a concretização do sistema de percepção espaço-temporal local apresentado. As técnicas e algoritmos propostos foram implementados em *software* ROS e Matlab.

### 5.1 Arquitectura de *hardware*

Para que o sistema de percepção proposto seja testado em cenários reais *indoor*, realizando os procedimentos de aquisição de dados e construção das grelhas de ocupação local 2D, utilizou-se um sistema robótico móvel Nomad Scout equipado com um anel de três sonares, um Kinect e um laser 3D\*, como demonstra a figura 5.1a. A configuração utilizada permite que o robô possua três camadas de percepção sensorial que abrangem um campo de visão dianteiro (ver figura 5.1b).



**Figura 5.1:** (a) Nomad Scout utilizado para a implementação prática do trabalho. (b) Modelo do Nomad Scout para visualização no Rviz e respectivo campo de visão vertical de cada um dos sensores.

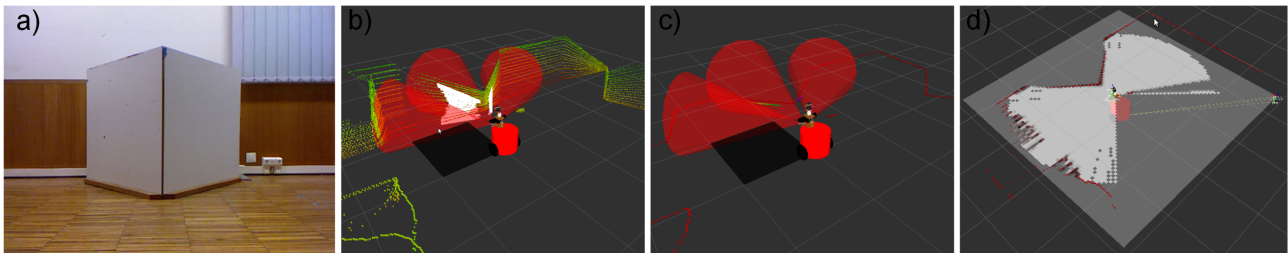
Os sonares estão conectados a um microcontrolador Arduino que está responsável por activá-los sequencialmente para evitar problemas de multi-ecos. O Arduino e as duas restantes modalidades sensoriais são ligadas a um computador portátil em comunicação série por porta USB. Uma das principais preocupações foi integrar devidamente todos os componentes de *hardware*, nomeadamente, as modalidades sensoriais responsáveis por obter a percepção sobre o ambiente.

A leitura dos dados da odometria do robô seria uma escolha aceitável para obter a sua localização. No entanto, decidiu-se utilizar mais um laser, situado na zona inferior, cujos dados

permitem obter a localização recorrendo ao método de SLAM implementado na RobChair (com base no trabalho realizado em [Perdigão, 2014]).

## 5.2 Arquitectura de *software*

Os três primeiros módulos da estrutura proposta neste trabalho (ver figura 1.3) foram desenvolvidos utilizando *software* ROS (C++) e os outros dois utilizando o *software* Matlab.

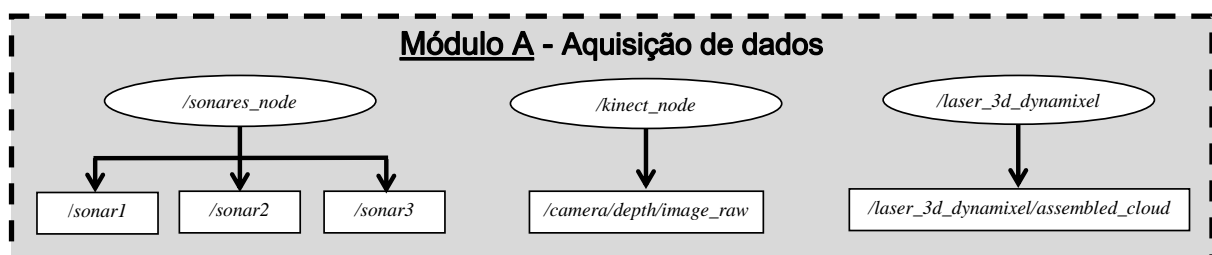


**Figura 5.2:** (a) Imagem RGB do cenário obtida pelo Kinect. Visualização em Rviz: (b) dos cones de três sonares, da nuvem de pontos (cor branco) do Kinect e da nuvem de pontos do laser 3D\*. (c) dos cones de três sonares, dos pontos 2D (cor vermelho) do laser 3D\* e dos restantes pontos 2D do Kinect. (d) Grelha de ocupação local 2D construída a partir dos pontos 2D (respectivos a um varrimento 2D) obtidos pelo laser 3D\*.

### 5.2.1 Implementação em ROS

#### Aquisição de dados

O *software* responsável pela aquisição de dados foi desenvolvido nos nós `/sonares_node`, `/kinect_node` e `/laser_3d_dynamixel` (ver figura 5.3).



**Figura 5.3:** Nós e tópicos associados ao módulo responsável pela aquisição de dados.

O primeiro nó está responsável por devolver três medidas de profundidade (à frequência de  $11Hz$ ) de cada um dos sonares através dos tópicos `/sonar1`, `/sonar2` e `/sonar3`. O segundo nó (disponível pela comunidade *open-source* do ROS) devolve uma imagem de profundidade do Kinect (convertível numa nuvem de pontos, à frequência de  $30Hz$ ) através do tópico `/camera/depth/image_raw`. O terceiro nó devolve uma nuvem de pontos do laser 3D\* (à frequência

de 2Hz) através do tópico `/laser_3d_dynamixel/assembled_cloud`. Na figura 5.2b visualiza-se esse conjunto de dados sensoriais obtidos a partir do ambiente da figura 5.2a.

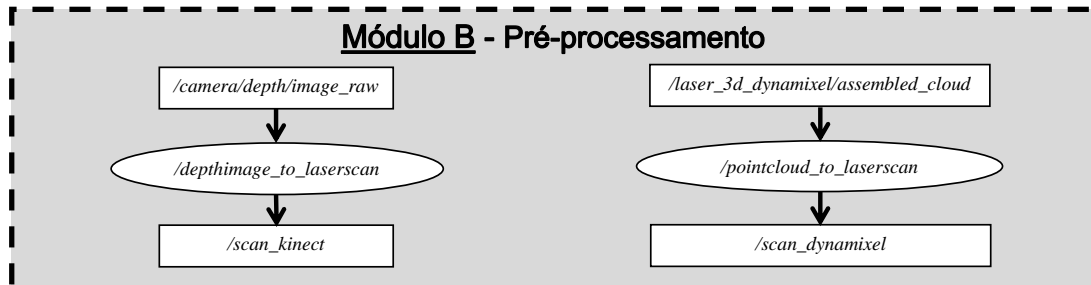
Neste módulo, as implementações desenvolvidas ao nível de *software* fazem com que exista um tipo de ligação viável entre as camadas de *hardware* e *software*, conhecido como *drivers*.

### Pré-processamento

Este pré-processamento tem como objectivo principal a projecção dos pontos do plano 3D para o plano 2D, resultando assim num varrimento 2D. O *software* responsável pela projecção dos dados para 2D está disponível, pela comunidade *open-source*, para o ROS. Desse modo, decidiu-se utilizar o *software* disponível e adaptá-lo, por exemplo, de forma a que os pontos que incidiram na superfície do chão fossem removidos.

O *software* responsável pela projecção dos dados do Kinect para o plano 2D foi desenvolvido no nó `/depthimage_to_laserscan`. Esse nó recebe a imagem de profundidade devolvida pelo Kinect através do tópico `/camera/depth/image_raw` e devolve um varrimento 2D para o tópico `/scan_kinect`.

No entanto, o *software* responsável pela projecção dos dados do laser 3D\* para o plano 2D foi desenvolvido no nó `/pointcloud_to_laserscan`. Esse nó recebe uma nuvem de pontos 3D através do tópico `/laser_3d_dynamixel/assembled_cloud` e devolve um varrimento 2D no tópico `/scan_dynamixel` (ver figura 5.4).



**Figura 5.4:** Nós e tópicos associados ao módulo responsável pelo pré-processamento.

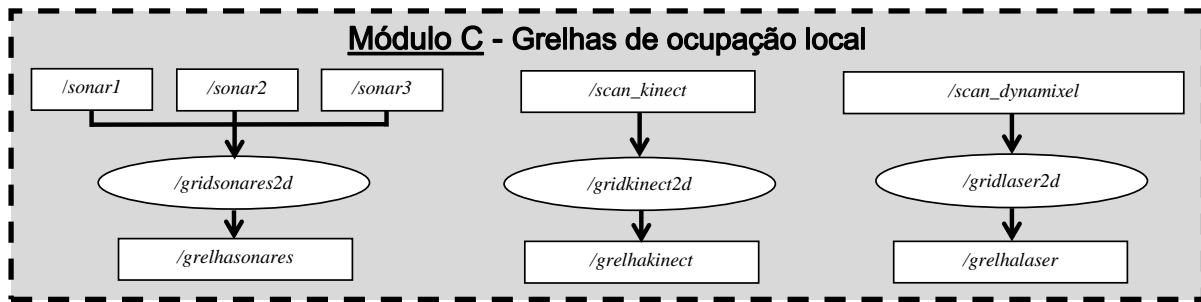
Estes *softwares* tiveram de ser adaptados aos objectivos propostos neste trabalho. A figura 5.2c demonstra a projecção no plano 2D para os pontos da figura 5.2b. Os pontos 2D têm como referência o sistema de coordenadas do respectivo sensor.

