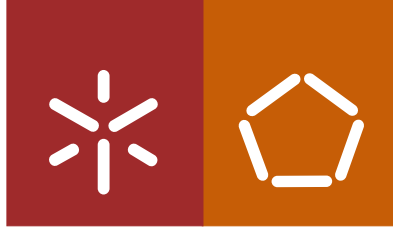


Universidade do Minho
Escola de Engenharia

Nair Isabel Braga Simões Alves

Uma Solução para Navegação Indoor



Universidade do Minho

Escola de Engenharia

Nair Isabel Braga Simões Alves

Uma Solução para Navegação Indoor

Dissertação de Mestrado
Mestrado Integrado em Engenharia Biomédica

Trabalho realizado sob a orientação do
Professor Jorge Gustavo Rocha

Outubro de 2012

Declaração

Nome: Nair Isabel Braga Simões Alves

Endereço eletrónico: nairbsalves@gmail.com

Bilhete de Identidade: 13563552

Título da Dissertação: Uma Solução para Navegação Indoor

Orientador: Professor Jorge Gustavo Rocha

Ano de conclusão: 2012

Designação do Mestrado: Mestrado Integrado em Engenharia Biomédica

Area de Especialização: Ramo de Informática Médica

É AUTORIZADA A REPRODUÇÃO PARCIAL DESTA DISSERTAÇÃO APENAS PARA EFEITOS DE INVESTIGAÇÃO, MEDIANTE DECLARAÇÃO ESCRITA DO INTERESSADO, QUE A TAL SE COMPROMETE,

Universidade do Minho, ___ / ___ / ___

Assinatura: _____

Agradecimentos

Em primeiro lugar, expresso o meu profundo agradecimento ao Professor Jorge Gustavo Rocha por toda a orientação e o apoio demonstrado ao longo deste projeto. Agradeço pela motivação e disponibilidade que permitiram com sucesso a finalização da presente dissertação.

Sem menor gratidão, um obrigada aos meus colegas de gabinete: Juliana, Pedro, Bruno e Nuno, pelo fantástico ambiente de trabalho proporcionado. Ao Suxy e Marco, grandes amigos e parceiros, pela troca de ideias, pela ajuda e, sobretudo pelo companheirismo demonstrado. Aos meus amigos de sempre e para sempre: à Maria, ao Hori, à Di, à Claudia, à Teca, ao Pifo, pela compreensão, divertimento e, acima de tudo, pela amizade incondicional ao longo destes cinco anos.

Não podia deixar de agradecer aos meus pais e irmão por todo o carinho, apoio e confiança. Por fim, um especial agradecimento ao Rui pela dedicação, pelo carinho e paciência incansáveis.

Um sincero Obrigada a todos!

Resumo

Os sistemas de localização e navegação *indoor* permitem determinar a posição de pessoas ou certos objetos em grandes edifícios, assim como ajudam na navegação e orientação dentro destes ambientes fechados. A sua aplicabilidade pode-se estender a diversas áreas, tais como: hospitalar, o turismo, e até infantaria militar ou um guia para invisuais.

De acordo com as necessidades específicas de localização *indoor*, a abordagem clássica do **GPS** é inadequada, uma vez que este é completamente inoperacional em espaços fechados. A navegação *indoor* levanta assim desafios adicionais quando comparada com o *outdoor*. O interesse no *indoor* tem-se expandido significativamente, e como tal, diversos métodos têm vindo a ser estudados e apresentados, com resultados, custos e contextos bastante diferentes entre si.

Dentro deste contexto, a presente dissertação tem como objetivo criar uma solução para a navegação no interior de edifícios, que resolva os problemas característicos do *indoor*. Para tal, foi necessário fazer um levantamento bibliográfico, principalmente na *web* devido a ser uma área muito recente. Dessa pesquisa resultou uma avaliação dos métodos e soluções de *indoor* existentes.

Perante a dificuldade de encontrar uma forma consensual de representar espaços *indoor*, foi apresentado um possível modelo para a captação e representação destes espaços. Assim, tendo como base os dados provenientes do **Open Street Map (OSM)** e um software *open source* de *routing* multi-modal - o **Open Trip Planner (OTP)**, foi criada uma plataforma de *routing* adequada à perceção da informação *indoor*. Esta plataforma - NavIOS, encontra-se disponível na *Web*.

Os objetivos deste projeto foram alcançados com sucesso, conseguindo-se uma representação simples dos espaços *indoor*, uma solução de *routing* para estes espaços, e que esta representação fosse perceptível à solução criada. Apenas não se

conseguiram disponibilizar na plataforma os dados 3D, para visualização e navegação, apesar de estes dados terem sido previamente preparados. Este será um passo futuro, na continuação deste projeto.

Abstract

The indoor navigation and localization systems allow to determine the position of people and objects within large buildings, as well as they help in the process of navigation and orientation in such closed environments. These systems can be applied in many different fields, such as the hospital, the tourism and the military ones, as well as a guide for blind people.

According to the specific needs of the indoor localization, the classic approach of [GPS](#) turns out to be inadequate due to its inoperability in closed spaces. This way, the indoor navigation raises additional problems when compared to the outdoor navigation. The interest in such systems have been increasing over the past few years, leading to a wide number of studies concerning different methods, with different results, costs and contexts.

In this context, the present work has the purpose of creating a solution for the navigation within buildings, to overcome the typical problems of indoor. It was necessary a literature review, mainly focused on web resources, since this is a very recent problem. An evaluation of the indoor methods and solutions that exist resulted from the review.

Facing the difficulty to find a consensual way to represent indoor spaces, it was presented a possible method to capture and represent information within these spaces. Thus, relying on the data from [Open Street Map \(OSM\)](#) and a multi-modal routing open-source software - [Open Trip Planner \(OTP\)](#), it was created a web-based routing platform - [NavIOS](#). This platform is able to perceive the indoor information.

The objectives of this project were achieved successfully since it was obtained a simple way of indoor space representation, a routing solution for such spaces, and the perception from this solution to the way of representation created. The

only objective we can consider as not concluded was the 3D visualization and navigation data in the platform, although this data had been prepared earlier. This represents the next step of the project, to be worked on in the future.

Conteúdo

Lista de Figuras	xiii
Lista de Tabelas	xv
Lista de Códigos	xvii
Lista de Acrónimos	xviii
1 Introdução	1
1.1 Contexto e Motivação	1
1.2 Objetivos	3
1.3 Estrutura do Documento	4
2 Sobre a Navegação Indoor	7
2.1 Posicionamento	7
2.1.1 Pedestrian Dead Reckoning	8
2.1.2 Rádio Frequência (RF)	15
2.1.3 Análise de Imagem	24
2.2 Representação dos Espaços	31
2.2.1 Como obter os dados 3D	31
2.2.2 Conetividade (suporte ao routing)	32
2.3 Soluções sobre o OSM	33
2.4 Soluções Comerciais	38
3 Levantamento a partir de imagens	41
3.1 Aquisição de Dados	42

3.2	Geração de Modelos	45
3.3	Visualização dos Modelos	52
4	NavIOS: Arquitetura e Implementação	55
4.1	Arquitetura	55
4.2	Tratamento de Dados	57
4.2.1	Dados 3D	59
4.2.2	Geoserver	60
4.2.3	JOSM	61
4.3	Implementação	66
4.3.1	Geração do Grafo	67
4.3.2	Interface	70
5	Testes e Resultados	77
5.1	Caso 1: Perdido no DI	77
5.2	Caso 2: Apanhar autocarro para a Av. Central	80
5.3	Caso 3: Passeio de bicicleta em Braga	82
6	Conclusão e Trabalho Futuro	85
6.1	Conclusão	85
6.2	Trabalho Futuro	87
	Bibliografia	88
A	Publicações	97
A.1	Localização em Espaços Interiores	97
A.2	OSM Indoor: moving forward	97

Lista de Figuras

2.1	Localização dos sensores na técnica PDR (adaptada de [1]).	9
2.2	Método de deteção de passos (exemplo) [2].	11
2.3	Diagrama representativo do processo <i>map matching</i> (adaptada de [3]).	14
2.4	Divisão das tecnologias rádio frequência por classes.	16
2.5	<i>Beacon</i> de posicionamento <i>indoor</i> (adaptada de [4]).	18
2.6	Exemplo da localização de um objeto perdido: mala representada pelo círculo vermelho e o utilizador pelo círculo verde (adaptada de [4]).	18
2.7	Áreas de potenciais usos da tecnologia NFC (adaptada de [5]).	21
2.8	Exemplos da aplicação da tecnologia NFC (adaptada de ^{5 6 7} e [6]).	22
2.9	Exemplo do sistema de posicionamento <i>indoor</i> aplicado em [7].	25
2.10	Exemplo de um <i>QR-Code</i> : (a) Fase de captura do <i>qr-code</i> ; (b) Decodificação da informação subjacente ao <i>qr-code</i> em a).	28
2.11	Interface <i>web</i> da aplicação <i>IndoorOSM</i> com dados 2D (a) e 3D (b), Universidade Heidelberg [8, 9].	35
2.12	Interface <i>web</i> da aplicação <i>Campus GIS</i> , Universidade Klagenfurt [10].	36
2.13	Interface <i>web</i> da aplicação <i>Campus Guiden</i> , NTNU [11].	37
2.14	Interface móvel das aplicações <i>Google Maps 6.0</i> , <i>Navteq</i> e <i>Bing Maps</i> respetivamente (adaptada de ^{12 13 14}).	39
3.1	Fluxo típico da metodologia SFM.	42
3.2	Diagrama esquemático de um sistema de medição baseado em triangulação ótica (adaptada de [12]).	44

3.3	Exemplo teste no VisualSFM: (a) <i>Upload</i> das fotos - 1a fase do processo; (b) Reconstrução e visualização da nuvem pontos - fase final do processo.	47
3.4	Exemplo teste no My3DScanner: (a) Visualização da nuvem de pontos na página <i>web</i> do serviço; (b) Visualização da <i>mesh</i> originada.	49
3.5	Exemplo teste no 123D Catch: (a) Visualização da nuvem de pontos com as posições da câmara ativas (' <i>textura + wirefram</i> '); (b) Visualização da nuvem de pontos só com a componente ' <i>wireframe</i> ').	50
3.6	Exemplo teste no Apero/MICMAC: (a) e (b) visualização no MeshLab do modelo obtido.	52
3.7	Exemplo teste do importe de um ficheiro PLY no MeshLab.	53
4.1	Diagrama representativo do modelo MVC, exemplificando a relação entre as três camadas da arquitetura (modelo, visão e controlador).	57
4.2	Processo geral de tratamento dos dados.	58
4.3	Resultado da conversão do modelo 3D do DI para Collada (. <i>dae</i>).	59
4.4	Exemplo da pré-visualização no Geoserver do <i>layer</i> relativo ao piso três do DI.	60
4.5	Exemplo da construção do grafo no JOSM para o piso 1, tendo como auxílio a planta do piso (ficheiro . <i>osm</i>).	62
4.6	<i>Tags</i> utilizadas para descrever um nodo.	63
4.7	<i>Tags</i> utilizadas para descrever uma aresta.	63
4.8	Representação do elevador no grafo de <i>routing</i>	64
4.9	Exemplo da representação da escadas no grafo de <i>routing</i>	65
4.10	Visualização no JOSM do grafo de <i>routing</i>	65
4.11	Interface gráfica principal da aplicação proposta - NavIOS.	71
4.12	<i>Tags</i> utilizadas para descrever um edifício.	71
4.13	Representação da informação <i>indoor</i> relativa ao Departamento de Informática, com a visualização dos nodos do piso 3.	73
5.1	Routing que apresenta o caminho mais curto para o tipo de transporte 'a Pé': (a) visualização da parte da rota relativa ao piso 2; (b) visualização da parte da rota relativa ao piso 3.	78

5.2	Routing que apresenta o caminho mais curto para o tipo de transporte 'cadeira de rodas': (a) visualização da parte da rota relativa ao piso 0; (b) visualização da parte da rota relativa ao piso 3.	79
5.3	Routing que apresenta o caminho mais curto para o tipo de transporte 'Autocarro + a Pé'.	81
5.4	Routing que apresenta o caminho mais curto para tipo de transporte 'Autocarro + a Pé'.	81
5.5	Parâmetros escolhidos aquando do planeamento de uma viagem personalizada para o tipo de transporte 'bicicleta'.	82
5.6	Routing que apresenta a viagem personalizada para o tipo de transporte 'bicicleta'.	83

Lista de Tabelas

2.1	Quadro de comparação das tecnologias abordadas durante a secção.	30
2.2	Comparação de cada proposta apresentada nesta secção com uma lista de características importantes a destacar.	38

Lista de Códigos

4.1	Comando de conversão de <code>.dxf</code> em <code>.shp</code>	60
4.2	Comando de conversão de <code>.shp</code> em <code>.gpx</code> , e por fim, para <code>.osm</code>	61
4.3	Módulo de importação do ficheiro <code>GTFS</code>	68
4.4	Módulo de importação do <code>OSM</code>	69
4.5	Módulo de importação do ficheiro de <code>altimetria</code>	69
4.6	Mensagens finais no processo de geração do grafo.	70
4.7	Função que efetua a <i>query</i> à <code>XAPI</code> para carregar a informação dos edifícios com pisos.	72
4.8	Função que efetua a <i>query</i> à <code>XAPI</code> para carregar os nodos de um determinado piso.	73
4.9	Função 'criar <i>layer</i> dos edifícios e <i>combobox</i> com os pisos'.	74
4.10	Função 'cálculo do nodo mais próximo'.	75
4.11	Exemplo de um pedido feito no <code>OTP</code>	76

Lista de Acrónimos

- 2D** Bidimensional. xi, 3, 4, 27, 31, 34, 35, 39, 41, 48, 55, 58–60, 85–87
- 3D** Tridimensional. vi, viii, xi, xii, 3, 4, 31, 34, 35, 41–43, 45–54, 57–59, 85–87
- Apero** Aérotriangulation Photogrammétrique Expérimentale Relativement Opérationnelle. 51
- BIM** Building Information Model. 31, 32, 57
- CAD** Computer Aided Design. 34, 45, 57
- GPS** Global Position System. v, vii, 1, 2, 7, 8, 10, 12, 13, 17, 19, 33, 39, 85
- INS** Inertial Navigation Systems. 8, 13
- ISM** Industrial Scientific and Medical. 20
- LBS** Location Based Services. 7, 28
- LS3D** Laser Scanning 3D. 42, 43
- MAC address** Media Access Control address. 17
- MAP** Map-Aided Positioning. 13
- MEMS** Micro-Electro-Mechanical Systems. 9
- MICMAC** Multi-Images Correspondances, Méthodes Automatiques de Corrélation. 51

- MVC** Model View Controller. xii, 56, 57
- NFC** Near Field Communication. xi, 2, 8, 15, 21–23, 39
- OSM** Open Street Map. v, vii, xvii, 33, 34, 37, 38, 58, 60, 61, 64, 66, 68, 69, 71, 72, 86
- OTP** Open Trip Planner. v, vii, xvii, 55, 66, 69, 75, 76, 86
- PDA** Personal Digital Assistant. 20
- PDR** Pedestrian Dead Reckoning. 8–10, 13, 14
- PLY** Polygon File Format. xii, 47, 48, 53
- PNA** Personal Navigation Assistants. 13
- PNS** Pedestrian Navigation Systems. 11
- RFID** Radio-Frequency Identification. 2, 8, 15, 19–21, 39
- RSS** Received Signal Strength. 15
- RSSI** Received Signal Strength Indicator. 17
- SFM** Structure From Motions. xi, 41, 42, 48
- SIFT** Scale-Invariant Feature Transform. 25
- SURF** Speeded Up Robust Features. 25
- TLS** Terrestrial Laser Scanning. 44, 45
- UWB** Ultra Wide Band. 8, 15, 19
- VGI** Volunteered Geographic Information. 33, 34
- WLAN** Wireless Local Area Network. 14, 15

Capítulo 1

Introdução



[cartoonstock.com, 2012]

1.1 Contexto e Motivação

O problema da localização divide-se na realidade em dois subproblemas com características próprias: a localização em espaço aberto (*outdoor*), que devido à localização via satélite, possui há já alguns anos uma solução popular e global - o *Global Position System (GPS)*; e a localização em espaços fechados (*indoor*), com condicionantes bastantes diferentes do outdoor.