### Grelhas de ocupação local

A implementação, em *software*, do módulo responsável pela construção em tempo real das grelhas de ocupação local 2D dos sonares, do Kinect e do laser 3D\* é uma das principais contribuições deste capítulo.

Este *software* está desenvolvido nos nós `/gridsonares2D`, `/gridkinect2D` e `/gridlaser2D`. Esses nós recebem os dados de profundidade dos três sonares pelos tópicos `/sonar1`, `/sonar2`





**Figura 5.5:** Nós e tópicos associados ao módulo responsável pela construção das grelhas de ocupação local 2D.

e */sonar3*, o varrimento 2D do Kinect pelo tópico */scan\_kinect* e o varrimento 2D do laser 3D\* pelo tópico */scan\_dynamixel* (ver figura 5.5). O *software* desenvolvido neste módulo disponibiliza três grelhas de ocupação nos tópicos */grelhasonares*, */grelhakinect* e */grelhalaser*, respectivamente. A figura 5.2d exibe uma grelha de ocupação local 2D construída a partir dos dados obtidos pela projecção 2D do sensor laser 3D\*.

### 5.2.2 Implementação em Matlab

Os tópicos */grelhasonares*, */grelhakinect* e */grelhalaser* definem-se como estruturas de dados matriciais que contêm informação sobre a probabilidade de ocupação de cada uma das células das grelhas construídas pelos sonares, pelo Kinect e pelo laser 3D\*, respectivamente. Estas matrizes foram armazenadas num ficheiro para, posteriormente, se proceder à implementação dos módulos de fusão e actualização temporal. Dessa forma, o método proposto para fusão das grelhas de ocupação local 2D assim como o algoritmo responsável pela actualização temporal da grelha resultante do processo de fusão foram implementados *offline* em *software* Matlab.

# Capítulo 6

## Resultados experimentais

Neste capítulo descreve-se um conjunto de experiências que avaliam as partes fundamentais do desempenho do sistema proposto, de maneira a testar e validar o seu comportamento perante várias situações que possam surgir no ambiente. Na secção 6.1 o sistema é confrontado com um conjunto de situações de falha expectável, que advém dos problemas relacionados com as questões intrínsecas e limitações dos sensores. Na secção 6.2 confronta-se o sistema em alguns cenários pertinentes que possam surgir durante a navegação do robô móvel, com o objectivo de avaliar a robustez do sistema perante situações que têm em vista a sua aplicação real. A secção 6.3 refere-se à validação do sistema de percepção na questão da integração temporal.

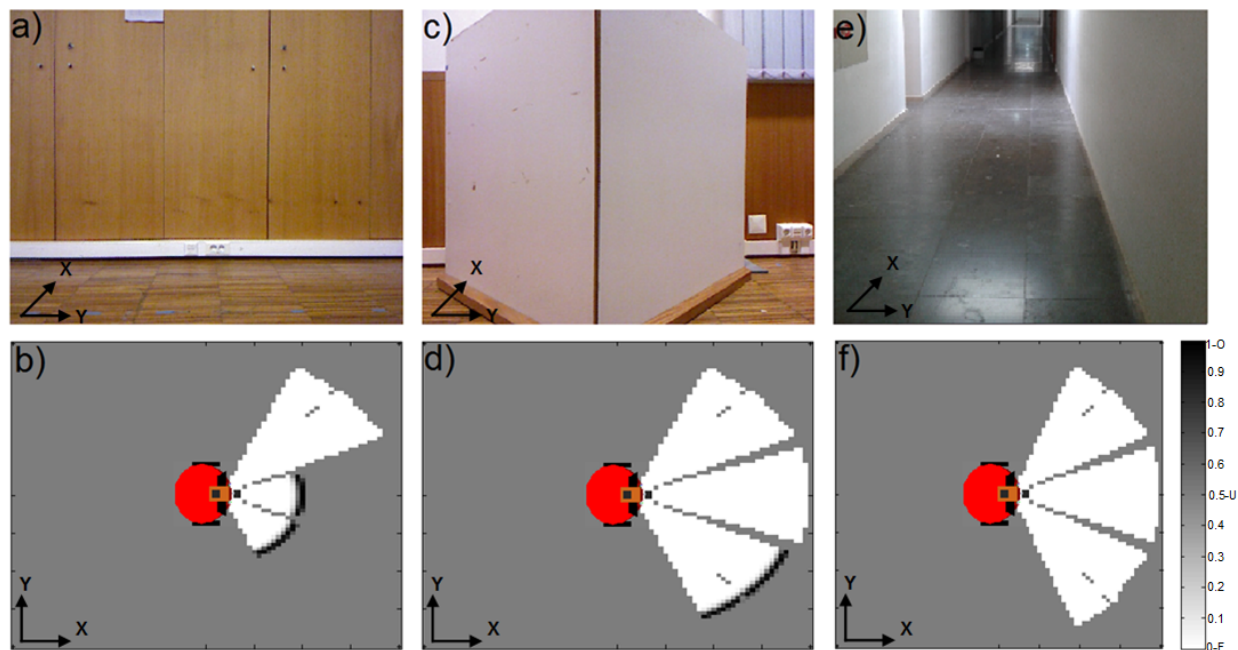
### 6.1 Análise de resultados

Ao longo deste documento ficou claro que a modelação do ambiente, nomeadamente em grelhas de ocupação local 2D, está totalmente dependente do funcionamento dos sensores. Se os sensores por algum motivo falharem, esses problemas vão manifestar-se no processo de construção das grelhas, surgindo com fraca interpretação acerca do ambiente circundante. As questões intrínsecas ou limitações dos sensores são a maior causa desses problemas.

Considere-se que o robô não está em movimento e, que num determinado instante de tempo, três grelhas de ocupação local 2D são criadas individualmente a partir de informação dos sonares, Kinect ou laser 3D\*. As figuras 6.1, 6.2 e 6.3 mostram as grelhas construídas por cada um dos sensores enunciados. Como se pode observar, as grelhas demonstram-se pouco esclarecedoras em situações onde os sensores falharam, à partida por razões óbvias.

#### **Problemas na grelha utilizando os sonares**

Na figura 6.1 observa-se o problema das reflexões especulares dos sonares. Suponha-se que o centro geométrico do robô está posicionado a 100 *cm* da parede da figura 6.1a. Devido à não incidência directa de um dos sonares, a sua medida é superior à distância real. Por consequência, na figura 6.1b observa-se que todas as células abrangidas pelo campo de visão desse sensor estão indicadas como regiões livres. O mesmo acontece quando o robô se encontra num corredor estreito, como o da figura 6.1e. Neste caso, os dois sonares laterais também devolvem medidas erradas acerca da distância real, fazendo com que a grelha da figura 6.1f também indique essas regiões como livres. Um outro exemplo de mais problemas de reflexões especulares ocorre em esquinas. Neste tipo de cenário, como o da figura 6.1c, apesar do sonar



**Figura 6.1:** a) e b) são a imagem RGB da cena e correspondente grelha local 2D construída pelos sonares, respectivamente. Assim como para os pares c), d) e e), f). Esta figura retrata o problema das reflexões especulares dos sonares.

central estar a incidir directamente para o obstáculo, a uma distância de  $50\text{ cm}$ , ele devolve uma distância bastante superior à real. A figura 6.1d mostra a grelha construída.

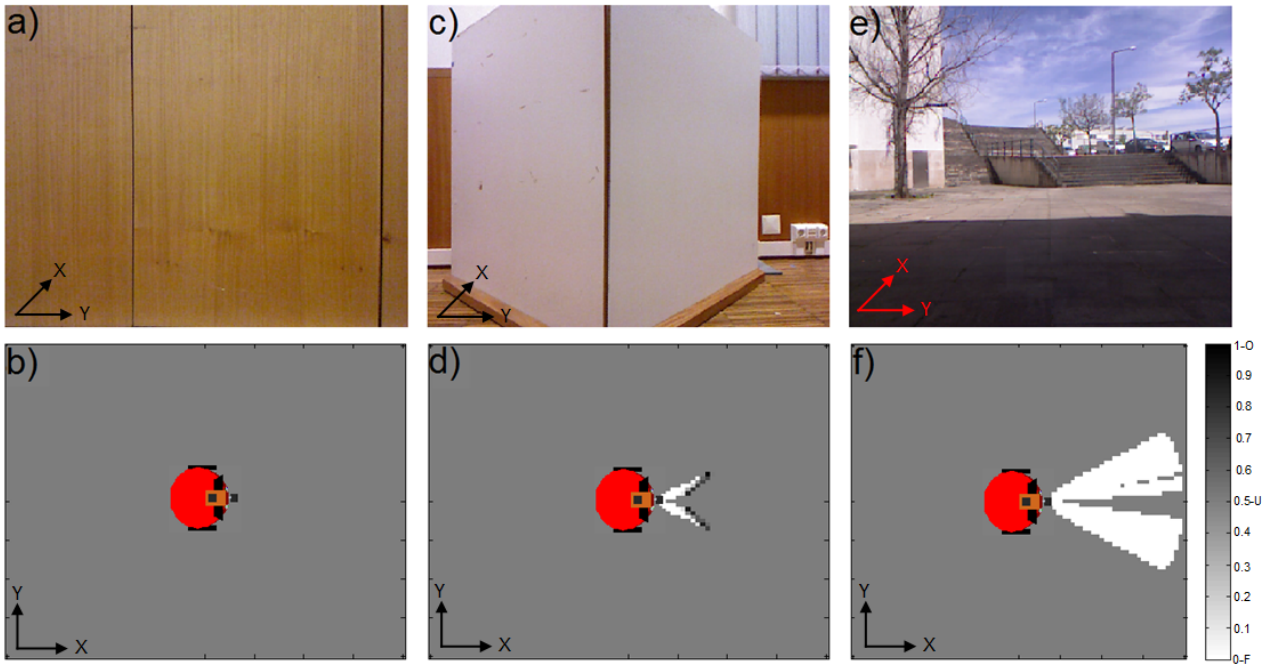
### Problemas na grelha utilizando o Kinect

Na figura 6.2 podem-se constatar os problemas ocorridos durante a modelação das grelhas utilizando o sensor Kinect. Considere-se que o centro geométrico do robô é posicionado a  $50\text{ cm}$  de uma parede (figura 6.2a) ou de uma esquina (figura 6.2c). Em ambas as situações este sensor perde as suas capacidades de observação, sendo esta uma das suas maiores limitações. Em ambientes *indoor*, não ter a capacidade de observar obstáculos muito próximos (aproximadamente, inferiores a  $60\text{ cm}$ ) é inaceitável. Este problema implica que um elevado número de células das grelhas, das figuras 6.2b e 6.2d, abrangidas pelo campo de visão do Kinect sejam identificadas como regiões desconhecidas. A figura 6.2b é um caso extremo, onde todo o campo de visão foi afectado.