A navegação em espaços abertos tem evoluído muito ao longo dos últimos anos. Para isso, contribuiu muito a proliferação de chips **GPS** de muito baixo custo, a liberalização de produção de cartografia e de imagens aéreas de alta resolução, e a consequente disponibilização de mapas a baixo custo, acessíveis ao público geral. O **GPS** tornou-se de facto um *standard* em navegação. No entanto, este dispositivo falha em áreas urbanas onde há pouca visibilidade do céu, e é completamente inoperacional em espaços fechados, como parques de estacionamento ou centros comerciais.

Com a contínua urbanização e com o crescente número de edifícios de grandes dimensões, o raio de ação do **GPS** é cada vez mais restrito de dia para dia, acompanhado de uma diminuição do número de espaços abertos. O facto do estilo de vida moderno estar cada vez mais sedentário, e com isso, as pessoas passarem muito do seu tempo em espaços públicos, onde não chega o sinal **GPS**, nem câmaras fotográficas dos aviões e onde não entram cartógrafos, é também uma das razões para a necessidade de um sistema de posicionamento direcionado para ambientes *indoor*. Estima-se que em média uma pessoa dispense 90% do seu tempo em espaços fechados [13]. Hospitais, centros comerciais, universidades, escolas, lojas do cidadão, entre outros, são espaços fechados com dinâmicas enormes em termos de movimentação de pessoas, onde ainda não há soluções de navegação que se possam comparar às que já existem para o exterior.

No contexto da navegação *indoor* proliferam diversos métodos sem, no entanto, existir um que domine o conceito para todos os casos de utilização desejados. Existem diversas propostas recentemente encontradas na literatura [14, 15], e há já algumas soluções comerciais. Algumas destas propostas baseiam-se na contagem dos passos a partir da utilização combinada de sensores de aceleração e orientação. Outras soluções tentam adaptar os princípios tradicionais da triangulação a espaços *indoor*, recorrendo a repetidores de sinal **GPS** (por exemplo **GPS+beacon**), à rede Wi-Fi ou a outras opções rádio (como **NFC** e **RFID**). Em termos de análise de imagem, existem algumas soluções que identificam as características naturais de uma imagem, outras que acrescentam ao ambiente marcas fiduciais para ajudar a reconhecer pontos conhecidos (por exemplo, as cores e as letras utilizadas em alguns parques de estacionamento). Contudo, cada um destes métodos apresenta limitações particulares.

Um Hospital é um local público, que movimenta muitas pessoas, quer os profissionais que aí prestam serviços, quer todo o tipo de visitantes: médicos, doentes, acompanhantes e paramédicos. Estes últimos, dado que não frequentam assiduamente o local, têm sempre dificuldade em se orientar. O hospital, dado o número de serviços e de salas, é comparável a uma pequena cidade, em termos da diversidade de destinos que se podem querer. É neste contexto que surge a necessidade de arranjar uma solução para a navegação no interior dos Hospitais, solucionando assim as necessidades dos vários perfis de utilização, desde as visitas aos profissionais de saúde.

Por outro lado, a visualização de grandes espaços como hospitais, centros comerciais, entre outros, geralmente é reduzida a planos 2D, e há uma capacidade limitada de apresentar os melhores percursos de uma forma facilmente perceptível. Com a utilização de modelos 3D, cada vez mais aplicados atualmente, esta questão pode ser em grande parte resolvida. Isto porque a visualização e orientação dentro dos edifícios de grandes dimensões torna-se mais perceptível para as pessoas, com este tipo de modelos. Além disto, cada vez mais existe a possibilidade de interagir com modelos 3D através de plataformas *web*, o que torna esta questão mais desafiante e promissora.

1.2 Objetivos

Dentro de todo o contexto de navegação *indoor*, esta dissertação apresenta como principal objetivo criar uma solução para a navegação no interior de edifícios, que resolva os problemas característicos do *indoor*.

Para tal, alguns objetivos mais específicos a realizar são:

- ◇ O levantamento e estudo das tecnologias existentes no contexto de localização e navegação *indoor*, assim como soluções existentes;
- ◇ O estudo das técnicas específicas para o levantamento de estruturas 3D a partir de imagens;
- ◇ A criação do modelo 3D (do edifício em estudo) e sua visualização;
- ◇ A geração do grafo para cálculos de *routing*;

- ◇ A Navegação e apresentação dos percursos em 2D e 3D;
- ◇ O Teste e avaliação da solução proposta.

1.3 Estrutura do Documento

A presente dissertação foi estruturada em 6 capítulos, organizados da seguinte forma:

Capítulo 1 - Introdução

O presente capítulo é um capítulo introdutório, onde se apresenta o contexto e a motivação que deu origem a este trabalho, os objetivos propostos, e por fim, a estrutura da dissertação.

Capítulo 2 - Sobre Navegação Indoor

Neste capítulo é feito um levantamento dos conceitos de localização e navegação em espaços fechados (*indoor*). Começa-se por apontar as limitações das tecnologias para espaços abertos (*outdoor*), para depois se apresentarem as tecnologias de posicionamento dentro de edifícios, as diferentes abordagens à representação desses espaços e por fim referem-se as principais iniciativas, quer sobre os dados do *Open Street Map*, quer comerciais.

Capítulo 3 - Levantamento a partir de imagens

No terceiro capítulo são apresentadas as técnicas específicas para o levantamento de estruturas 3D a partir de imagens. São também ilustrados alguns testes efetuados com software para criação de nuvens de pontos e modelos 3D.

Capítulo 4 - Arquitetura e Implementação

O quarto capítulo apresenta a solução *indoor* proposta na presente dissertação. É descrita detalhadamente a arquitetura escolhida, assim como o tratamento de dados necessário, e por fim, apresentado sumariamente todo o processo na implementação da aplicação. A descrição da arquitetura está alinhada com o modelo MVC.

Capítulo 5 - Testes e Resultados

Neste penúltimo capítulo é apresentada a avaliação da aplicação desenvolvida.

Demonstram-se as várias soluções de navegação para diferentes origens e destinos, de acordo com os parâmetros de *routing*.

Capítulo 6 - Conclusões e Sugestões Futuras

Por fim, no sexto capítulo, é feita uma análise crítica ao trabalho desenvolvido e apresentam-se sugestões para a continuação deste trabalho.

Capítulo 2

Sobre a Navegação Indoor

O problema da localização encontra-se dividido, na realidade, em duas categorias: a localização em espaço aberto (*outdoor*), e a localização em espaços fechados (*indoor*). Como já referido anteriormente, os sinais de GPS são completamente inoperacionais em espaços fechados, portanto, um sistema de posicionamento diferente é necessário para *indoor*. Nas duas últimas décadas, vários sistemas de localização têm vindo a ser introduzidos [16, 17, 18, 19, 20, 21].

2.1 Posicionamento

Os sistemas de posicionamento *indoor* são utilizados para determinar a posição de pessoas ou certos objetos, em grandes edifícios ou ambientes fechados. Nos últimos anos, estes sistemas tornaram-se muito populares, pode-se até mesmo dizer, que estes estão em voga. No entanto, a navegação dentro de edifícios ainda se apresenta como um desafio.

A sua aplicabilidade pode-se estender a inúmeras áreas, tais como: serviços baseados em localização (*Location Based Services (LBS)*, do inglês), hospitalar (localizar pacientes num hospital), o turismo e, até mesmo infantaria militar ou um guia para invisuais. Diversos métodos de localização *indoor* têm vindo a ser estudados, com resultados, custos e contextos bastante diferentes entre si [14]. Nesta secção serão abordados alguns destes métodos através duma descrição breve da sua metodologia, uma análise das suas potencialidades e os seus pontos fracos.

As técnicas de posicionamento podem-se dividir, de acordo com as tecnologias utilizadas, em três grupos: *Pedestrian Dead Reckoning (PDR)*, *Radio Frequency based* (WiFi, GPS+beacon, UWB, RFID, Bluetooth e NFC) e *Image Analysis based*.

2.1.1 Pedestrian Dead Reckoning

O *Pedestrian Dead Reckoning (PDR)* é uma das técnicas com maior relevância na localização *indoor*, sendo baseado nos populares Sistemas de Localização Inercial (*Inertial Navigation Systems (INS)*). Esta técnica tem vindo a ser aplicada em vários projetos [22, 23, 24].

A sua metodologia, baseada no processo *Dead Reckoning*, adapta-se à dinâmica (caraterísticas específicas) do caminhar humano [25]. Portanto, a base principal desta técnica assenta na estimação acumulada do deslocamento (norma e direção) da pessoa/objeto (neste caso o utilizador do sistema) para, conseqüentemente, determinar a sua posição atual [2]. Para conseguir isto, o dispositivo apresenta sensores integrados, os quais serão brevemente descritos mais abaixo.

A precisão desta técnica varia bastante de acordo com inúmeros fatores, como: a precisão dos sensores utilizados, as caraterísticas físicas dos passos do sujeito, o trajeto, o local onde o utilizador leva o dispositivo de medição e variáveis do ambiente (por exemplo, a existência ou não de interferências magnéticas). Por esta mesma razão, na literatura existente não se encontram muitos dados absolutos para este método, contudo, em alguns artigos está reportado um erro inferior a 5% da distância percorrida [2, 26, 27]. Embora esta seja uma boa precisão, pode não ser suficiente, uma vez que o erro deste método é cumulativo. Para muitas aplicações previstas de posicionamento *indoor*, o erro de posicionamento máximo de 5-10 metros (por exemplo, escala de um quarto) é necessário [28].

2.1.1.1 Sensores

Sensores inerciais independentes num sistema PDR, como acelerómetros, bússolas/magnetómetros e giroscópios, são utilizados para detetar a ocorrência de passos e proporcionar um meio de determinar a distância e direção em que estes foram dados. Dependendo da posição destes sensores autónomos (ver Figura 2.1),

o método PDR pode ser classificado em duas categorias: (a) *waist-mounted* e (b) *shoe-mounted* [29].

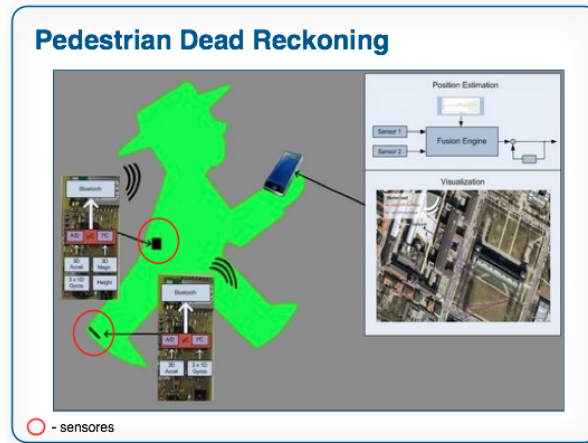


Figura 2.1: Localização dos sensores na técnica PDR (adaptada de [1]).

Acelerómetros

Dentro de todos os modelos e aplicabilidades que possam surgir para acelerómetros, a sua função mantém-se sempre a mesma: medição de forças de aceleração. Partindo da segunda Lei de Newton, que diz que a força gerada por um objeto é definida pela sua massa e aceleração, é possível construir vários tipos de acelerómetros tais como: acelerómetros baseados em MEMS, capacitivos e piezoelétricos. Por outras palavras, acelerómetros são sensores que medem a aceleração, isto é, a taxa de variação da velocidade ao longo do tempo, e é definida pela sua magnitude e direção. A determinação da inclinação, só se torna possível se o acelerómetro ler a aceleração linear (aceleração que o dispositivo é sujeito) e a força gravitacional em cada eixo [24]. Sistemas de três eixos encontram-se presentes em grande parte dos *smartphones* actuais. Dentro das inúmeras aplicações nestes dispositivos, uma das mais populares é por exemplo, a alternância entre o modo de ecrã horizontal e vertical, considerando a inclinação do *smartphone* [30].

Bússolas/Magnetómetros

A bússola, considerada um célebre instrumento de navegação, é utilizada para determinar a direcção relativa aos pólos magnéticos terrestres. No entanto, a sua

precisão é fortemente afetada por interferências magnéticas provocadas pela proximidade a outros campos magnéticos, ou metais ferrosos. Para além disso, após uma perturbação como a derivada de um movimento brusco, este sensor demora algum tempo até estabilizar novamente a sua orientação. Hoje em dia, já se consegue encontrar facilmente este sensor em diversos dispositivos móveis. Recorrendo a magnetómetros (2 ou 3 eixos) consegue-se também determinar a orientação do campo magnético. Estes sensores, para além da direção do campo magnético dominante, medem ainda a intensidade da força resultante do campo magnético [24].