Devido a questões intrínsecas deste sensor, as superfícies em vidro não são detectáveis. Na presença de tais superfícies (como da figura 6.2e) a grelha não apresenta qualquer indício da sua presença (ver figura 6.2f). Note-se que na figura 6.2f, algumas células dentro do campo de visão do sensor foram identificadas como regiões desconhecidas, o que se deve à limitação da distância máxima (aproximadamente  $6\text{ m}$ ) atingível pelo sensor.

### Problemas na grelha utilizando o laser 3D\*

Na figura 6.3b e 6.3c comprova-se a incapacidade do laser 3D\* detectar as superfícies em vidro como da figura 6.3a. Nesta situação, com o robô posicionado a  $50\text{ cm}$  de distância dessa



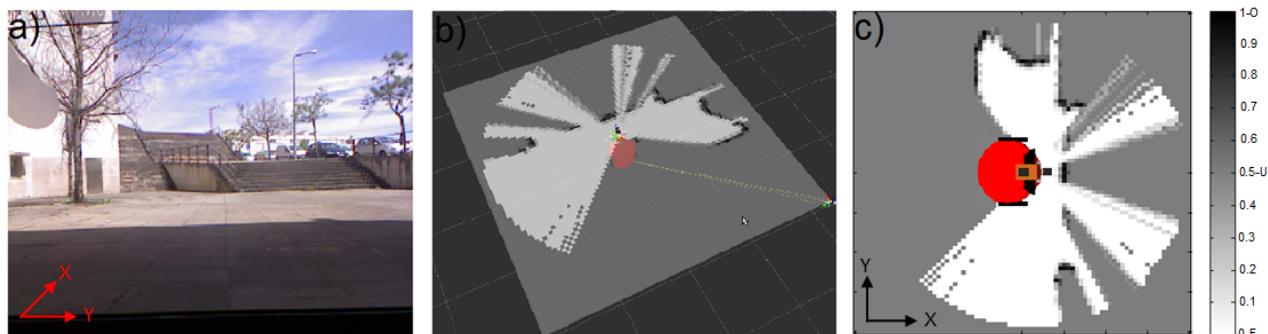
**Figura 6.2:** a) e b) são a imagem RGB da cena e correspondente grelha local 2D construída pelo Kinect, respectivamente. Assim como para os pares c), d) e e), f). O problema dos obstáculos muito próximos e das superfícies em vidro são ilustrados nesta figura.

superfície, verifica-se que a maior parte dos feixes não detectam o obstáculo. No entanto, observa-se que alguns deles foram capazes de indiciar a existência do obstáculo, fazendo com que algumas das células estejam indicadas como regiões ocupadas. Não esquecendo que o campo de visão do laser 3D\* é tridimensional, a explicação para isto deve-se a alguma parte do vidro que contenha alguma poeira ou outro tipo de sujidade que foi detectável. A outra explicação deve-se à detecção do autocolante fixado (canto superior esquerdo da figura 6.3a).

No anexo A é apresentada uma análise quantitativa acerca das medidas devolvidas pelo sonar e laser.

## 6.2 Cenários complexos

Para testar e validar as capacidades de percepção do sistema proposto neste trabalho, nomeadamente no reconhecimento de obstáculos em grelhas de ocupação local 2D, o robô é confrontado com um conjunto de cenários que tem em vista a aplicação prática do sistema de percepção local em ambientes reais. Na figura 6.4 considera-se que o robô está na presença de um vaso que se encontra a 100 centímetros de distância do seu centro geométrico. Nesta situação, devido às folhas, verifica-se que as grelhas de ocupação local 2D construídas pelo Kinect (figura 6.4b) e laser 3D\* (figura 6.4c) apresentam informação duvidosa acerca da localização do obstáculo. Na integração dessas grelhas com a grelha dos sonares (figura 6.4d) a dúvida deixa de persistir, resultando na grelha da figura 6.4e. Na figura 6.5 considera-se que o robô está na presença de uma pessoa que se encontra a 50 centímetros de distância do seu centro ge-



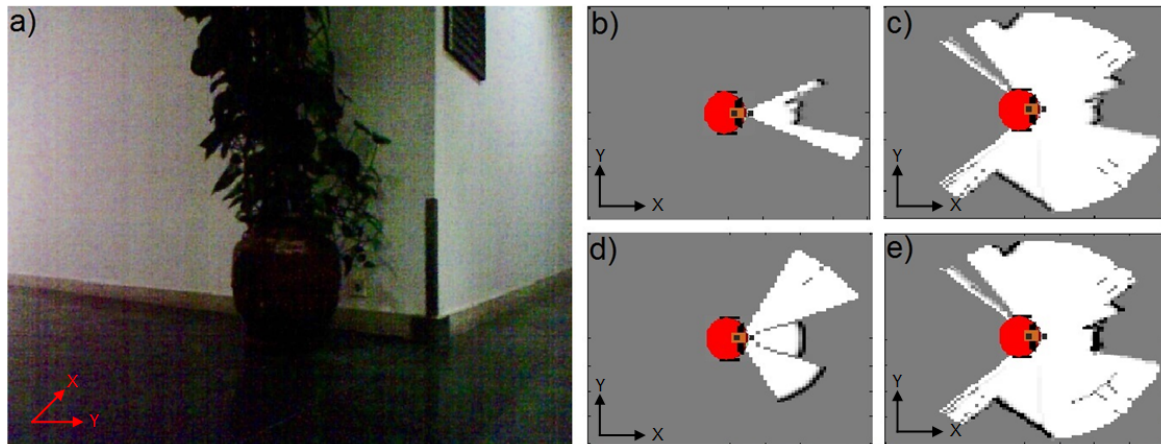
**Figura 6.3:** a) é a imagem RGB da cena. b) e c) correspondem à grelha local 2D construída pelo laser 3D\*. Em b) a grelha é visualizada em Rviz. É ilustrado o problema das superfícies em vidro nesta figura.

ométrico. Nesta situação, observa-se que a grelha de ocupação local 2D construída pelo Kinect (figura 6.5b) não contém qualquer tipo de informação acerca da ocupação das células devido à limitação do sensor. Por outro lado, neste tipo de situações, as grelhas construídas pelo laser 3D\* (figura 6.5c) e pelos sonares (6.5d) demonstram-se robustas na detecção. A integração das grelhas resulta na grelha da figura 6.5e, onde a localização do obstáculo aparece evidenciada. Na figura 6.6 considera-se que o robô está na presença de uma passagem por uma porta que se encontra a 100 centímetros de distância do seu centro geométrico. Neste cenário, observa-se que a grelha de ocupação local 2D construída pelos sonares (figura 6.6d) torna-se incapaz de detectar passagens estreitas. Por outro lado, as grelhas construídas pelo Kinect (figura 6.6b) e pelo laser 3D\* (6.6c) demonstram-se eficazes na detecção deste tipo de travessias. A integração das grelhas resulta na grelha da figura 6.6e, onde a informação de ocupação imposta pelos sonares já não impõe restrições na navegação do robô em cenários semelhantes a este.

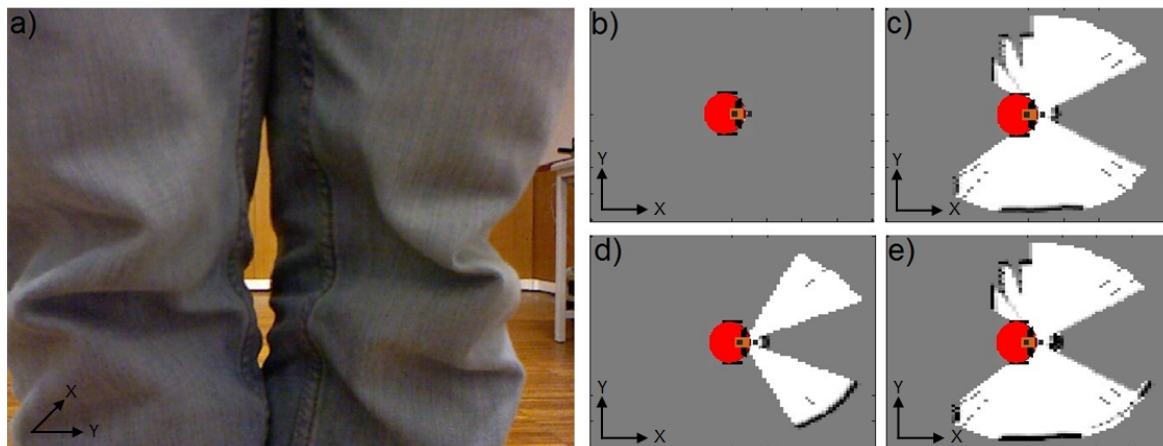
A figura 6.7 assemelha-se à figura 6.6. Contudo, considera-se que o robô está na presença de uma passagem por uma porta que se encontra a 400 centímetros de distância do seu centro geométrico. O problema ocorre quando o sonar central, de facto, detectou um obstáculo que realmente lhe impõe restrições na navegação (ver obstáculo em 6.7a que se encontra na superfície do chão). Neste cenário observa-se que a grelha de ocupação local 2D construída pelos sonares (figura 6.7d) teve a capacidade de observar esse obstáculo. Por outro lado, as grelhas construídas pelo Kinect (figura 6.7b) e pelo laser 3D\* (figura 6.7c) não detectaram qualquer indício da presença dele. Este é um caso de estudo desafiante na robótica móvel, por exemplo, em situações como, na presença de um tapete na superfície do chão pode fazer com que o robô móvel perca o controlo e/ou a estabilidade necessária para continuar a navegar normalmente.