Giróscopios

Os giroscópios eletrónicos determinam a orientação do objeto, em qualquer momento, tendo como base os princípios do momento angular. Normalmente, estes sensores apresentam alguns erros de calibração, como o *offset* e *drift*¹, podendo estes serem corrigidos através do uso simultâneo de outros sensores que não estejam sujeitos ao mesmo tipo de problemas (por exemplo, acelerómetro, magnetómetros, entre outros.). Além disso, as medições dos giroscópios podem ser afetados por condições do ambiente, como a temperatura [31]. Atualmente, já se existem alguns dispositivos móveis que incluem estes sensores, como o *iPhone 4*, embora ainda menos populares que os acelerómetros [30].

2.1.1.2 Metodologia do PDR

Inicialmente, este tipo de métodos requer uma posição inicial conseguida através de um método alternativo (geralmente o GPS), para consequentemente poder determinar a posição atual a partir da estimativa do deslocamento efetuado.

Conseguida esta posição, o algoritmo PDR é naturalmente decomposto em três parâmetros principais [28]:

1. Detecção de passos;
2. Estimativa do comprimento do passo;
3. Determinação da orientação.

¹O *offset* é um desfasamento constante que corresponde ao valor reportado pelo giroscópio em repouso. Por sua vez, o *drift* é originário do acumular de erros de precisão aquando da medição.

Deteção de passos

A deteção de passos constitui um parâmetro importante no algoritmo de *Dead Reckoning* nos *Pedestrian Navigation Systems (PNS)*, e como tal, diversos autores têm vindo a estudar este parâmetro, existindo já várias propostas.

Toda a informação necessária para detetar a ocorrência de um passo encontra-se no sinal do acelerómetro. A determinação de um *threshold*, para uma deteção de passo confiável, não é uma tarefa simples, visto que o padrão do sinal de impacto depende do tipo de movimento (subir ou descer escadas, correr, rastejar, etc.) e do tipo de terreno sobre o qual a pedestre caminha (superfície dura ou mole, areia). Outro fator que influencia as leituras da aceleração, e conseqüentemente, a deteção de passo, é a localização do dispositivo. Idealmente, o dispositivo deve estar fixo e numa posição baixa, por exemplo no tornozelo [26]. Nos dispositivos *handheld* (p.ex., *smartphones*), o impacto da ocorrência de passos é enfraquecido e menos preciso. Contudo, com a crescente massificação do uso de *smartphones* e de sensores mais precisos, diverso trabalho científico está a ser desenvolvido nesta área [32].

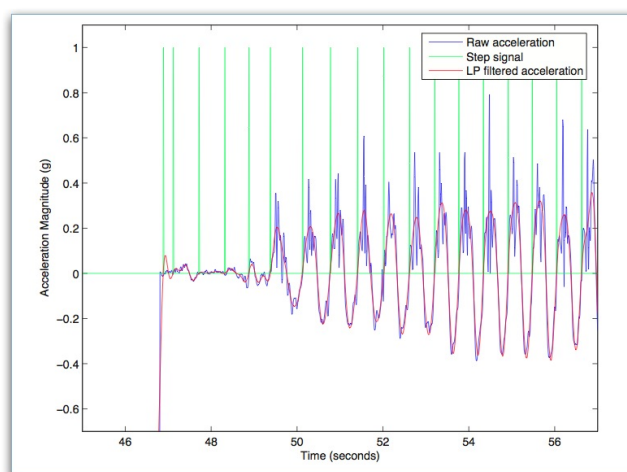


Figura 2.2: Método de deteção de passos (exemplo) [2].

A técnica mais elementar apresenta a sua base no algoritmo proposto por uma patente de 1996 [33]. Este algoritmo consiste na identificação do máximo do sinal do acelerómetro, dentro de um intervalo fixo de tempo. O valor mínimo para que

esse máximo possa ser considerado um passo, e o intervalo mínimo de tempo entre passos consecutivos, são algumas restrições adicionais que ajudam na prevenção de possíveis falsos positivos (muito frequentes em dispositivos *handheld*). Outros estudos proporcionam algumas ideias para melhorar os seus resultados, através de processos de pré-processamento do sinal, como a utilização de filtros passa-baixo no sinal do acelerómetro (ver exemplo na Figura 2.2) [2]. Assim, parte do ruído indesejável é eliminado a partir das leituras do sensor.

Estimativa do comprimento do passo

O comprimento do passo pode ser definido como a distância entre dois impactos sequenciais provocados pelo calcanhar. Não se apresenta como um valor constante, e é afetado por diversos fatores como: o ritmo de passada do utilizador, as situações de subida ou descida de escadas, mudanças de direcção e a inclinação do terreno onde o pedestre se desloca. Assim sendo, este parâmetro deve ser determinado continuamente durante a caminhada, de forma a aumentar a precisão [26].

Entre os inúmeros estudos realizados, cada autor parece apresentar um método diferente para o cálculo do comprimento do passo. Contudo, uma característica comum parece ser, em quase todos os métodos, o facto de a média do comprimento do passo do utilizador ser conhecida.

Na proposta de [2], uma rede neuronal treinada é utilizada para traduzir a relação entre as variações dos acelerómetros e o comprimento do passo. Um algoritmo de calibração do passo em tempo-real foi proposto em [34], usando um filtro de *Kalman*, com medidas de posicionamento GPS e três parâmetros estimados: frequência, amplitude e média. Na referência [33], o comprimento do passo é estimado com base numa relação linear entre a medida da frequência do passo e o comprimento passo (p.ex., maior ritmo implica normalmente maior comprimento de passo). No artigo [32], o método proposto consiste na utilização do cálculo do integral duplo da aceleração em conjunto com a informação sobre o tipo de movimento.

Determinação da orientação

A precisão da orientação pode afetar bastante o desempenho de um sistema

PDR. Afim de aumentar a precisão da bússola tradicional, alguns métodos têm sido propostos combinando este sensor com outros.

Em [2], é proposta uma combinação de medições de uma bússola magnética e cursos de GPS, ao longo do terreno. Estes sinais foram utilizados individualmente quando disponíveis, e nenhuma tentativa foi efetuada para uni-los sistematicamente. Esta combinação ajudou nos casos em que o pedestre dava passos laterais, por exemplo, uma vez que a bússola proporciona a orientação e não necessariamente a direção de deslocação.

Outro método possível é a combinação de uma bússola magnética com um giroscópio [32]. Nesta proposta, a fonte principal de informação da orientação é o giroscópio, e a bússola é apenas usada como uma medida de referência. A combinação dos dados obtidos pelos dois sensores é realizada através da utilização de um filtro de *Kalman*.

2.1.1.3 Variante: *Map-Aided* (uso de plantas dos edifícios)

Os sistemas *Map-Aided Positioning (MAP)* são uma variante dos sistemas de localização *indoor* no geral. Esta abordagem advém dos já conhecidos *Personal Navigation Assistants (PNA)* para a navegação automóvel [35]. Atualmente, a maioria dos edifícios públicos já possuem as suas plantas em formato digital. De acordo com a pesquisa bibliográfica realizada, nenhum formato *standard* foi encontrado relativo aos mapas dos edifícios, portanto o método de conversão depende de caso para caso, depende do formato digital da planta, e claro, do formato do mapa desejado [24]. A Figura 2.3 ilustra um diagrama de blocos geral representativo do processo de *map matching*.

Nos sistemas INS, as plantas dos edifícios são usadas como medida adicional de redução dos problemas de precisão dos sensores e dos erros cumulativos. Esta abordagem apresenta muitas vantagens, visto que aumenta a precisão do sistema sem necessidade de qualquer mudança no edifício, nem o uso de outros sensores.

A integração destas plantas, de forma a auxiliar a precisão de um INS, pode ser efetuada de várias formas:

- simples introdução de restrições, como a impossibilidade do atravessamento de paredes e outras heurísticas [36];

- aplicação de algoritmos específicos para a estimativa condicionada do estado de um sistema, como por exemplo, filtros de partículas [29].
- criação de um grafo representativo dos trajectos principais, que possa traduzir eficazmente o posicionamento do sujeito [37].



Figura 2.3: Diagrama representativo do processo *map matching* (adaptada de [3]).

Muito do estudo e trabalho desenvolvido nesta variante é também aplicável a outros sistemas de localização *indoor* como, por exemplo, os que usam *Wireless Local Area Network (WLAN)*. Vários artigos publicados estão direcionados para estes métodos, e não propriamente para o *PDR*, contudo, uma vez que o *PDR* é um sistema que acumula erro, o uso desta variante de correção assume ainda uma maior importância.

2.1.1.4 Vantagens e Desvantagens

Em suma, as principais vantagens e desvantagens desta técnica podem ser destacadas.

Vantagens

- ◇ Método adaptável a uma diversidade de contextos;
- ◇ Técnica não intrusiva, i.e. não requer alterações do espaço;
- ◇ Existe já um considerável leque de investigação com inúmeras abordagens diferentes;
- ◇ As suas variantes (p.ex., o uso de plantas dos edifícios) beneficiam de informação normalmente já existente.

Desvantagens

- ◇ Existência de erro acumulado;
- ◇ Impossibilidade de determinação de alguns passos (o uso de elevadores, por exemplo);
- ◇ Volatilidade do sistema face a interferências magnéticas;
- ◇ Diferente posicionamento dos dispositivos podem afetar algumas das abordagens.
- ◇ Complexidade de alguns dos conceitos utilizados para gerir incertezas (Filtros de *Kalman*, por exemplo).

2.1.2 Rádio Frequência (RF)

Família de métodos de localização *indoor* que utilizam uma ou mais propriedades das ondas electromagnéticas para identificar o posicionamento de uma pessoa ou objeto. As ondas rádio conseguem "viajar" facilmente através de paredes e corpos humanos, e como tal, estes sistemas de posicionamento apresentam uma área de cobertura maior e requerem menos *hardware*, comparando com outros sistemas. Técnicas de triangulação e *fingerprinting* são amplamente utilizadas em sistemas de posicionamento baseados em RF. Em ambientes *indoor* complexos, a localização *fingerprinting* (i.e., com mapeamento prévio do espaço) apresenta-se como um método eficaz para estimar a posição, o qual usa características relativas a localização tais como: o *Received Signal Strength (RSS)* e a informação de localização dos transmissores, para determinar a posição de um utilizador ou dispositivo [15].

Dentro de todas as tecnologias que se englobam neste grupo, a mais comum e já muito estudada é a **WLAN** (IEEE 802.11, também conhecida como *Wi-Fi*). Contudo, novas aproximações tem sido desenvolvidas com recurso ao *Bluetooth*, à banda ultralarga (**UWB**), ao **RFID**, ou ao **NFC**. Estes métodos podem se dividir em duas classes: uns baseados no método da triangulação (*Wi-Fi*, *GPS+beacon* e *UWB*) e outros relativos a proximidade (*RFID*, *Bluetooth* e *NFC*) (Figura 2.4).

A precisão alcançada varia de acordo com a tecnologia usada. Segundo estudos nesta área é possível alcançar uma precisão na ordem dos 2-3 metros com estes métodos [14, 38, 39]. Para determinar a orientação de uma pessoa ou objeto, devem ser utilizados instrumentos auxiliares como a bússola ou o giroscópio. Acrescenta-se que estes sistemas podem também beneficiar do uso das plantas de edifícios, como por exemplo, pode ser verificado em [40].

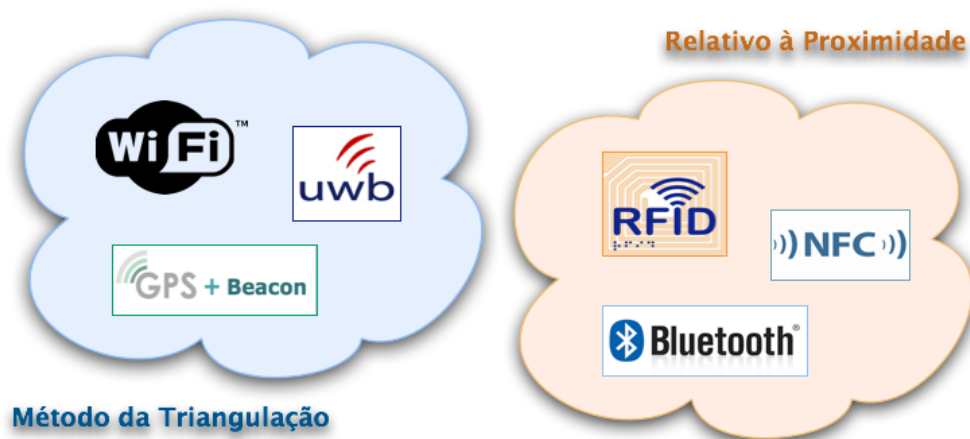


Figura 2.4: Divisão das tecnologias rádio frequência por classes.

Wi-Fi

Wireless Fidelity (Wi-Fi) é o nome comum para o IEEE 802.11 *standard*. Atualmente, a conectividade *wireless* encontra-se mais prevalente do que nunca na nossa vida quotidiana, existindo em quase todos os lugares (p.ex., universidades, hospitais, centro comerciais, museus, e aeroportos). É amplamente utilizada e integrada em inúmeros dispositivos *wireless*, como PDAs, computadores, e telemóveis. Portanto, torna-se prático e de baixo custo o uso de pontos de acesso para determinar a localização do utilizador em ambientes fechados [15]. A técnica baseada em *fingerprinting* é a mais utilizada nestes sistemas, onde um mapa dos pontos fortes do sinal recebido é aplicado para inferir a posição, com base na comparação entre a medição atual da potência do sinal e o respetivo mapa. São diversos os sistemas de localização que utilizam a tecnologia Wi-Fi [41, 42, 43, 44, 45].

Um ponto de acesso Wi-Fi transmite continuamente um sinal e cada dispositivo na vizinhança imediata recebe este sinal. O sinal inclui o *Media Access Control address (MAC address)* que identifica exclusivamente o ponto de acesso e, além disso, todos os dispositivos ao alcance são capazes de medir a potência do sinal do ponto de acesso, conhecido como *Received Signal Strength Indicator (RSSI)* [46]. Quanto maior for a potência do sinal, mais precisa vai ser a posição estimada

A precisão depende de alguns fatores, como por exemplo, o número de pontos de acesso, a propagação *multipath*, ou até mesmo de interferências por outras técnicas. Tipicamente, nestes sistemas a precisão é aproximadamente de 3 a 30 metros, com uma taxa de atualização na gama de alguns segundos [38]. Wi-Fi pode também ser utilizado em combinação com outras tecnologias de sensoriamento, de forma a produzir resultados de posicionamento mais precisos e contínuos.

GPS + Beacon

A utilização conjunta de diferentes tecnologias, de forma a melhorar os resultados obtidos, é uma questão pertinente a ter em conta.