Durante o desenvolvimento deste trabalho e acerca dos resultados experimentais obtidos entre as grelhas de ocupação local 2D construídas, indica-se um conjunto de vantagens e desvantagens da utilização dos sonares, do Kinect e do laser 3D\*.

As vantagens do Kinect sobre o laser 3D\* são: o custo de aquisição do Kinect é bastante inferior e a frequência de operação é maior. As desvantagens do Kinect sobre o laser 3D\* são: para



**Figura 6.4:** Na imagem a) observa-se o cenário, obtido pela imagem RGB do Kinect. As grelhas b), c), d) e e) são construídas pelo Kinect, pelo laser 3D\*, pelos sonares e pelo processo de fusão, respectivamente.



**Figura 6.5:** Na imagem a) observa-se o cenário, obtido pela imagem RGB do Kinect. As grelhas b), c), d) e e) são construídas pelo Kinect, pelo laser 3D\*, pelos sonares e pelo processo de fusão, respectivamente.

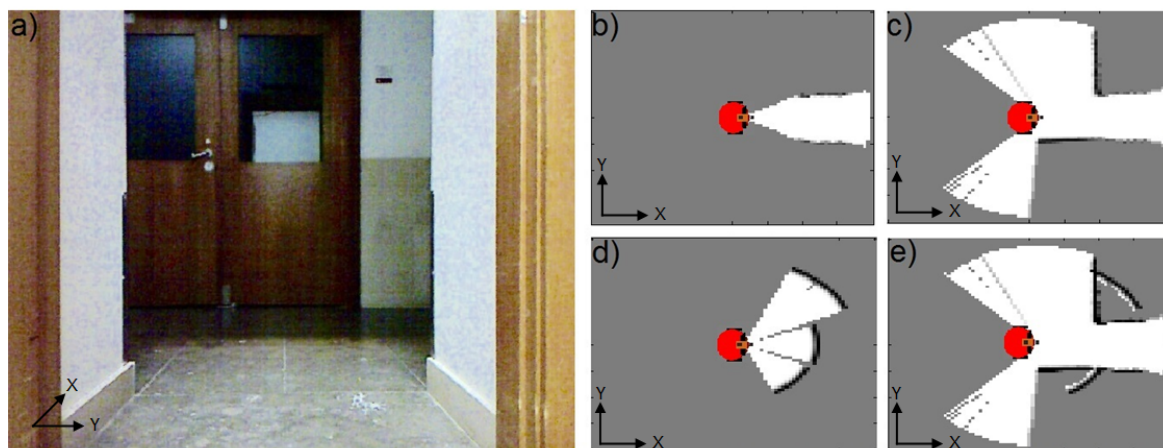
objectos relativamente próximos do Kinect, perdem-se as capacidades de detecção/observação do obstáculo e o campo de visão é reduzido.

As vantagens dos sonares sobre o Kinect e laser 3D\* são: a capacidade de detectar mesmo na presença de obstáculos muito próximos (até 3 cm) e conseguem captar informação na presença de portas/janelas/paredes de vidro. As desvantagens dos sonares sobre o Kinect e laser 3D\* são: devido ao ângulo de abertura do seu campo de visão tornam-se incapazes de reconhecer passagens estreitas e os problemas de reflexões especulares, por exemplo, em esquinas e cantos.

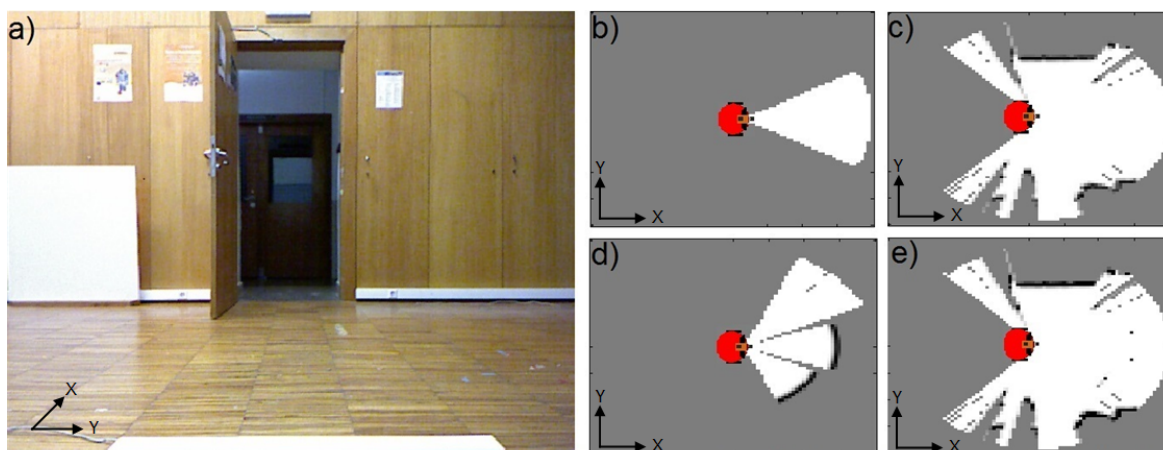
## 6.3 Cenário com integração temporal

Por forma a validar o método de actualização temporal proposto, procedeu-se à realização de experiências num cenário fechado, construído no laboratório de mecatrónica do ISR (figura





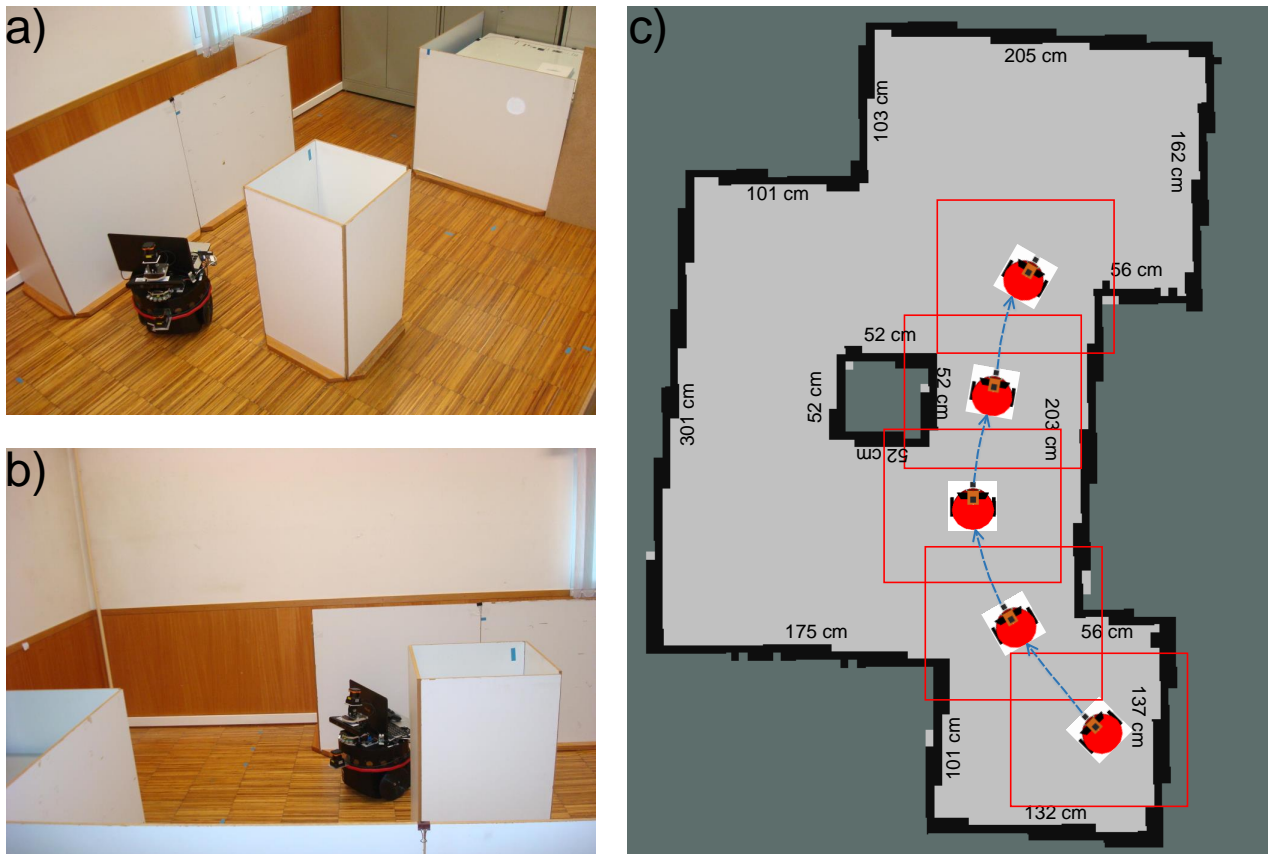
**Figura 6.6:** Na imagem a) observa-se o cenário, obtido pela imagem RGB do Kinect. As grelhas b), c), d) e e) são construídas pelo Kinect, pelo laser 3D\*, pelos sonares e pelo processo de fusão, respectivamente.