Recentemente, a Nokia Research Center² tem vindo a trabalhar num conceito de navegação *indoor* que conduz a navegação estilo GPS a um nível superior. O sistema, designado de *Nokia Indoor Navigation*, tem como base *beacons* de posicionamento (ver Figura 2.5) que atuam como "satélites", mas apenas para interiores, e encontram-se colocados no teto do edifício. Em conjunto com o *software* integrado no dispositivo Nokia, o sistema é capaz de identificar exatamente a posição atual, dar instruções para pontos específicos e localizar o quão longe a pessoa está do destino desejado. Tudo isto em tempo real e com uma precisão até 30 cm [4].

O sistema pode também ser aplicado para manter o controlo dos pertencentes do utilizador. Colocando um dispositivo especial de rastreio na mala do utilizador, por exemplo, o sistema irá alertar se o utilizador o deixar em algum lugar do edifício, e exatamente onde o deixou (ver Figura 2.6). Esta característica do sistema torna-se útil se o respetivo utilizador tiver o mau hábito de perder os seus pertencentes.

²research.nokia.com/news/12425

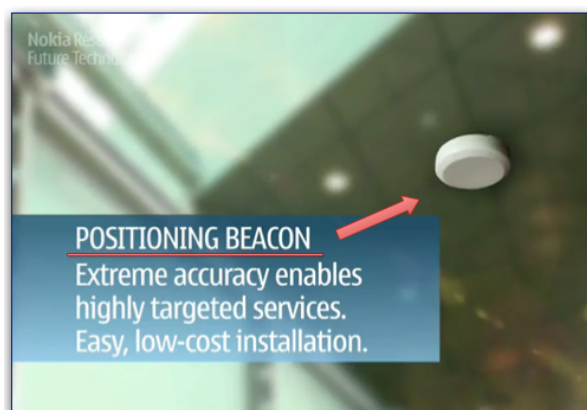


Figura 2.5: *Beacon* de posicionamento *indoor* (adaptada de [4]).

Existem diversos cenários onde esta tecnologia poderá ser muito útil, como locais públicos, edifícios de escritórios, entre outros. Embora a Nokia reporte que os *beacons* sejam de baixo custo, a dependência permanece para a implantação de uma infra-estrutura adicional [4]. Uma grande vantagem do sistema é o facto de este ser capaz de comutar automaticamente do modo *outdoor* para o modo de navegação *indoor*, possibilitando assim experiência ao utilizador para facilmente transferir-se entre os dois modos de navegação. Nokia (Navteq³) já possui mapas de shoppings e aeroportos, portanto, a localização e navegação *indoor* é uma progressão natural.

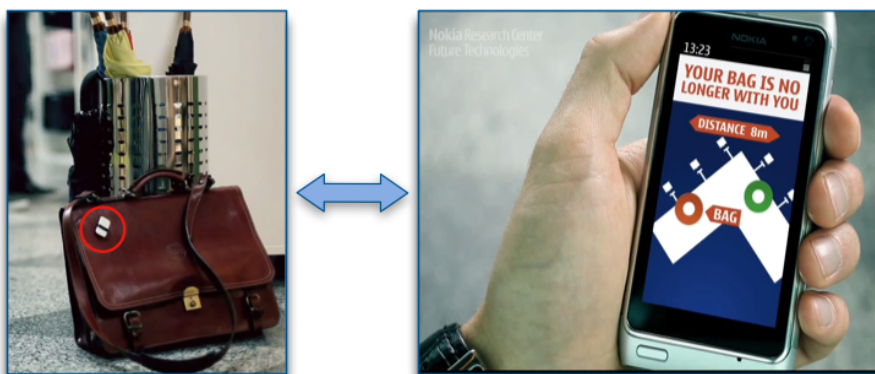


Figura 2.6: Exemplo da localização de um objeto perdido: mala representada pelo círculo vermelho e o utilizador pelo círculo verde (adaptada de [4]).

³www.navteq.com

UWB

Os sinais *Ultra Wide Band (UWB)* usados para posicionamento estão recentemente a receber cada vez mais atenção, devido à sua capacidade de fornecer informação de posicionamento com uma precisão na escala dos centímetros [47].

A tecnologia **UWB** consiste essencialmente na transmissão e recepção de pulsos ultra-curtos de energia electromagnética. Os pulsos **UWB** caracterizam-se por uma transmissão de muito baixa energia (< 10 microwatts) e uma largura de banda ampla (>1 Gigahertz) [48]. Esta característica torna possível filtrar os sinais refletidos do sinal original, i.e. minoriza os efeitos da NLOS⁴, o que proporciona uma maior precisão e uma maior confiabilidade. A utilização de uma grande variedade de componentes de frequência aumenta a probabilidade de um sinal passar em torno de um obstáculo, oferecendo maior resolução. Além disso, o sistema está sujeito a menos interferências de outras frequências de rádio que se encontram em utilização na área. Acrescenta-se que os sensores **UWB** são baratos, o que torna o sistema de posicionamento numa solução económica [15].

Associada a outras técnicas de posicionamento, como o **GPS** ou o **GNSS**, o **UWB** pode proporcionar uma localização contínua do exterior para a posição no interior, e vice-versa.

RFID

O *Radio-Frequency Identification (RFID)* é um meio de armazenamento e recuperação de informação através de transmissão eletromagnética para um circuito integrado compatível com rádio-frequência, e recentemente, é visto como um meio de melhorar os processos de tratamento de dados [15].

Os sistemas de posicionamento **RFID** são normalmente aplicados em ambientes *indoor* complexos, como escritórios e hospitais. Como uma tecnologia sem fios, o **RFID** proporciona uma identificação flexível e barata do utilizador ou do dispositivo, com aproximadamente uma precisão de 1 a 3 metros [14]. Esta tecnologia pode substituir técnicas de identificação, como os códigos de barras, e ser utilizada para a conceção de diversos produtos e serviços [15]. Alguns sistemas populares de localização que usam a tecnologia **RFID** são SpotON [49] e LAND-MARC [50].

⁴ *Non Line-of-Sight*, que significa nenhuma exigência de linha de visão.

A principal vantagem do uso do **RFID** é o facto de este não necessitar nem de um contato direto nem de uma linha de visão (*line of sight*) direta com os objetos de posicionamento. Outra vantagem é o baixo custo de instalação para áreas pequenas, contudo, para grandes áreas, os custos de implementação e substituição do equipamento já aumenta [51]. Por outro lado, uma desvantagem destes sistemas, é que estes sofrem da distorção *multi-path* dos sinais rádio refletidos por paredes em ambientes *indoor*.

Os sistemas **RFID** apresentam vários componentes básicos, incluindo um número de *readers* e *tags* **RFID**, e a comunicação entre eles. Os *readers* lêem os dados emitidos a partir das *tags*, e ambos os componentes, para transmitir e receber dados, usam uma frequência rádio definida e um protocolo. As *tags* **RFID** são caracterizadas como passivas ou ativas. As *tags* passivas são recetores, operam sem bateria, são mais leves, mais pequenas e mais baratas, contudo, a sua área de cobertura é pequena. Já as *tags* ativas são transdutores (*transceivers*), que transmitem ativamente o seu ID e outra informação adicional. O custo destas *tags* é mais elevado, mas por outro lado, a sua área de cobertura é maior [15, 38].

Bluetooth

O *Bluetooth* (o IEEE 802.15.1 standard) caracteriza-se como uma simples e segura tecnologia de comunicação de curto alcance, e encontra-se em toda a parte. A tecnologia *Bluetooth*, similar ao Wi-Fi, tem sido implementada em inúmeros tipos de dispositivos, tais como telemóveis, computadores, câmaras, PDAs, entre outros. Destina-se a substituir os cabos de conexão de dispositivos, mantendo níveis altos de segurança. As principais características desta tecnologia são a robustez, baixo consumo de energia, e baixo custo. A especificação *Bluetooth* define uma estrutura uniforme para um vasto leque de dispositivos para se poderem conectar e comunicar uns com os outros [52]. Esta tecnologia funciona na frequência de banda *Industrial Scientific and Medical (ISM)* de 2.4 GHz, a máxima distância de comunicação é até 100 metros (*Bluetooth 2.0 Standards*), e os dispositivos conseguem enviar um máximo de 3Mb por segundo [15, 38].

Uma desvantagem dos sistemas de posicionamento baseados em *Bluetooth* é o facto do sistema poder apenas proporcionar uma precisão de 2 a 3 metros, com

um atraso de cerca de 20 segundos [15]. A sua implementação pode ser de elevado custo. Posicionamento sem fio usando *Bluetooth* tem sido recentemente estudado [53, 54].

NFC

Near Field Communication (NFC) consiste numa tecnologia sem fios de curto alcance que permite a comunicação entre objetos com a mesma tecnologia, por uma distância de menos de 10 centímetros. O NFC é baseado em padrões de RFID, e é projetado para fazer um mundo mais fácil e mais conveniente, melhorando a forma de fazer transações bancárias, trocar conteúdo e conetar dispositivos [55]. Esta técnica é similar ao *Bluetooth*, pois ambas possibilitam a comunicação sem fio e a troca de dados entre dispositivos. Contudo, NFC oferece mais segurança do que o *Bluetooth*, e não requer parelhamento dos dispositivos, funcionando só com um simples "toque".

Dispositivos que utilizam NFC podem operar em modo ativo ou passivo. Um dispositivo passivo, como uma *tag* NFC, contém informação que outros dispositivos podem ler, mas não lê essa mesma informação, apenas a transmite. Estas *tags* podem ser incorporadas em produtos, adesivos, cartões de visitas e pulseiras, por exemplo. Dispositivos ativos podem ler as informações e transmiti-las. Por exemplo, um *smartphone*, não só seria capaz de recolher a informação das *tags* como também de trocar informações com outros dispositivos compatíveis, e até mesmo alterar a informação da *tag*, se autorizado para tal [55].



Figura 2.7: Áreas de potenciais usos da tecnologia NFC (adaptada de [5]).

Embora o **NFC** seja uma técnica intuitiva, segura e versátil, ainda apresenta algumas desvantagens, como por exemplo: a falta de conhecimento, pois é uma tecnologia relativamente desconhecida, embora já exista há algum tempo; e limitações técnicas, como limite de velocidade e da quantidade de dados que se pode transmitir.

Diversas são as áreas de potenciais usos desta tecnologia (ver Figura 2.7), sendo as principais: controlo de acesso, saúde, recolha e troca de informação, cartões de fidelidade, pagamentos e transporte. Estudos recentes nesta área têm sido feitos e propostas para sistemas de navegação *indoor* usando a tecnologia **NFC** têm surgido. Um dos exemplos é o sistema chamado de *NFC Internal* [56], que permite ao utilizador determinar a sua posição atual dentro de um edifício apenas tocando com o dispositivo móvel nas *tags* que se encontram distribuídas pelo edifício.



Figura 2.8: Exemplos da aplicação da tecnologia **NFC** (adaptada de ^{5 6 7} e [6]).

Atualmente, a maior força do **NFC** encontra-se nas mãos do **Google Wallet**⁵, serviço de pagamentos do Google que substitui a utilização de cartões de crédito e débito, por *smartphones* e leitores com o **NFC** integrado. Em Londres, existe o exemplo do **Oyster Card**⁶, cartão **NFC** que é lido pelas portas ao entrar e sair das estações de metro e também em autocarros. Em países como o Japão, já é possível conferir esta tecnologia sendo utilizada no dia a dia, por exemplo, em

⁵www.google.com/wallet/index.html

⁶www.nfcworld.com/2011/07/12/38537/transport-for-london-to-accept-nfc-payments-from

sistemas de metro ou comboio (**Strappy**⁷), onde permite que se comprem viagens apenas com a aproximação do telemóvel aos respectivos dispositivos, e também em postos de venda [6]. Estes exemplos da aplicação do **NFC** encontram-se ilustrados na Figura 2.8.

Muitos acreditam que a tecnologia **NFC** deve substituir os códigos de barras e até mesmo os cartões de crédito. Assim, objetos comuns do quotidiano transformam-se em "objetos inteligentes", capazes de armazenar e transmitir informações.

2.1.2.1 Vantagens e Desvantagens

Em suma, os principais prós e contras da família de métodos RF são destacadas.

Vantagens

- ◇ O uso de uma infra-estrutura normalmente existente em grande parte dos edifícios (no caso do Wi-Fi);
- ◇ Sistemas que apresentam uma área de cobertura maior e requerem de menos *hardware* (comparando com outros sistemas);
- ◇ Diverso trabalho científico já desenvolvido e algumas implementações comerciais realizadas.

Desvantagens

- ◇ Problemas com reflexões e atenuações de sinal (efeitos NLOS);
- ◇ Necessidade de alteração do espaço com a implementação de novas redes de RF (nos casos em que a tecnologia adotada não está ainda implementada);
- ◇ Restrições existentes à propagação de ondas de RF em determinados locais (p.ex., Hospitais);
- ◇ Possíveis conflitos entre bandas de frequência do espectro;
- ◇ Uma disposição dos emissores que não tenha em conta as condicionantes física do espaço pode influenciar negativamente o cálculo do posicionamento.

⁷www.shifteast.com/nfc-comes-to-tokyo-trains-with-strappy

2.1.3 Análise de Imagem

O uso de sistemas de posicionamento com base na análise de imagem pretende, à semelhança do que é feito pelo ser humano, determinar a sua posição através do estudo visual do espaço à sua volta, utilizando como referência características do ambiente (naturais ou artificiais) que nos são conhecidas. Dependendo da abordagem escolhida e das condições do ambiente (p.ex., iluminação ou oclusões), este tipo de soluções pode proporcionar uma precisão na ordem dos centímetros ou até mesmo de alguns poucos metros [14] [7].

Os métodos que envolvem análise de imagem podem-se caracterizar em dois grupos: (1) os que procuram identificar e acompanhar o movimento de determinadas características naturais da imagem, aquando do deslocamento da pessoa, e com base nisso tirar conclusões sobre o posicionamento; (2) os que usam a simples leitura de marcas fiduciais, introduzidas intencionalmente no ambiente, e que apresentam uma localização específica associadas a si.

Reconhecimento de Características Naturais

Neste tipo de métodos é efetuada a análise das características naturais de uma imagem (com recurso a câmaras tradicionais ou omnidirecionais), sem a necessidade de introduzir objetos artificiais para auxiliar no posicionamento da pessoa.