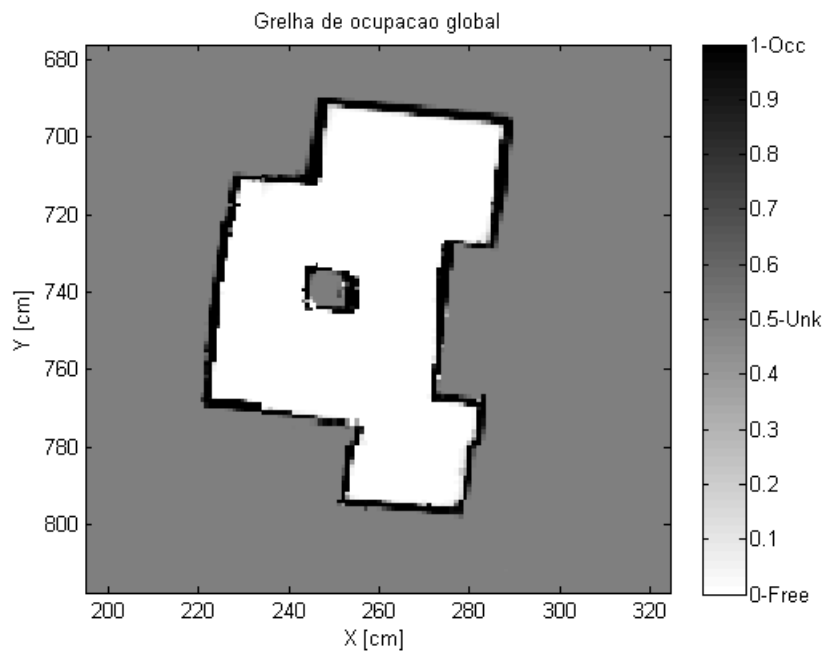
**Figura 6.7:** Na imagem a) observa-se o cenário, obtido pela imagem RGB do Kinect. As grelhas b), c), d) e e) são construídas pelo Kinect, pelo laser 3D\*, pelos sonares e pelo processo de fusão, respectivamente.

6.8a e 6.8b). Através da utilização do *hardware* e do *software* (em ROS) descrito na secção 5, iniciou-se a recolha de um conjunto de 100 grelhas de ocupação local 2D obtidas nesse cenário. Para cada uma das grelhas está associada uma estimativa da localização (*pose*) do robô móvel.

Com uma acumulação sucessiva das 100 grelhas locais é possível visualizar o processo de construção de um modelo global do ambiente (ver figura ilustrativa 6.8c). Este princípio, foi aplicado para validar qualitativamente o algoritmo de actualização temporal da grelha de ocupação local 2D, em comparação com uma grelha global criada pelo método Hector SLAM (usada como *ground truth*). Para este propósito, utilizou-se apenas as grelhas de ocupação local 2D criadas a partir da informação do laser 3D\*, com os resultados apresentados na figura 6.9. Em contrapartida, na figura 6.11 apresentam-se os resultados obtidos a partir das grelhas de ocupação local 2D criadas a partir da informação do Kinect, do laser 3D\* e da fusão. Para obter estes resultados realizaram-se novas experiências no cenário da figura 6.10.

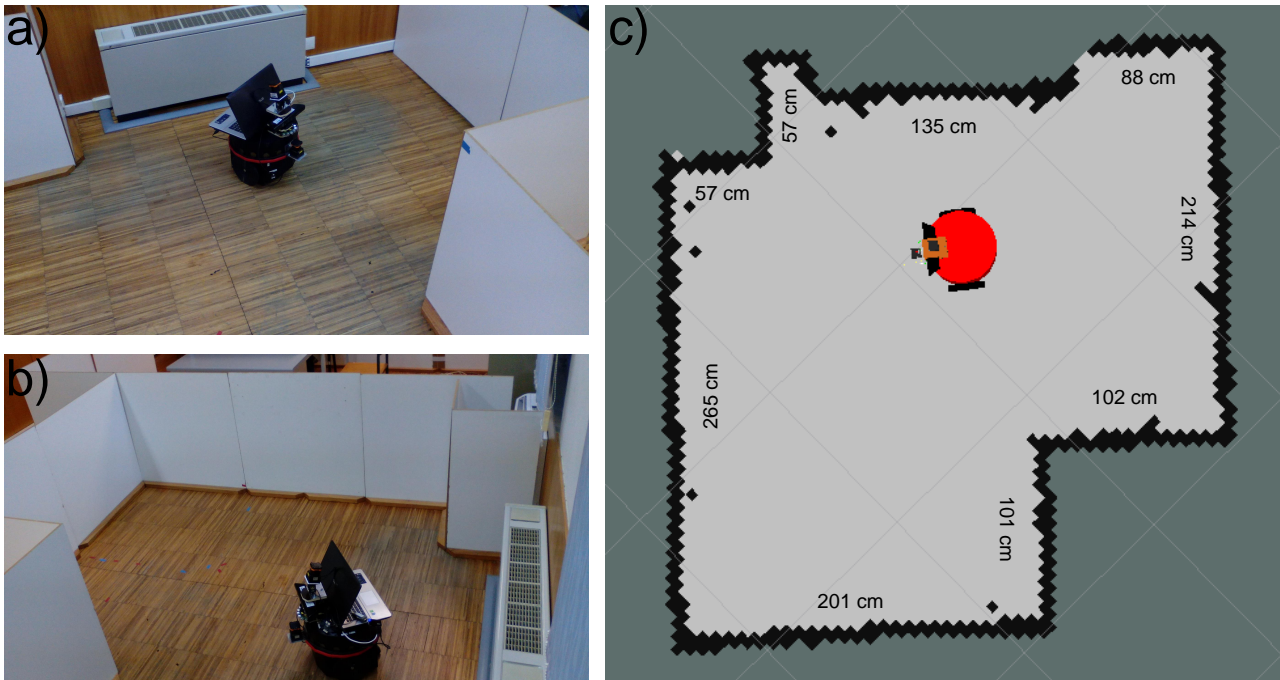


**Figura 6.8:** a) e b) cenário no laboratório de mecatrónica do ISR. c) sucessão de grelhas locais ao longo de um percurso (ilustrativo). O mapa global obtido por Hector SLAM é utilizado como *ground truth*.

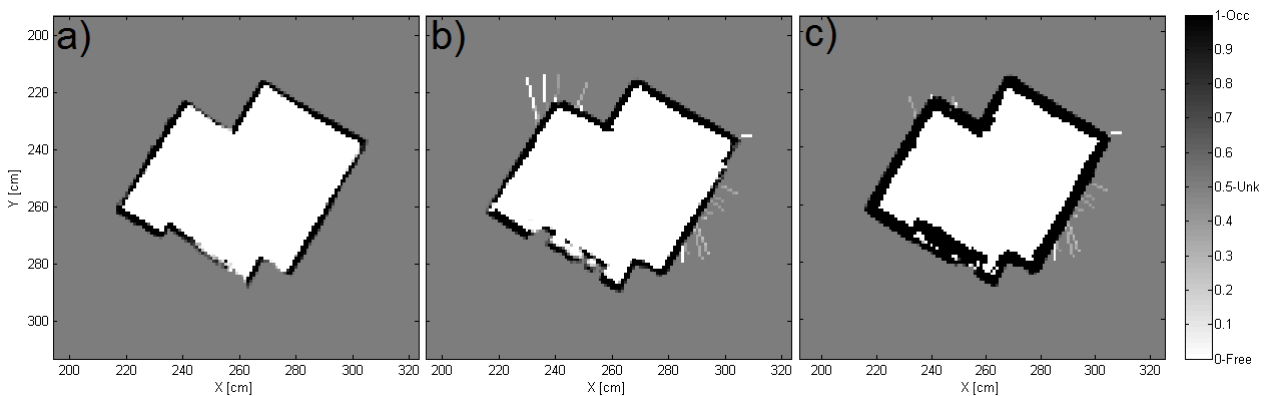


**Figura 6.9:** Resultado do algoritmo proposto para actualização temporal da grelha de ocupação local 2D (implementado em ambiente Matlab). Em comparação com o resultado obtido na figura 6.8c, verificam-se resultados semelhantes e que validam o algoritmo proposto.





**Figura 6.10:** a) e b) cenário no laboratório de mecatrónica do ISR. c) Mapa global obtido por Hector SLAM é utilizado como *ground truth*.



**Figura 6.11:** Resultado do algoritmo proposto para actualização temporal da grelha de ocupação local 2D (implementado em ambiente Matlab). Em comparação com o resultado obtido na figura 6.10c, verificam-se resultados semelhantes e que validam o algoritmo proposto. Grelha de ocupação global obtida a partir das grelhas locais do: a) Kinect, b) laser 3D\* e c) fusão.