Dentro destes métodos existem ainda dois casos distintos: os casos em que as câmaras apresentam uma posição fixa; e aqueles em que a câmara é móvel sendo normalmente transportada pela pessoa. Ambas as abordagens seguem tipicamente um modelo bem definido organizado por quatro fases [57]:

1. Aquisição da imagem a partir da posição atual da pessoa;
2. Segmentação da imagem e extração de características;
3. Correspondência da imagem obtida à representação do espaço armazenada em memória;
4. Computação da posição e orientação da câmara que originou a imagem.

Para o caso de câmaras fixas, existem já algumas soluções implementadas, como por exemplo a do projeto *Easy Living* da Microsoft [58]. O sistema determina a localização e a identidade de várias pessoas, num típico ambiente sala de estar, utilizando dois conjuntos de câmaras estéreo a cores integradas nas paredes da sala. As imagens estéreo são utilizadas para localizar as pessoas, e as imagens a cores usadas para manter as suas identidades. A precisão do sistema não pode ser garantida, por ser variável devido a várias fontes de interferência (como luz e o tempo). Por exemplo, o simples ato de ligar e desligar a luz de uma sala reduz a precisão do controlo do movimento de uma pessoa. Uma desvantagem deste sistema é o facto de este utilizar várias câmaras para cobrir todos os cantos e oclusões do respetivo espaço *indoor*, e como tal, torna-se uma solução dispendiosa. Por outro lado, a não necessidade do utilizador transportar qualquer dispositivo de localização, caracteriza-se numa grande vantagem.

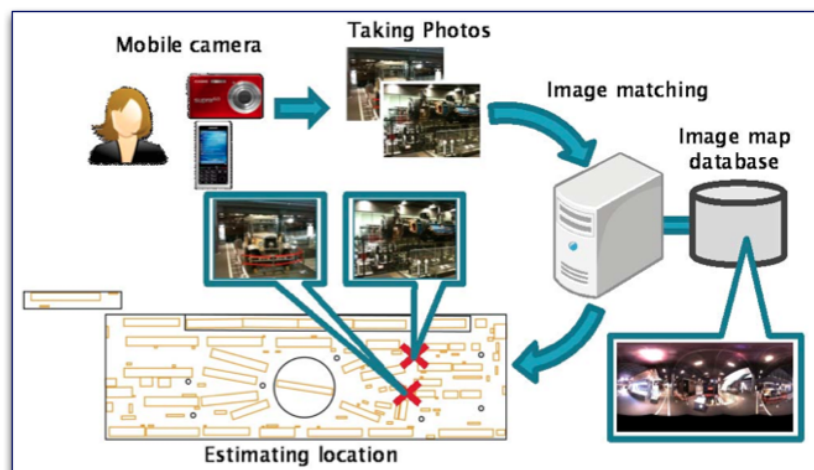


Figura 2.9: Exemplo do sistema de posicionamento *indoor* aplicado em [7].

No caso da utilização de câmaras móveis (ver exemplo na Figura 2.9), o processo aplicado na localização divide-se em duas fases distintas. Uma primeira fase consiste na identificação das diferentes características do espaço (p.ex., através de SIFT [59] ou SURF [60]), conhecidas por *feature points*, para posterior mapeamento da representação visual destas à sua localização. Esta estrutura é então guardada, juntamente com a orientação e posição da câmara utilizada aquando da captura da imagem.

Numa fase posterior, as imagens adquiridas a partir da câmara são comparadas com as armazenadas durante a primeira fase do método. Aquando desta fase, as imagens sofrem algumas transformações, de forma a testar diferentes possibilidades de posição, orientação ou iluminação da imagem até que uma correspondência significativa entre as componentes de cada uma das imagens seja encontrada. Para este processo podem ser aplicadas várias abordagens, desde o *Straight-line Matching* [57] ao método do vizinho mais próximo (*nearest neighbor*) [7]. Características como a cor ou a forma das componentes são utilizadas aquando da fase de *tracking*.

Recentemente, têm surgido várias propostas nesta área [61] e [62].

Alguns estudos reportam a utilização de duas câmaras como responsável pela melhoria considerável da precisão do sistema [63].

Marcas Fiduciais

Uma questão relativamente diferente é a relacionada com a deteção de marcas fiduciais. Marcas fiduciais são imagens artificiais adicionadas ao ambiente real de modo a auxiliar a localização da pessoa, a partir da identificação de padrões de mais fácil análise do que as características naturais da imagem [64]. As cores e as letras utilizadas em alguns parques de estacionamento são um exemplo de marcas fiduciais que ajudam à navegação *indoor*. É importante que as características visuais estejam claramente visíveis, de forma a que o processo de rastreamento seja efetuado com sucesso. Segundo [65], uma marca fiducial para ser eficaz deve seguir determinados critérios, como:

- * Suportar uma determinação inequívoca da orientação e posição relativa da câmara que capturou a imagem;
- * Não favorecer uma orientação em relação a outra;
- * Não ser facilmente confundida com outra marca fiducial do mesmo conjunto;
- * Ser facilmente identificada por algoritmos rápidos e simples;
- * Ser detetável ao longo de um largo intervalo de captura (da câmara).

As características visuais mais relevantes nestas marcas encontram-se associadas com as cores e os formatos utilizados (formas geométricas, como retângulos e círculos, são os mais comuns) [64]. Cada uma destas marcas fiduciais deve apresentar um identificador único (ID), que é utilizado para encontrar a posição da imagem após esta ter sido introduzida no ambiente.

Em [66] é proposto um sistema de localização *indoor* eficiente e de baixo custo, que utiliza uma *web* câmara e marcas fiduciais (com o alfabeto nelas). Para o reconhecimento do carater (letra do alfabeto) na marca é utilizada uma rede neuronal artificial.

(i) QR-codes como marcas fiduciais?!

QR ou *Quick Response Codes* são um tipo de código de barras 2D utilizados para codificar e descodificar dados a grande velocidade.

Foram originalmente desenvolvidos no Japão, em 1994 por *Denso Wave*, com o propósito de serem utilizados no rastreio de peças na indústria automóvel [67]. Desde então, o uso de *QR-codes* tem crescido largamente para um ponto em que é cada vez mais comum encontrar *QR-codes* em qualquer lugar: em revistas (tanto em anúncios como artigos), jornais, cartões de empresas, em produtos de consumo do dia a dia como uma simples embalagem de champô, e até mesmo em cartazes na rua ou nas paredes. Tudo o que é preciso para aceder à informação armazenada nestes códigos é um telemóvel com câmara e uma aplicação de leitura de *qr-codes*. A Figura 2.10 ilustra um exemplo de captura de um *QR-code* e a respetiva descodificação da informação através de uma aplicação *android*.

Um *QR-Code* é um código matriz que foi concebido principalmente para ser um símbolo que é facilmente interpretado por câmaras. Contém informação em ambas as direções (vertical e horizontal), e armazena um volume de informação consideravelmente maior que os códigos de barras tradicionais. São capazes de lidar com vários tipos de dados, como números, caracteres alfanuméricos, caracteres japoneses (Kanji/Kana,), símbolos e binários [67]. Portanto, quando capturados, estes códigos podem, entre outras coisas, fornecer informação de um contato de uma pessoa ou empresa, enviar uma mensagem de texto a pedir ajuda ou informações adicionais, e direcionar para *links* de sites que contêm mais informação sobre

um produto ou serviço, ou mesmo *links* de sites pessoais. Este tipo de códigos têm também a capacidade de correção de erros, assim, os dados podem ser recuperados mesmo quando partes substanciais dos códigos são distorcidas ou danificadas.

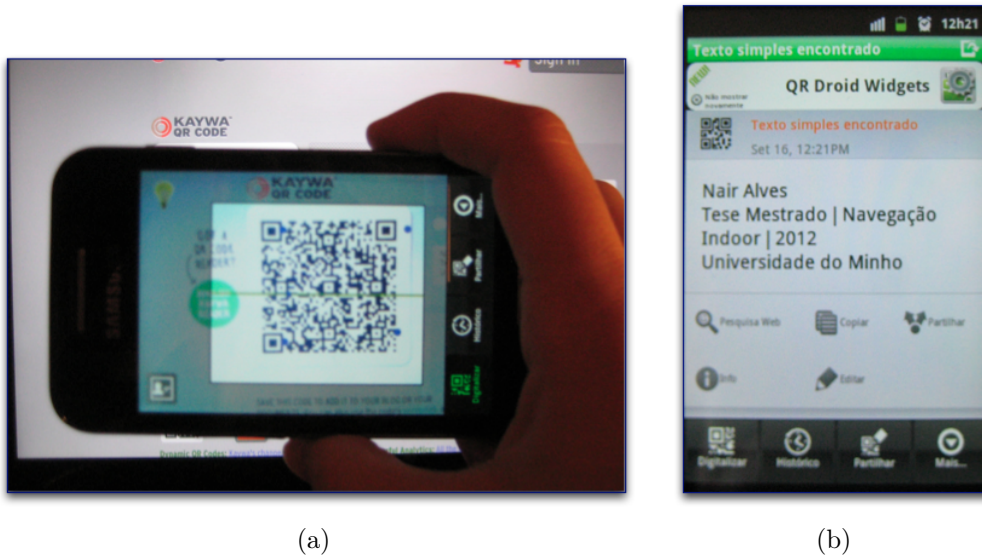


Figura 2.10: Exemplo de um *QR-Code*: (a) Fase de captura do *qr-code*; (b) Decodificação da informação subjacente ao *qr-code* em a).

Todas estas características tornam os *QR-codes* num tipo de codificação cada vez mais popular, e portanto, muito apelativo para os sistemas de localização *indoor*.

Como descrito em [19], a aplicação de *QR-codes* é uma boa escolha para os *LBS indoor* devido à sua rápida legibilidade. Com o auxílio da informação obtida a partir de um *qr-code*, uma aplicação *LBS* pode localizar um utilizador de forma simples. Este tipo de códigos é facilmente introduzido em qualquer parte de um edifício, com um custo de implementação bastante baixo, e desta forma, ter-se uma grande área cobertura. Pode também ser utilizado em paralelo com outras tecnologias (p.ex., sensores de *smartphones*) para continuamente localizar um utilizador, e a sua precisão no posicionamento não é afetada pelo meio ambiente. Uma desvantagem da utilização destes sistemas de localização é facto de existir a necessidade de deslocamento por parte do utilizador até muito perto da marca fiducial, para uma melhor linha de visão.

2.1.3.1 Vantagens e Desvantagens

Em suma, as principais vantagens e desvantagens dos sistemas baseados em análise de imagem podem ser destacadas.

Vantagens

- ◇ Elevada precisão;
- ◇ Ausência de necessidade de alteração do ambiente (no caso do reconhecimento de características naturais);
- ◇ Uso de tecnologias existentes atualmente nos dispositivos móveis (câmara).

Desvantagens

- ◇ Necessidade de alteração do ambiente (no caso das marcas fiduciais);
- ◇ Dificuldades com a imprevisibilidade do ambiente (p.ex., oclusões temporárias);
- ◇ Influência das condições de iluminação na qualidade dos resultados;
- ◇ Necessidade de deslocamento até à marca fiducial (no caso destas serem usadas apenas para *close range*).

Síntese

Ao longo desta secção foram apresentados inúmeros métodos que solucionam o problema da localização *indoor*. Além dos métodos introduzidos, existem ainda outros dentro do contexto da localização *indoor* que, devido às suas características, não se apresentam tão indicados para o problema em questão. Destes, são exemplo os métodos que utilizam sinais ultra-sons e infra-vermelhos, que apesar do seu alcance reduzido apresentam precisões elevadas na ordem dos centímetro [14].

Observando a Tabela 2.1, que apresenta o resumo das principais características de cada método, pode-se verificar que, apesar de cada proposta de solução ter as suas particularidades, não existe nenhuma que se destaque claramente em relação às restantes. A chave para resolver esta questão encontra-se na combinação de diferentes técnicas de posicionamento. Portanto, a escolha do melhor método está largamente influenciada pelos requisitos e circunstâncias de cada caso.

Tabela 2.1: Quadro de comparação das tecnologias abordadas durante a secção.

	<i>Precisão</i>	<i>Vantagens</i>	<i>Desvantagens</i>
PDR	> 95% do trajeto	<ul style="list-style-type: none"> ○ Adaptável; ○ Não necessita de alteração do espaço. 	<ul style="list-style-type: none"> ○ Erro acumulado; ○ Precisão dependente de diversos fatores.
Wi-Fi	3 m	<ul style="list-style-type: none"> ○ Aproveita uma arquitetura já existente; ○ Baixo custo. 	<ul style="list-style-type: none"> ○ Problemas em lidar com NLOS; ○ Problemas com a propagação <i>multipath</i>.
GPS + Beacon	30 cm	<ul style="list-style-type: none"> ○ Comutação automática entre <i>outdoor</i> e <i>indoor</i>; ○ Dispositivo adicional para rastreio de objetos perdidos. 	<ul style="list-style-type: none"> ○ Algum custo na implementação.
UWB	na ordem dos cm	<ul style="list-style-type: none"> ○ Elevada precisão; ○ Baixa energia; ○ Largura da banda ampla. 	<ul style="list-style-type: none"> ○ Custo elevado de implementação; ○ Não é usual.
RFID	1-3 m	<ul style="list-style-type: none"> ○ Baixo Custo 	<ul style="list-style-type: none"> ○ Curto Alcance; ○ NLOS; ○ Intrusiva.
Bluetooth	2-3 m	<ul style="list-style-type: none"> ○ Baixo custo; ○ Baixo consumo de energia. 	<ul style="list-style-type: none"> ○ Elevado custo de implementação.
NFC	na ordem dos cm	<ul style="list-style-type: none"> ○ Versátil; ○ Seguro; ○ Intuitivo. 	<ul style="list-style-type: none"> ○ Pouco conhecida ainda; ○ Limite de velocidade e na quantidade de dados.
Análise Imagem (Carat.Nat.)	na ordem dos cm	<ul style="list-style-type: none"> ○ Orientação e localização precisas; ○ Sem necessidade de alteração do ambiente. 	<ul style="list-style-type: none"> ○ Complexo; ○ Vulnerável a alterações do ambiente.
Marcas Fiduciais	na ordem dos cm	<ul style="list-style-type: none"> ○ Orientação e localização precisas; ○ Mais simples que a análise das características naturais. 	<ul style="list-style-type: none"> ○ Necessidade de alteração do espaço envolvente; ○ Vulnerável às condições do ambiente.

2.2 Representação dos Espaços

O primeiro passo para uma boa navegação dentro de um edifício consiste numa boa representação topológica do mesmo [68]. Quanto mais precisa for a informação fornecida, mais precisos serão os resultados proporcionados pela aplicação de navegação. Existem representações com base na planta do edifício (*floor footprints*) e representações baseadas no modelo do edifício. Ambas podem ser orientadas sintática ou semanticamente. A distinção entre espaços *indoor* e modelos 3D é necessariamente discutida, visto que a distinção entre eles não se encontra, de certa forma, bem definida.

Os conteúdos da informação que suportam a navegação *indoor* 3D devem incluir a planta de cada piso, a descrição semântica de cada componente do edifício, registos da posição do pedestre, posição das saídas do edifício, entre outros. A informação de controlo da navegação deve incluir os elementos topológicos (que definem nodos e arestas) para análise da rede, e os elementos geométricos (que definem os componentes de visualização) para a representação da rota [68].

2.2.1 Como obter os dados 3D

Comparando com as plantas 2D, os modelos 3D de edifícios são uma forma mais poderosa e com mais capacidade para representar um edifício, uma vez que estes armazenam muito mais informação. Contudo, a complexidade associada a um 3D *Building Information Model (BIM)* é a sua maior desvantagem. Os sistemas de informação geográfica 3D, conhecidos como uma ferramenta ideal para representar a geometria 3D, semântica assim como topológica, têm vindo gradualmente a ser utilizados em diversas áreas [68].

Nos últimos tempos, é possível encontrar cada vez mais modelos 3D. Existem basicamente duas formas de os adquirir: ou recorrendo às ferramentas típicas de desenho (AutoCAD⁸, Blender⁹ e Revit¹⁰, por exemplo); ou gerar os modelos a partir de imagens dos edifícios. Na primeira opção, incluem-se os modelos gerados por profissionais como arquitetos e designers. Receber os BIM de quem construiu

⁸usa.autodesk.com/autocad

⁹www.blender.org

¹⁰usa.autodesk.com/revit

o edifício seria o ideal, pois além da forma tem-se também alguma semântica, como as funcionalidades dos espaços. Para além destes profissionais, existem também ferramentas simples e *open source*, como o **Sketchup**¹¹ e o **BuildingMaker**¹², que permitem criar modelos de uma forma mais intuitiva e, eventualmente, menos rigorosa. Na ausência da informação anterior, podem-se aplicar técnicas de levantamento a partir de imagens dos edifícios, onde será preciso obter a forma, e posteriormente, complementar com alguma semântica manualmente. Dado que se estudaram várias técnicas dentro desta classe de algoritmos, as mesmas serão apresentadas mais à frente, no Capítulo 3.

2.2.2 Conetividade (suporte ao routing)

Na literatura, os termos "grafo" e "rede" são aplicados no mesmo contexto, isto é, como um modelo de conetividade composto por arestas e nodos [69]. Esta visualização nodo/aresta da rede apresenta uma vantagem significativa na navegação, uma vez que um caminho através de uma rede é facilmente representado como um conjunto de nodos de decisão. Um importante requisito para os sistemas de navegação é a capacidade de encontrar as melhores rotas para os utilizadores. "Melhores rotas", podem ser rotas mais curtas, rotas que consomem menos tempo, ou rotas de acordo com as exigências do utilizador [70]. No *outdoor*, estas são frequentemente calculadas com base numa rede de estradas. Contudo, em ambientes *indoor*, a construção de uma rede que apoie a procura do caminho mais curto é um fator mais desafiador. Por exemplo, a navegação entre diferentes pisos, que na realidade podem ter pontos com a mesma localização (longitude e latitude), é um problema a ter em conta. A rede resultante para um espaço *indoor* deve então estar ciente das distâncias entre os nodos de pisos diferentes.

Então, qual a melhor forma de se chegar ao grafo? De uma forma implícita, a partir do BIM, ou de uma forma explícita, isto é, manualmente? Não existem ainda respostas consensuais para estas questões. As abordagens escolhidas para o caso em estudo serão posteriormente apresentadas no Capítulo 4.

Técnicas de *routing* utilizam algoritmos que encontram o caminho mais curto

¹¹www.sketchup.com

¹²www.sketchup.com/3dwh/buildingmaker.html

entre dois locais. Geralmente, a relação de ligação entre os nodos de um grafo é representada por uma matriz bidimensional. A relação entre os nodos representa o peso, ou o custo da viagem de um nodo para outro, como a distância, o tempo, ou o grau de conveniência. Pode também ser representada por uma lista de arestas (*edges*). A escolha da estrutura de dados afetará o tamanho da base de dados, assim como o desempenho do algoritmo. Dentro dos algoritmos de *routing* destacam-se alguns mais comuns, como: *Dijkstra's*, *Bellman-Ford*, e A^* [71].

2.3 Soluções sobre o OSM

Uma nova e diferente tendência para recolha de dados tem evoluído, nos últimos anos, reportando uma recolha de informação geográfica voluntária e colaborativa. De acordo com Elwood [72], para esta tendência existem diversos termos diferentes, porém um dos termos mais populares é *Volunteered Geographic Information (VGI)*. VGI compreende o efeito que um crescente grupo de utilizadores cria, agrupa e divulga dados geográficos e espaciais de uma maneira colaborativa e voluntária. É uma combinação de elementos da *Web 2.0* e uma inteligente coletiva [73].

O mais popular e bem sucedido projeto VGI é provavelmente o *Open Street Map (OSM)*¹³. OSM apresenta-se como uma plataforma de mapeamento online (como a Wikimapia¹⁴), que permite aos utilizadores contribuir e editarem cooperativamente dados espaciais. Por outras palavras, OSM é um esforço internacional para criar uma fonte livre de informação de mapas, a partir de um esforço voluntário [73]. Os dados em OSM são criados de diferentes formas, por exemplo, com a aquisição de dados originais (capturados pelos utilizadores através de dispositivos GPS), como também, com a contribuição das pessoas com dados baseados em imagens aéreas (Bing Maps, por exemplo) ou com conhecimento local das pessoas sobre a região em que vivem. Desde 2004, o projeto cresceu rapidamente e em julho de 2012, já havia mais de 650.000 utilizadores registados e mais de 2.750 milhões pontos rastreados na base dados [74].

O crescente interesse em edifícios e na necessidade de soluções de navegação

¹³www.openstreetmap.org

¹⁴Serviço Web de pesquisa e visualização de mapas que utiliza imagens de satélite do planeta Terra para identificar lugares. Acessível em: wikimapia.org

indoor maduras, ou outros serviços baseados em localização *indoor*, como mapas *indoor* ou *indoor routing*, está a aumentar com tempo. Os fornecedores de informação dificilmente são capazes de capturar comercialmente dados *indoor* para grandes áreas. Como tal, pode-se dizer que existe um enorme potencial dentro das comunidades VGI (principalmente no OSM) para capturar e fornecer informação sobre ambientes *indoor* que estão abertos ao público (centros comerciais, hospitais, ou aeroportos por exemplo).

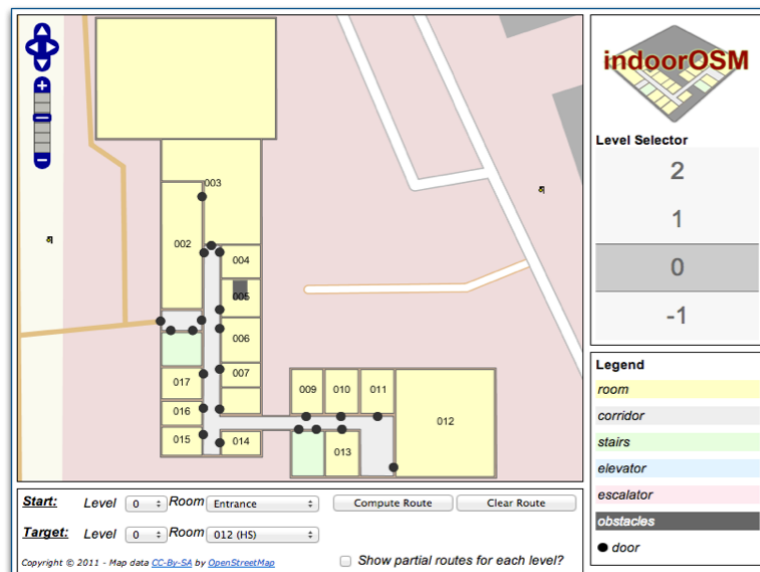
Na verdade, existem já algumas soluções para espaços *indoor* que integram o OSM. De acordo com [75], a *GIScience Group* da Universidade de Heidelberg, *Alpen Adria University Klagenfurt* na Áustria, e a *University of Science and Technology* da Noruega já desenvolveram as suas próprias aplicações de navegação *indoor* para os seus campus universitários. Outro exemplo, é a aplicação '*Fo-otPath*' [18] criada pela Universidade *RWTH Aachen* na Alemanha. Abordando o problema numa perspetiva 3D, existem também algumas contribuições que se encontram disponíveis para consulta em [76].

IndoorOSM, Heidelberg

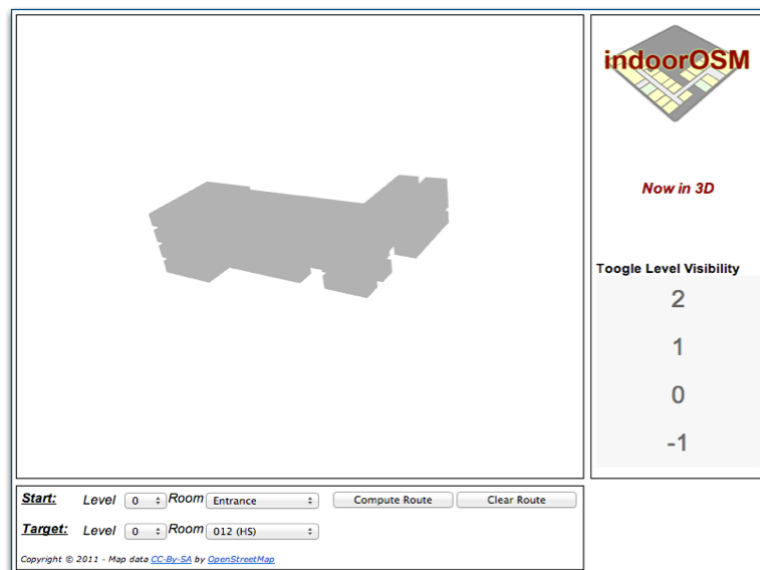
O *GIScience Group* da Universidade de Heidelberg desenvolveu uma promissora proposta de mapeamento *indoor*, chamada de IndoorOSM¹⁵ [77], que tem como principal objetivo demonstrar o poder e a diversidade de um OSM "estendido" ao *indoor*. A aplicação já se encontra disponível através de uma plataforma *web*, incluindo um mapa *indoor* 2D [8] e também um modelo do edifício 3D [9], ambos com capacidade de *routing* (ver Figura 2.11). A versão 2D do mapa possibilita a visualização da estrutura interior do edifício e permite a seleção individual de cada piso. Cada piso é representado como uma planta (*CAD floor plans*) em que as diferentes partes do edifício como os quartos, corredores, elevadores e escadas são ilustradas como polígonos coloridos, e por sua vez, as portas são representadas como pequenos pontos pretos circulares. Por outro lado, a versão 3D baseada em XML3D, proporciona a visualização das diferentes partes do edifício como modelos 3D, e permite também a seleção individual de cada piso (ocultar ou apresentar). Além disso, as rotas resultantes do cálculo do *routing* são apresentadas como um

¹⁵wiki.openstreetmap.org/wiki/IndoorOSM

grafo 3D.



(a)



(b)

Figura 2.11: Interface *web* da aplicação IndoorOSM com dados 2D (a) e 3D (b), Universidade Heidelberg [8, 9].

Campus GIS, Áustria

A aplicação Campus GIS [10] disponibilizada pela Universidade de *Klagenfurt*, na Áustria, proporciona serviços de *indoor routing* para estudantes, funcionários e visitantes. A Figura 2.12 ilustra a interface *web* da aplicação. Os utilizadores introduzem um ponto partida e um ponto de chegada, e a aplicação determina a rota através do campus, inclusive através de diferentes andares dos edifícios, e com acessibilidade para cadeiras de rodas opcional. Esta aplicação foi desenvolvida por um professor da universidade (através da empresa de consultoria GOMOGI¹⁶), e foi concebida para combater o problema dos atrasos dos novos alunos as salas devido a estes frequentemente se perderem pelo campus. Todo o sistema é baseado em tecnologias *open source*, incluindo: *Geoserver*, *GeoWebCache*, *Postgresql/PostGIS*, *OpenLayers*, *GeoExt*, *ExtJS* e utiliza o *PGRouting* para o algoritmo de *routing*.

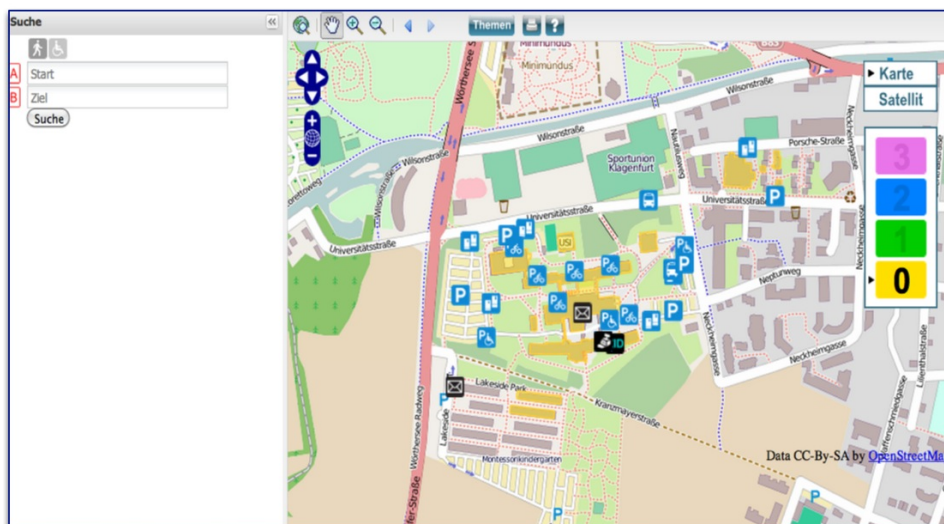


Figura 2.12: Interface *web* da aplicação Campus GIS, Universidade Klagenfurt [10].