# Capítulo 7

## Conclusão e Trabalho futuro

### 7.1 Conclusão

O trabalho realizado nesta dissertação focou-se no desenvolvimento de um sistema de percepção local que utiliza o método das grelhas de ocupação para construir um modelo do ambiente com base nos dados de profundidade fornecidos por três tipos de modalidades sensoriais, para aplicações na área da robótica móvel. Para cada tipo de tecnologia sensorial é construída uma grelha de ocupação local (i.e., centrada no sistema robótico). Como foi demonstrado nesta dissertação, as grelhas criadas individualmente não apresentam condições para modelar correctamente certas situações em ambientes reais devido às limitações e problemas que advêm dos sensores. Assim, de modo a que eles possam complementar-se em situações onde os outros possam falhar, foi apresentado um método de fusão entre as três grelhas construídas individualmente, sendo o resultado dessa fusão representado numa única grelha de ocupação local 2D. O objectivo final é utilizar essa grelha para actualização temporal das probabilidades de ocupação acerca das regiões do ambiente circundante durante a navegação do robô móvel.

No decorrer desta dissertação foi possível concluir que:

- Para o desenvolvimento do sistema de percepção, a utilização de sensores com características de observação 2D para construção de um modelo métrico do ambiente pode apresentar um conjunto de problemas (por exemplo, observar apenas as pernas de uma cadeira pode não evitar a colisão com esse obstáculo). Para reduzir esse tipo de problemas utilizaram-se sensores de observação 3D, que ainda assim, recorrendo a apenas um desses sensores, vários problemas ainda se mantêm (por exemplo, detecção de paredes/portas/janelas em vidro, obstáculos muito próximos e reflexões especulares).
- A informação das distâncias, fornecida pelos sensores, é indispensável para tarefas de percepção em ambiente reais. Cada tipo de sensor é importante na detecção de obstáculos que não foram observados pelos outros.
- As grelhas de ocupação local 2D demonstraram-se um tipo de representação do ambiente eficaz e robusto para conjugar a informação devolvida pelos sensores.
- A precisão da grelha de ocupação local 2D construída depende também da solução apresentada para o problema da localização. Nesta dissertação, foi assumido que esse problema está resolvido e a localização do robô é conhecida.

## 7.2 Trabalho futuro

Algumas questões que não foram abordadas nesta dissertação podem ser implementadas em trabalho futuro. As várias sugestões enunciadas têm como principal objectivo testar e validar a aplicação do sistema de percepção local para tarefas de assistência na navegação de robôs móveis:

- Integrar mais dispositivos sensoriais de baixo custo, nomeadamente sonares. Apesar de ser um dispositivo com várias limitações, apresenta capacidades de observação de extrema importância e torna-se uma mais valia para detecção de qualquer tipo de obstáculos;
- Explorar e avaliar novos métodos de fusão multi-sensorial em grelhas de ocupação com recurso às modalidades sensoriais propostas neste trabalho. Alguns dos métodos de fusão em grelhas de ocupação estão referenciados na tabela 2.3. Os resultados podem ser comparados com o método de fusão implementado;
- Com a informação sensorial fornecida pelo Kinect e pelo Laser 3D\*, existe a possibilidade de investigar a representação do ambiente em grelhas de ocupação local 3D. O aumento nos requisitos de memória e maior esforço computacional podem injustificar a sua utilização para tarefas de navegação dos robôs móveis;
- Testar e validar o sistema de percepção local proposto para o desenvolvimento de um sistema de segurança activo. Este sistema de segurança activo pode estar direccionado a cadeiras de rodas robóticas para utilizadores com limitações motoras graves, facilitando a sua navegação e evitando risco de colisão com os obstáculos do ambiente circundante;
- Utilizar a grelha de ocupação local 2D para desempenhar algoritmos de planeamento de trajectórias locais. A capacidade para o robô móvel atravessar correctamente a passagem de uma porta pode ser resolvida com estes métodos.

# Bibliografia

- [Adarve et al., 2012] Adarve, J., Perrollaz, M., Makris, A., and Laugier, C. (2012). Computing occupancy grids from multiple sensors using linear opinion pools. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 4074–4079.
- [Berger, 1985] Berger, J. O. (1985). *Statistical decision theory and Bayesian analysis*. Springer-Verlag, New York.
- [Bresenham, 1965] Bresenham, J. E. (1965). Algorithm for computer control of a digital plotter. *IBM Syst. J.*, 4(1):25–30.
- [Chou and Liu, 2013] Chou, Y. and Liu, J. (2013). A Robotic Indoor 3D Mapping System Using a 2D Laser Range Finder Mounted on a Rotating Four-Bar Linkage of a Mobile Platform. *International Journal Of Advanced Robotic System*.
- [Coué et al., 2006] Coué, C., Pradalier, C., Laugier, C., Fraichard, T., and Bessiere, P. (2006). Bayesian Occupancy Filtering for Multitarget Tracking: an Automotive Application. *The International Journal of Robotics Research*, 25(1):19–30.
- [El-laithy et al., 2012] El-laithy, R., Huang, J., and Yeh, M. (2012). Study on the use of microsoft kinect for robotics applications. In *Position Location and Navigation Symposium (PLANS), 2012 IEEE/ION*, pages 1280–1288.
- [Elfes, 1987] Elfes, A. (1987). Sonar-based real-world mapping and navigation. *Robotics and Automation, IEEE Journal of*, 3(3):249–265.
- [Elfes, 1989] Elfes, A. (1989). Using occupancy grids for mobile robot perception and navigation. *Computer*, 22(6):46–57.
- [Filliat and Meyer, 2003] Filliat, D. and Meyer, J.-A. (2003). Map-based navigation in mobile robots. – i. a review of localization strategies.
- [Fong, 2001] Fong, T. W. (2001). *Collaborative Control: A Robot-Centric Model for Vehicle Teleoperation*. PhD thesis, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA.
- [Hahnel, 2005] Hahnel, D. (2005). *Mapping with mobile robots*. PhD thesis, University of Freiburg. <http://d-nb.info/974035599>. Available from: <http://freidok.ub.uni-freiburg.de/volltexte/1632/index.html>.
- [Hoseinnezhad et al., 2002] Hoseinnezhad, R., Moshiri, B., and Asharif, M. (2002). Sensor fusion for ultrasonic and laser arrays in mobile robotics: a comparative study of fuzzy,

- dempster and bayesian approaches. In *Sensors, 2002. Proceedings of IEEE*, volume 2, pages 1682–1689 vol.2.
- [Kohlbrecher et al., 2011] Kohlbrecher, S., Meyer, J., von Stryk, O., and Klingauf, U. (2011). A flexible and scalable slam system with full 3d motion estimation. In *Proc. IEEE International Symposium on Safety, Security and Rescue Robotics (SSRR)*. IEEE.
- [Leonard and Durrant-Whyte, 1991] Leonard, J. and Durrant-Whyte, H. (1991). Simultaneous map building and localization for an autonomous mobile robot. In *Intelligent Robots and Systems '91. Intelligence for Mechanical Systems, Proceedings IROS '91. IEEE/RSJ International Workshop on*, pages 1442–1447 vol.3.
- [Li, 2013] Li, Y. (2013). *Stereo vision and LIDAR based Dynamic Occupancy Grid mapping : Application to scenes analysis for Intelligent Vehicles*. Theses, Université de Technologie de Belfort-Montbeliard.
- [Lopes, 2012] Lopes, A. C. (2012). *Mobile Robot Assisted Navigation based on Collaborative Control*. Phd thesis, University of Coimbra.
- [Lopes et al., 2013] Lopes, A. C., Pires, G., and Nunes, U. (2013). Assisted navigation for a brain-actuated intelligent wheelchair. *Robotics and Autonomous Systems*, 61(3):245–258.
- [Lu and Milios, 1997] Lu, F. and Milios, E. (1997). Globally consistent range scan alignment for environment mapping. *Autonomous Robots*, 4(4):333–349.
- [Moita and Nunes, 2001] Moita, F. and Nunes, U. (2001). Multi-echo technique for feature detection and identification using simple sonar configurations. In *Advanced Intelligent Mechatronics, 2001. Proceedings. 2001 IEEE/ASME International Conference on*, volume 1, pages 389–394 vol.1.
- [Mojtahedzadeh, 2011] Mojtahedzadeh, R. (2011). Robot obstacle avoidance using the kinect. Master’s thesis, KTH, School of Computer Science and Communication (CSC).
- [Montesano et al., 2008] Montesano, L., Minguez, J., and Montano, L. (2008). Modeling dynamic scenarios for local sensor-based motion planning. *Autonomous Robots*, 25(3):231–251.
- [Moras et al., 2011] Moras, J., Cherfaoui, V., and Bonnifait, P. (2011). Credibilist occupancy grids for vehicle perception in dynamic environments. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 84–89.
- [Moravec, 1988] Moravec, H. (1988). Sensor fusion in certainty grids for mobile robots. *AI Mag.*, 9(2):61–74.