Campus Guiden, Noruega

A Universidade de Ciência e Tecnologia da Noruega (*Norwegian University of Science and Technology*, NTNU) destaca-se como o primeiro caso de navegação *in-*

¹⁶www.gomogi.com

door numa universidade. A aplicação, designada de **Campus Guide** [78], resulta de um projeto de investigação e desenvolvimento conduzido pela *Wireless Trondheim AS* e pela NTNU, que teve a sua origem em 2008 com um projeto de um estudante. O serviço encontra-se disponível para utilizadores *iPhone* e *Android*, como também na *Web* [11]. A Figura 2.13 ilustra a interface *web* da aplicação. A particularidade desta aplicação é o facto de servir como guia tanto para *outdoor* como para *indoor*. O sistema consegue determinar a posição *indoor* com uma precisão até 5-10 metros utilizando a rede Wi-Fi da universidade, que contém mais de 1800 wireless routers. Para o *outdoor* o sistema utiliza o posicionamento GPS, que proporciona aproximadamente a mesma precisão que no *indoor*. O sistema baseia-se em várias tecnologias, como: *OSM*, *OpenLayers*, *jQuery* e *jsGettext*.



Figura 2.13: Interface *web* da aplicação **Campus Guiden**, NTNU [11].

Em suma, estas iniciativas comprovam que o atual modelo de dados do *OSM* é capaz de representar espaços indoor. Este modelo de dados, apesar da sua simplicidade, permite representar qualquer tipo de edifícios, através das suas primitivas: nodos, arestas e relações. Estas iniciativas serão a base deste projeto. A Tabela 2.2 proporciona uma visão geral das soluções apresentadas nesta secção, comparando cada proposta com uma lista de características agrupadas em quatro categorias.

Tabela 2.2: Comparação de cada proposta apresentada nesta secção com uma lista de características importantes a destacar.

	<i>IndoorOSM</i>	<i>Campus GIS</i>	<i>Campus Guiden</i>
Representação dos Espaços	<ul style="list-style-type: none"> mapa 2D e/ou modelo 3D 	<ul style="list-style-type: none"> mapa 2D 	<ul style="list-style-type: none"> mapa 2D
Interação	<ul style="list-style-type: none"> Seleção piso. 	<ul style="list-style-type: none"> Seleção piso; Opção de ocultar/apresentar <i>features</i> no mapa; Procura interativa dos compartimentos e visualizador. 	<ul style="list-style-type: none"> Seleção piso; Opção de localizar/apresentar <i>features</i> no mapa; Posição atual automática ou manual; Procura interativa dos compartimentos e visualizador.
Routing	<ul style="list-style-type: none"> Routing entre diferentes pisos do mesmo edifício. 	<ul style="list-style-type: none"> Routing entre diferentes edifícios e pisos; Routing adicional para cadeira rodas. 	<ul style="list-style-type: none"> Routing entre diferentes edifícios e pisos.
Visualização	<ul style="list-style-type: none"> Apenas dados OSM; <i>Display</i> mais organizado; Zoom limitado (out); Representação total ou parcial do caminho; Legenda. 	<ul style="list-style-type: none"> OSM+<i>Layers</i>; Zoom limitado; Opção de vista: mapa ou satélite; Legenda + informação adicional. 	<ul style="list-style-type: none"> OSM+<i>Layers</i>; Zoom in/out; Disponível em diferentes línguas; Legenda + informação adicional.

2.4 Soluções Comerciais

Um grande número de iniciativas/soluções para navegação *indoor* foram propostas, incluindo soluções comerciais. Contudo, como já referido, ainda não existe um conceito mestre que seja usado para todos os casos.

Sistemas como a Google¹⁷ (*Google Maps* 6.0), Nokia¹⁸ (*Navteq*) e Bing¹⁹ (*Bing Indoor Maps*) são exemplos de algumas das recentes soluções comerciais (ver exemplo das respetivas interfaces móveis na Figura 2.14). Estas soluções tentam adaptar os princípios tradicionais da triangulação para ambientes *indoor*, utilizando repe-

¹⁷googleblog.blogspot.pt/2011/11/new-frontier-for-google-maps-mapping.html

¹⁸www.gpsbusinessnews.com/NAVTEQ-Gets-Serious-About-Indoor-Maps_a2901.html

¹⁹siliconfilter.com/bing-maps-for-mobile-goes-indoors-with-mall-and-airport-maps

tidores de sinal GPS (p.ex., GPS+Beacon), Wi-Fi ou outras opções rádio (como NFC ou RFID). As três aplicações, embora ainda em desenvolvimento, já se encontram disponíveis e incluem mapas *indoor* (em 2D) de vários centros comerciais e aeroportos, principalmente nos EUA e Japão, estando assim esta navegação ainda limitada a determinadas regiões.

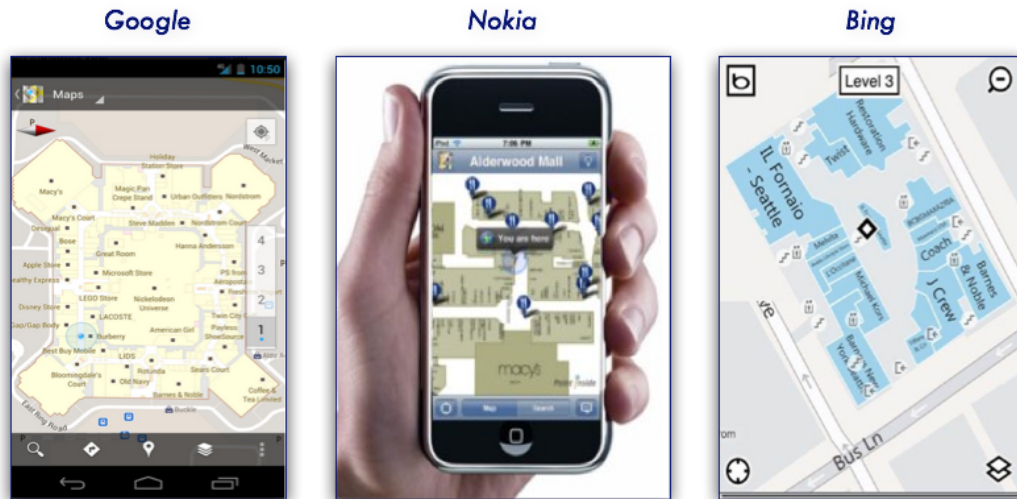


Figura 2.14: Interface móvel das aplicações *Google Maps 6.0*, *Navteq* e *Bing Maps* respetivamente (adaptada de ^{12 13 14}).

Capítulo 3

Levantamento a partir de imagens

Nos últimos tempos, o enorme desenvolvimento de câmaras digitais de alta qualidade e de baixo custo, assim como do poder computacional dos computadores pessoais, levou a que surgisse uma comunidade muito ativa de pesquisa em fotogrametria e visão por computador. Segundo a literatura atual, o "antigo sonho" de modelar o mundo em 3D à escala usando apenas imagens está agora mais próximo de se tornar realidade, pelo menos ao nível dos algoritmos [79]. No presente capítulo serão descritas algumas das ferramentas e técnicas para a obtenção de informação sobre a geometria de cenários 3D a partir de imagens 2D.

A metodologia adotada para o presente estudo foi o *Structure From Motions (SFM)*, que consiste numa metodologia fotogramétrica para a reconstrução 3D a partir de várias imagens separadas por distâncias de base pequena. Por outras palavras, *SFM* refere-se ao processo de encontrar a estrutura 3D de um objeto através da análise dos sinais de movimento, num determinado local, ao longo do tempo. A Figura 3.1 ilustra o fluxo típico desta técnica, *SFM*. Esta figura pode ser dividida em duas partes: (1) **Aquisição de dados**, que pode ser efetuada de três formas, através de um Lidar, Laser *scanner* ou Fotografias; e (2) **Geração do modelo 3D**, que pode ser ainda subdividido primeiramente em processos para criação da nuvem de pontos (*point clouds*¹), e posteriormente, ser então obtido o modelo 3D na íntegra. Estes dois pontos serão melhor descritos nas secções seguintes.

¹Uma *point cloud* consiste numa representação 3D precisa de um objeto real.

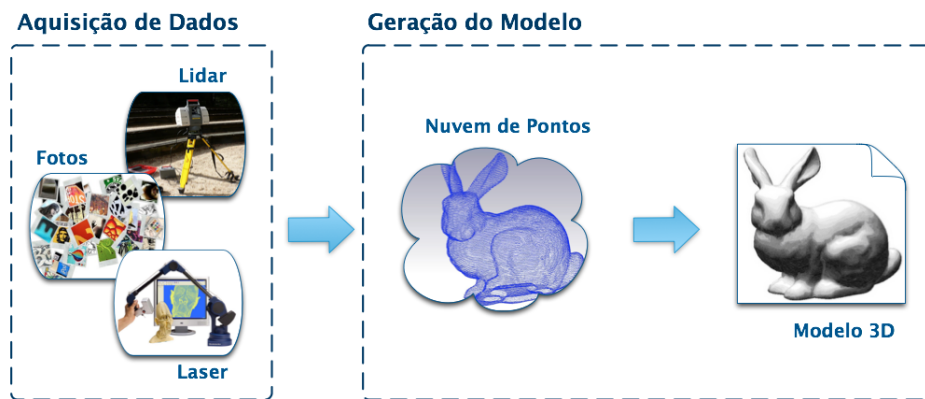


Figura 3.1: Fluxo típico da metodologia SFM.

3.1 Aquisição de Dados

O primeiro passo trata-se da aquisição de dados, ou seja, o levantamento dos dados (a forma, e possivelmente a aparência, p.ex, a cor) de um cenário ou objeto do qual se pretende obter a sua reconstrução 3D. Para elaborar este levantamento várias métodos podem ser adoptados, com o recurso de um Lidar ou Laser e/ou apenas a utilização de fotografias, devendo sempre considerar-se a adequação dos meios aos objetivos. Estes métodos vão desde os tradicionais levantamentos manuais até às modernas metodologias baseadas na *Digital Photogrammetry* e no *Laser Scanning 3D (LS3D)*. Cada tecnologia vem com as suas limitações, vantagens e custos.

Digital Photogrammetry

A fotogrametria (do inglês, *Photogrammetry*) consiste numa tecnologia passiva, sendo definida como a tecnologia que determina as propriedades geométricas de um objeto a partir de registos fotográficos desse mesmo objeto [80]. Não necessita de hardware específico, mas apenas de simples câmaras digitais. Tipicamente, esta tecnologia envolve a análise de uma ou mais fotografias (ou vídeos) usando software fotogramétrico de forma a determinar as relações espaciais. A título de curiosidade, a palavra fotogrametria, deriva de três palavras de origem grega: "*Photon*" - luz; "*Graphos*" - descrição; e "*Metron*" - medições, ou seja, medições efetuadas através de fotografias.

Para a criação de modelos 3D esta metodologia é normalmente designada de **digital stereophotogrammetry** [81]. De uma forma geral, duas ou mais fotografias do mesmo objeto, capturadas de diferentes ângulos, são analisadas usando-se o espaço onde as fotografias se sobrepõem ou pontos de referência comuns compartilhados pelas fotos, para conseqüentemente, criar um modelo digital 3D. As imagens são então mapeadas, pixel por pixel no modelo, para criar espaço mensurável em três dimensões.

A fotogrametria pode ser dividida em: aérea e terrestre [81]. A primeira, **fotogrametria aérea** (câmara colocada debaixo de um avião), é utilizada principalmente para produção de cartas topográficas, modelos digitais de terreno, fotografias rectificadas (ortofoto), e também em vários estudos como geológicos, ambientais e florestais. Na segunda, **fotogrametria terrestre** (câmaras portáteis ou colocadas num tripé), os mesmos procedimentos da fotogrametria aérea são aplicados, permitindo assim documentar, com elevada precisão, fachadas de edifícios e monumentos. Pode também ser aplicada para a criação de modelos 3D (de objetos ou áreas, como por exemplo, de edifícios), contudo estes não são normalmente utilizados no mapeamento topográfico.

Atualmente, a fotogrametria recorre às mais modernas tecnologias informáticas efetuando todo o processamento de imagens digitais, o que permite uma grande evolução na aquisição de dados geográficos. Existe um trabalho muito interessante, o IM2GPS [82], em que se tenta georreferenciar imagens aplicando um algoritmo muito simples de semelhança/correspondência entre fotos. Para estimar a posição de uma fotografia, este algoritmo utiliza a localização das fotografias mais parecidas que possui na base de dados. Contudo, além de se basear num algoritmo muito simples, este trabalho não é aplicável a espaços *indoor*, apenas *outdoor*.

Laser Scanning 3D

Não existe apenas uma tecnologia de *Laser Scanning 3D*. De certa forma, pode ser dividida em **aérea** (Lidar) e **terrestre**, consoante seja aplicada a partir de uma plataforma aérea em movimento (uma aeronave, por exemplo) ou aplicada a partir de uma plataforma imóvel situada ao nível do chão (por exemplo, sobre um tripé). Esta é uma divisão muito geral e na verdade as fronteiras não são assim tão

nítidas, pois o Lidar pode também ser aplicado a alguns sistemas terrestres [12]. Para efeitos da presente dissertação interessa considerar apenas o caso terrestre.

O *Terrestrial Laser Scanning (TLS)* consiste numa tecnologia ativa (os instrumentos recebem a luz que emitem), e comparando com a fotogrametria, é uma tecnologia relativamente mais recente. O TLS é essencialmente um procedimento pela qual se captura uma grande quantidade de coordenadas de pontos de uma forma sequencial e ordenada, e quase em tempo real, utilizando um feixe de luz laser. Os sistemas TLS geralmente operam num dos três princípios [12]:

1. Triangulação ótica - com precisão sub-milimétrica e alcance até 5m;
2. Tempo de voo (*time of flight*) - com precisão milimétrica a centimétrica e alcance até 1000;
3. Comparação de fase (*phase comparison*) - com precisão milimétrica e alcance até 100m.

Os primeiros sistemas são aplicados principalmente para o registo de objetos de pequena escala. Os dois últimos, são os mais adequados para o levantamento integral de edifícios. A título de exemplo, a estrutura básica de um sistema de triangulação é dado na Figura 3.2.

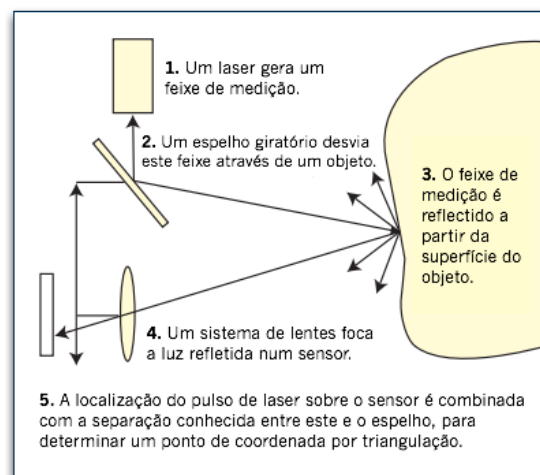


Figura 3.2: Diagrama esquemático de um sistema de medição baseado em triangulação ótica (adaptada de [12]).