- [Moravec and Elfes, 1985] Moravec, H. P. and Elfes, A. (1985). High resolution maps from wide angle sonar. In *Robotics and Automation. Proceedings. 1985 IEEE International Conference on*, volume 2, pages 116–121. IEEE.
- [Murarka and Kuipers, 2009] Murarka, A. and Kuipers, B. (2009). A stereo vision based mapping algorithm for detecting inclines, drop-offs, and obstacles for safe local navigation. In *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, pages 1646–1653.
- [Nègre et al., 2014] Nègre, A., Rummelhard, L., and Laugier, C. (2014). Hybrid Sampling Bayesian Occupancy Filter. In *IEEE Intelligent Vehicles Symposium (IV)*, Dearborn, United States.
- [Nguyen et al., 2012] Nguyen, T.-N., Michaelis, B., Al-Hamadi, A., Tornow, M., and Meinecke, M. (2012). Stereo-camera-based urban environment perception using occupancy grid and object tracking. *Intelligent Transportation Systems, IEEE Transactions on*, 13(1):154–165.
- [Pascoal et al., 2014] Pascoal, R., Santos, V., Premebida, C., and Nunes, U. (2014). Simultaneous segmentation and superquadrics fitting in laser-range data. *Vehicular Technology, IEEE Transactions on*, PP(99):1–1.
- [Pathak et al., 2007] Pathak, K., Birk, A., Poppinga, J., and Schwertfeger, S. (2007). 3d forward sensor modeling and application to occupancy grid based sensor fusion. In *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pages 2059–2064.
- [Perdigão, 2014] Perdigão, J. (2014). Collaborative-control based navigation of mobile human-centered robots. Master’s thesis, Universidade de Coimbra.
- [Petrovskaya et al., 2011] Petrovskaya, A., Perrollaz, M., Oliveira, L., Spinello, L., Triebel, R., Makris, R., david Yoder, J., Laugier, C., and Nunes, U. (2011). Awareness of road scene participants for autonomous driving.
- [Pires and Nunes, 2002] Pires, G. and Nunes, U. (2002). A wheelchair steered through voice commands and assisted by a reactive fuzzy-logic controller. *Journal of Intelligent and Robotic Systems*, 34(3):301–314.
- [Singh et al., 2010] Singh, V., Paul, R., Mehra, D., Gupta, A., Sharma, V. D., Jain, S., Agarwal, C., Garg, A., Gujral, S. S., Balakrishnan, M., Paul, K., Rao, P. V. M., and Manocha, D. (2010). Smart cane for the visually impaired: Design and controlled field testing of an affordable obstacle detection system.

- [Stepan et al., 2005] Stepan, P., Kulich, M., and Preucil, L. (2005). Robust data fusion with occupancy grid. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 35(1):106–115.
- [Tay et al., 2007] Tay, C., Mekhnacha, K., Chen, C., Yguel, M., and Laugier, C. (2007). An Efficient Formulation of the Bayesian Occupation Filter for Target Tracking in Dynamic Environments. *International Journal Of Autonomous Vehicles*.
- [Thrun, 1998] Thrun, S. (1998). Learning maps for indoor mobile robot navigation. *Artificial Intelligence*, 99:21–79.
- [Weiherer et al., 2013] Weiherer, T., Bouzouraa, S., and Hofmann, U. (2013). An interval based representation of occupancy information for driver assistance systems. In *Intelligent Transportation Systems - (ITSC), 2013 16th International IEEE Conference on*, pages 21–27.
- [Weiss et al., 2007] Weiss, T., Schiele, B., and Dietmayer, K. (2007). Robust driving path detection in urban and highway scenarios using a laser scanner and online occupancy grids. In *Intelligent Vehicles Symposium, 2007 IEEE*, pages 184–189.
- [Wulf and Wagner, 2003] Wulf, I. and Wagner, B. (2003). Fast 3D Scanning Methods for Laser Measurement Systems. In *Proceedings of the International Conference on Control Systems and Computer Science*, volume 1, pages 312–317, Bucharest, Romania.

# Anexos



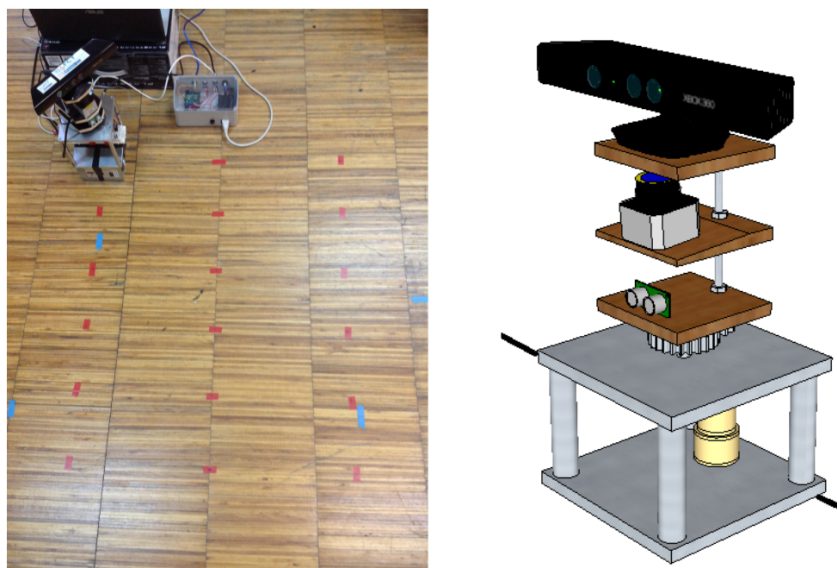
# Apêndice A

## Aquisição de dados com sistema mecatrónico

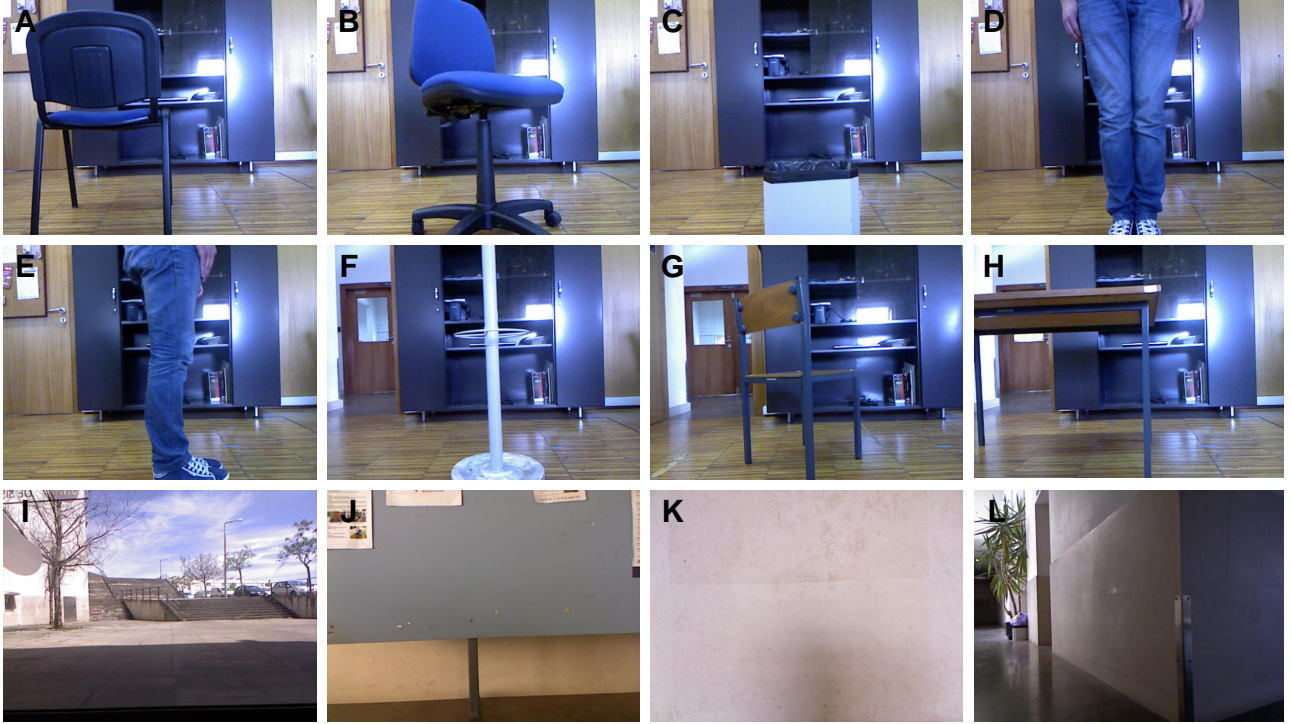
Na fase inicial deste trabalho foi adquirido um conjunto de dados que tinham por objectivo compreender os principais problemas associados aos sensores, nomeadamente, aos sonares e ao laser, perante a interacção com cenários reais. Para a realização desta experiência utilizou-se um sistema mecatrónico (figura A.1) equipado com um sonar SRF04, um laser Hokuyo e uma câmara RGB-D (apenas para aquisição de imagem RGB do cenário).

O sistema mecatrónico que incorpora os sensores referenciados é, mais precisamente, um sistema de posicionamento com um grau de liberdade ( $\theta$ ) de alta precisão. Um computador controla o sistema de posicionamento e envia os comandos para um módulo de electrónica através da portá série. Este módulo é constituído por um microcontrolador arduino e um driver de motores que executa o controlo, em malha fechada, de um motor DC.

O sonar foi colocado a uma altura de 190 cm do chão para evitar possíveis reflexões a partir dele. Um conjunto de 12 obstáculos (ver cenários reais da figura A.2), de diferentes dimensões e características, foram expostos em frente ao sonar e ao laser. Para cada obstáculo, considera-se um conjunto de localizações em que se alterna a distância  $r$  e o ângulo  $\theta$ . Para cada localização  $(r, \theta)$ , recolheram-se 200 leituras de cada sensor (o intervalo de tempo entre as aquisições consecutivas do sonar foi grande o suficiente para não ocorrerem interferências).



**Figura A.1:** Na imagem da esquerda observa-se o ambiente real utilizado para os procedimentos experimentais, assim como, o sistema mecatrónico, os sensores e o módulo de electrónica. Na figura da direita está ilustrado o sistema mecatrónico e os sensores, construído na ferramenta *SketchUp*.



**Figura A.2:** Imagens RGB obtidas a partir do Kinect. Apresentam-se o conjunto de cenários reais utilizados para aquisição sistemática de informação sensorial.

## Notação

Considere-se que  $(r, \theta)$  significa a localização do centro geométrico do sistema mecatrónico em relação ao obstáculo, onde  $r$  ( $cm$ ) representa a distância real, e  $\theta$  ( $^\circ$ ) o ângulo definido pela linha de vista dos sensores. As medidas foram realizadas num conjunto de localizações definidas como:  $r \in \mathfrak{R}$  com  $\mathfrak{R} = \{20, 40, 60, 80, 100\}$  e  $\theta \in \mathfrak{S}$  com  $\mathfrak{S} = \{-45, 0, 45\}$ .

Considere-se que  $x_i(r, \theta)$  é a leitura  $i$  adquirida na posição  $(r, \theta)$ . A média e o desvio padrão do conjunto das leituras, assumido como um conjunto de variáveis aleatórias discretas  $\{x_i(r, \theta)\}_{i=1 \dots 200}$ , são dadas respectivamente por:

$$\overline{x(r, \theta)} = E \{x_i(r, \theta)\}_{i=1 \dots 200}$$

$$\sigma(r, \theta) = \sqrt{(Var \{x_i(r, \theta)\})_{i=1 \dots 200}}$$

onde  $E \{.\}$  e  $Var \{.\}$  indicam as funções estatísticas da média e da variância no espaço amostral das leituras, respectivamente.

Nas tabelas A.1 e A.2 apresentam-se os resultados quantitativos (média e desvio padrão) acerca das medidas devolvidas pelo sonar e laser, respectivamente.

		Cenários reais												
		A	B	C	D	E	F	G	H	I	J	K	L	
$r = 20$	$\theta = 0$	$\bar{x}$	19.0	15.0	14.0	14.2	12.5	17.0	17.0	12.4	16.0	31.2	14.0	19.3
		$\sigma$	0.0	0.0	0.0	1.1	0.5	0.0	0.1	0.5	0.0	1.6	0.0	0.5
	$\theta = -45$	$\bar{x}$	43.2	39.5	26	33.8	35.5	43.1	41.2	228.4	-	-	-	-
		$\sigma$	0.4	0.6	0.0	0.8	1.6	0.6	0.4	1.1	-	-	-	-
	$\theta = 45$	$\bar{x}$	42.2	40.8	120.1	39.9	37.2	40.8	39.7	77.4	-	-	-	-
		$\sigma$	0.4	0.7	56.2	26.1	0.6	0.6	0.4	0.6	-	-	-	-
$r = 40$	$\theta = 0$	$\bar{x}$	39.5	36	34.6	35.6	31.8	37.4	39.5	33.3	36.0	45.9	35.0	35.0
		$\sigma$	0.5	0.7	0.7	0.8	1.2	0.7	5.1	0.4	0.0	0.4	0.7	0.0
	$\theta = -45$	$\bar{x}$	56.6	54.6	223.9	52.3	75.7	50	52.3	220.2	-	-	-	-
		$\sigma$	0.7	0.5	8.4	3.3	50.4	0.2	0.5	16.5	-	-	-	-
	$\theta = 45$	$\bar{x}$	54.3	52.3	172.3	46.9	49.6	52.9	51.2	161.4	-	-	-	-
		$\sigma$	0.5	0.6	0.6	1.3	2.3	1.8	0.4	0.6	-	-	-	-
$r = 60$	$\theta = 0$	$\bar{x}$	57.6	55.4	54.2	56.1	49.8	50.2	96.9	52.6	56.1	52.5	53.2	57.1
		$\sigma$	0.5	0.5	1.1	2.6	0.8	0.4	22.1	0.5	0.3	0.5	0.4	0.4
	$\theta = -45$	$\bar{x}$	71.3	57	221.2	69.2	73.1	63.3	68.9	189.2	-	-	-	-
		$\sigma$	0.5	0.5	9.8	2.9	29.1	0.5	0.2	49.6	-	-	-	-
	$\theta = 45$	$\bar{x}$	70	65.7	208.4	63	68.1	62.5	105.5	172.8	-	-	-	-
		$\sigma$	0.6	1.5	15.7	0.7	3.0	0.6	0.5	0.7	-	-	-	-
$r = 80$	$\theta = 0$	$\bar{x}$	76.7	66.8	73.2	69.6	70.1	67.1	106.6	72.1	75.7	68.9	72.7	76.7
		$\sigma$	0.5	8.6	0.5	2.5	0.6	0.2	0.7	0.5	0.5	0.6	0.5	0.7
	$\theta = -45$	$\bar{x}$	97.1	73.6	223.6	79.2	87.5	80.6	125.8	220.9	-	-	-	-
		$\sigma$	25.6	2.4	0.6	0.7	1.1	0.7	19.4	13.1	-	-	-	-
	$\theta = 45$	$\bar{x}$	86.6	79.9	252.6	80.4	82.6	78.6	121.3	84.2	-	-	-	-
		$\sigma$	0.7	1.1	4.2	1	3.3	0.7	0.6	0.4	-	-	-	-
$r = 100$	$\theta = 0$	$\bar{x}$	95.1	84.4	94.4	83.9	90.6	85.1	96.6	92.4	93.8	86.2	92.5	99.2
		$\sigma$	0.3	10.4	1.7	0.7	1.8	0.2	0.5	0.5	0.4	0.4	0.5	2.1
	$\theta = -45$	$\bar{x}$	164.9	97.2	226.2	97.9	106.8	97.4	238.1	218.6	-	-	-	-
		$\sigma$	58.3	1.8	6.6	1.1	2.5	0.5	14.2	24.7	-	-	-	-
	$\theta = 45$	$\bar{x}$	129.4	86.4	252.2	97.9	116	95.2	140.2	199.9	-	-	-	-
		$\sigma$	0.5	2.6	4.5	1.9	37.9	0.4	0.4	4.6	-	-	-	-

**Tabela A.1:** Resultados quantitativos (média e desvio padrão) obtidos a partir da aquisição de dados do sonar.

		Cenários reais									
		D	E	F	G	H	I	J	K	L	
$r = 20$	$\theta = 0$	$\bar{x}$	24.5	17.7	21.1	20.2	18.7	23.8	35.1	20.1	23.4
		$\sigma$	0.52	0.29	0.14	0.26	0.25	2.12	2.46	0.22	0.36
	$\theta = -45$	$\bar{x}$	38.2	36.2	46.2	43.3	42.7	-	-	-	-
		$\sigma$	0.45	0.26	0.27	0.31	0.29	-	-	-	-
	$\theta = 45$	$\bar{x}$	38.2	39.5	44.6	43.1	42.5	-	-	-	-
		$\sigma$	0.26	0.25	0.23	0.28	0.38	-	-	-	-
$r = 40$	$\theta = 0$	$\bar{x}$	45.2	34.5	40.8	40.3	37.4	43.9	83.9	40.5	39.7
		$\sigma$	0.34	0.34	0.19	0.35	0.23	2.89	0.42	0.28	0.47
	$\theta = -45$	$\bar{x}$	56.7	51.8	59.5	55.9	56.1	-	-	-	-
		$\sigma$	0.53	0.46	0.28	0.28	0.32	-	-	-	-
	$\theta = 45$	$\bar{x}$	60.8	50.5	58.5	54.9	53.9	-	-	-	-
		$\sigma$	0.34	0.38	0.23	0.30	0.34	-	-	-	-
$r = 60$	$\theta = 0$	$\bar{x}$	63	53.6	60.3	60.2	57.5	60.6	104.9	59.7	65.2
		$\sigma$	0.51	0.29	0.19	0.33	0.28	3.01	0.24	0.27	0.39
	$\theta = -45$	$\bar{x}$	76.5	71.9	76.6	72.3	71.9	-	-	-	-
		$\sigma$	0.29	0.25	0.29	0.35	0.35	-	-	-	-
	$\theta = 45$	$\bar{x}$	75.6	66.0	75.2	70.9	69.6	-	-	-	-
		$\sigma$	0.28	0.33	0.24	0.31	0.37	-	-	-	-
$r = 80$	$\theta = 0$	$\bar{x}$	84.1	76.1	82.5	80.8	79.1	82.8	125.1	80.1	87.1
		$\sigma$	0.35	0.35	0.29	0.37	0.31	2.93	0.2	0.22	0.37
	$\theta = -45$	$\bar{x}$	94.7	89.9	95.9	90.9	90.8	-	-	-	-
		$\sigma$	0.35	0.29	0.17	0.49	0.41	-	-	-	-
	$\theta = 45$	$\bar{x}$	94.7	97.9	93.8	89.2	88.4	-	-	-	-
		$\sigma$	0.37	0.33	0.26	0.43	0.33	-	-	-	-
$r = 100$	$\theta = 0$	$\bar{x}$	104.9	96.2	102.2	102.1	100.8	131.6	145.5	100.5	134.5
		$\sigma$	0.38	0.27	0.27	0.45	0.27	97.8	0.26	0.27	0.47
	$\theta = -45$	$\bar{x}$	114.1	108.8	114.5	111.1	110.6	-	-	-	-
		$\sigma$	0.34	0.28	0.18	0.49	0.44	-	-	-	-
	$\theta = 45$	$\bar{x}$	112.3	107.0	111.7	108.8	107.3	-	-	-	-
		$\sigma$	0.39	0.33	0.28	0.57	7.65	-	-	-	-

**Tabela A.2:** Resultados quantitativos (média e desvio padrão) obtidos a partir da aquisição de dados do laser (análise para apenas um feixe).