O resultado da aquisição de dados por *scanner laser 3D* posicionado numa única estação é o que se designa por **nuvem de pontos 3D**. Uma única nuvem de pontos pode ter desde milhares de pontos, até milhões de pontos.

Em suma, por um lado o **TLS** proporciona uma elevada automatização no processo de aquisição de modelos **3D** densos e geometricamente precisos, contudo, o custo da aquisição de um *scanner laser* e do respetivo software necessário à sua manipulação, assim como dos dados adquiridos, é ainda muito elevado. Outra desvantagem é o facto das aquisições no terreno normalmente consumirem muito tempo, com a informação de cor (RGB) de pouca qualidade para obtenção direta da textura. Por outro lado, apesar da captura **3D** detalhada a partir da Fotogrametria ser altamente computacional e ambígua, esta metodologia apresenta determinadas vantagens comparando com o **TLS**, como o seu baixo custo, a sua flexibilidade, a sua rápida aquisição no terreno e a produção direta de nuvens pontos com textura, aplicando diretamente a informação de cor (RGB) através da aquisição das imagens. No entanto, a precisão geométrica é fortemente dependente da textura dos objetos, da calibração da câmara, dos pontos de controlo utilizados, do número e da resolução das imagens usadas e da rede geométrica [83]. Portanto, a questão de qual dos dois métodos proporciona os melhores resultados relativo a diferentes critérios ainda permanece em aberto.

3.2 Geração de Modelos

O segundo passo é então o da geração de modelos. Nos últimos tempos, têm ocorrido alguns desenvolvimentos interessantes e tem-se verificado uma maior disponibilidade em relação a software *open-source*, que permite aos utilizadores criarem nuvens de pontos e modelos através do uso das suas câmaras digitais e software livre disponível online. O desenvolvimento tem sido bastante rápido e as opções de se poder utilizar diferentes bibliotecas vêm crescendo, à medida que mais pesquisas vão sendo feitas e à medida que os programadores individuais doam o seu tempo e trabalho, fornecendo soluções para criação de nuvens de pontos **3D** e *meshes*² [84]. Estas, por sua vez, podem ser diretamente importadas para **CAD** ou bibliotecas

²Consiste numa subdivisão poligonal da superfície de um modelo geométrico.

de modelagem.

Nesta secção, serão apresentadas algumas das soluções *open-source* para a criação de nuvens de pontos e modelos 3D, e também uma análise de alguns dos prós e contras de cada uma. A lista de software disponível atualmente é diversa, por exemplo, existem: Visual SFM, FIT3D, Microsoft Photosynth, SFM Toolkit, Insight3d, Autodesk 123 Catch, Apero, Hypr3d, CouldCompare, My 3Dscanner, Bundler, ARC 3D, entre outros.

Algumas destas aplicações foram estudadas e testadas, com o propósito de avaliar alguns parâmetros como a precisão dos modelos, o tempo de processamento e a facilidade de utilização. Estes parâmetros serão descritos de seguida. Para os testes realizados, em cada uma das aplicações, foi tido como base um conjunto de fotografias obtidas dentro e fora do campus da universidade, sem ter sido considerado qualquer requisito especial de cada software.

VisualSFM

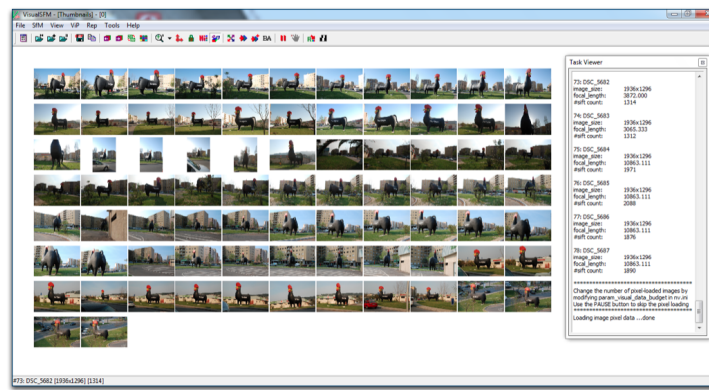
VisualSFM (*Structure From Motion*) consiste numa aplicação *open source*, que permite ao utilizador manter todas as fotos no seu computador e ter a capacidade de processá-las para criar um conjunto denso de nuvens de pontos.

O conceito básico desta aplicação teve a sua origem em 2006 com *Changchang Wu*, um estudante da universidade de North Carolina em Chapel Hill [84]. O que faz do VisualSFM uma aplicação interessante está no próprio nome - "visual". Esta aplicação apresenta uma interface gráfica e, embora ainda em desenvolvimento, apresenta uma excelente capacidade de rápido processamento de inúmeras fotos, tirando vantagem de computadores com placas gráficas nVidia e ATI. Encontra-se disponível para os diferentes sistemas operativos (*MacOS*, *Windows* e *Linux*), e toda a documentação necessária pode ser acedida através da página *web*³ da aplicação.

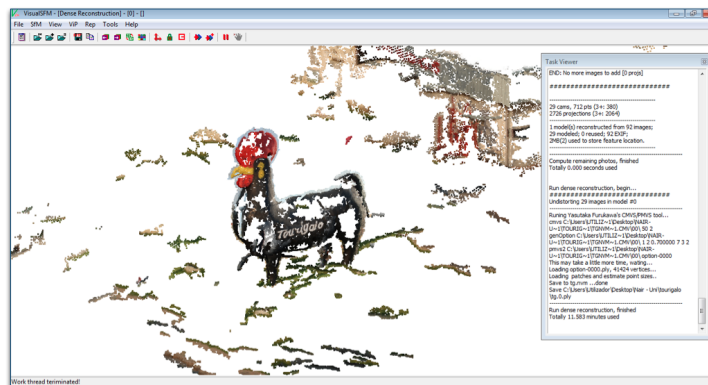
A reconstrução 3D no VisualSFM pode ser resumida em apenas quatro fases: (1) carregar as fotos pretendidas para o programa - ver Figura 3.3(a) (para o exemplo teste foram usadas 79 fotos); (2) realizar a função de correspondência entre as imagens (i.e., procura das características semelhantes); (3) cálculo da

³homes.cs.washington.edu/~ccwu/vsfm

reconstrução dispersa (i.e, as correspondências relevantes entre as imagens são agora determinadas para o cálculo das suas posições 3D num sistema de coordenadas relativo); e por fim, (4) processar a reconstrução densa. Quando completa esta última fase, é possível observar a nuvem de pontos na janela principal (ver Figura 3.3(b)). O tempo para completar uma reconstrução densa varia, podendo ser de alguns segundos para apenas algumas imagens, a várias horas para grandes conjuntos de fotos. A respetiva construção é guardada em formato **Polygon File Format (PLY)** e pode ser posteriormente aberta num software de visualização para processamento futuro.



(a)



(b)

Figura 3.3: Exemplo teste no VisualSFM: (a) *Upload* das fotos - 1a fase do processo; (b) Reconstrução e visualização da nuvem pontos - fase final do processo.

Acrescenta-se que a interface apresenta uma janela extra (à direita da janela

principal) designada ”*Task Viewer*” (ver Figura 3.3). Esta torna-se frequentemente útil, uma vez que possibilita observar todo o progresso dos diferentes processos (fase a fase), aquando da reconstrução 3D .

A aplicação **VisualSFM** mostrou-se simples e intuitiva, tendo sido obtido resultados razoáveis. É de notar, que a aplicação tenta corresponder todas as fotos, contudo, dependendo de como estas foram tiradas, qualquer área que não conseguiu ser correspondida pode originar um conjunto fragmentado de modelos e múltiplos modelos de reconstruções densas.

My3DScanner

My3DScanner⁴ é um serviço online gratuito, mas não *open source*, para criação de modelos 3D diretamente de fotografias digitais comuns (ou vídeos).

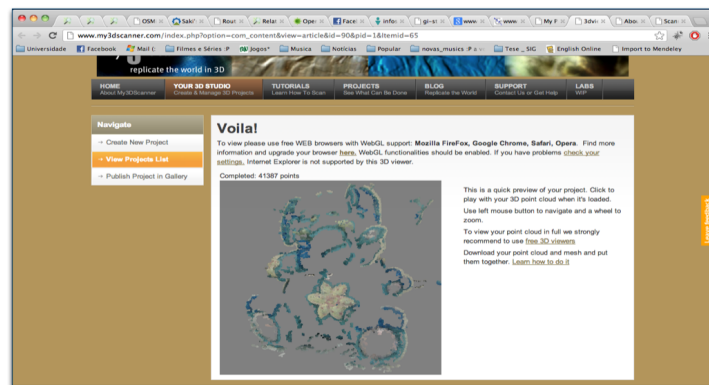
Este serviço baseia-se nos recentes avanços das tecnologias de visão por computador e caracteriza-se como um sistema *SFM* clássico, incluindo um modelo *Blunder*⁵ e um *PMVS*⁶. Apenas são necessárias algumas fotos 2D sobrepostas, capturadas de diferentes pontos de vista em torno do objeto, e a aplicação automaticamente reconstrói um objeto ou cena num modelo 3D .

Portanto, a reconstrução 3D no **My3DScanner** consegue ser resumida num pequeno conjunto de simples passos: (1) Recolha de informação visual 3D de um objeto ou cena com uma câmara digital ou telemóvel (capturar muitas fotos sobrepostas em torno do objeto sempre num ângulo fechado); (2) Fazer o registo no site **My3DScanner** e, de seguida, carregar todas as fotos para o servidor; (3) A aplicação processa as fotos, e após um tempo, disponibiliza a nuvem de pontos 3D para visualização no próprio site (ver Figura 3.4) ou para *download* (em formato **PLY**); e por fim (4) a aplicação gera a *mesh* a partir da nuvem de pontos, disponibilizando também para posterior visualização ou mesmo *download* (em formato **.OBJ**).

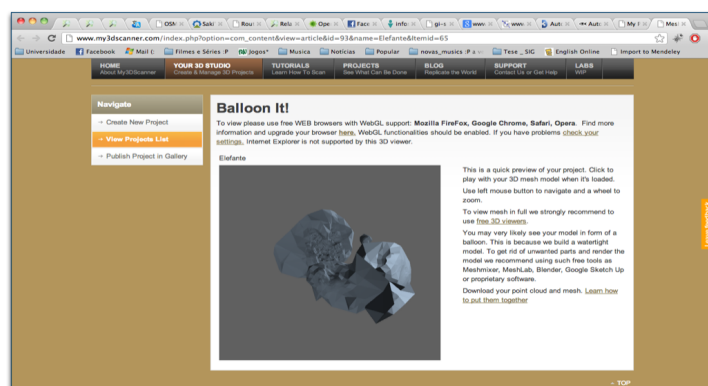
⁴www.my3dscanner.com

⁵phototour.cs.washington.edu/bundler

⁶grail.cs.washington.edu/software/pmvs



(a)



(b)

Figura 3.4: Exemplo teste no My3DScanner: (a) Visualização da nuvem de pontos na página *web* do serviço; (b) Visualização da *mesh* originada.

A aplicação My3DScanner mostrou-se muito simples e prática, apresentando a vantagem de não ser preciso descarregar qualquer software, realizando todo o processo online. Contudo, os resultados obtidos não foram tão bons quanto o esperado. Exige uma ótima qualidade das imagens capturadas, e apresenta um limite no número de fotos possíveis a carregar no site, máximo 100.

Autodesk 123D Catch

123D Catch⁷ consiste numa aplicação gratuita, mas não *open source*, que faz parte de um conjunto gratuito de aplicações 3D proporcionado pela Autodesk, que

⁷<http://www.123dapp.com/catch>

também inclui as aplicações 123D Sculpt e 123D Make. Esta aplicação proporciona a criação de excelentes modelos 3D a partir de fotografias digitais comuns. Só se encontra disponível para o sistema *Windows*, e recentemente, para dispositivos móveis como *iPhone 4/4S* e *iPad*. Além do software disponível para descarregar no PC, a *Autodesk* proporciona também uma versão desta aplicação via *web*⁸.



(a)



(b)

Figura 3.5: Exemplo teste no 123D Catch: (a) Visualização da nuvem de pontos com as posições da câmara ativas ('textura + wireframe'); (b) Visualização da nuvem de pontos só com a componente 'wireframe'.

A reconstrução 3D no 123D Catch apresenta-se como uma ação simples, os utilizadores apenas precisam de descarregar as suas imagens para a aplicação, que serão então enviadas para o processamento. Após alguns minutos é gerado

⁸<http://apps.123dapp.com/catch/>

o modelo 3D, e o mesmo é apresentado na interface da aplicação, podendo ser posteriormente editado através das várias ferramentas disponíveis no programa. Uma vantagem desta aplicação é a de possibilitar que o utilizador continue o seu trabalho, não sendo "obrigado" a esperar que o modelo seja gerado. Assim que o modelo esteja pronto, a aplicação envia um e-mail ao utilizador a notificá-lo da conclusão do processo.

A aplicação 123D Catch provou ser muito intuitiva, dinâmica e simples, tendo sido conseguidos excelentes resultados (ver Figura 3.5). A título de curiosidade, esta aplicação proporciona a possibilidade de enviar os respetivos dados para impressoras 3D, e assim, criar um modelo em miniatura do objeto real.

Apero/MICMAC

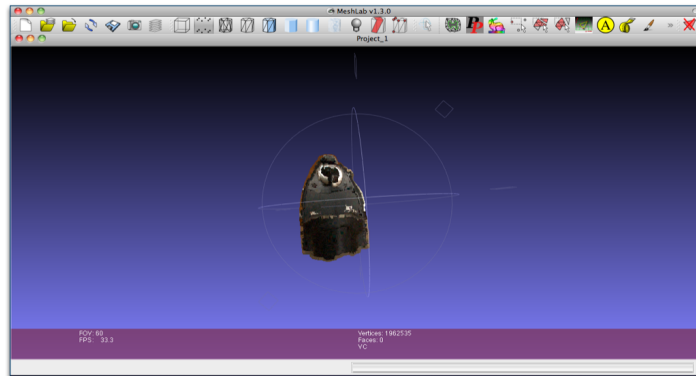
O *Institut Géographique National* (IGN), mais propriamente o laboratório Matis, tem desenvolvido um conjunto de ferramentas fotogramétricas⁹, **Apero** e **MICMAC**, para a criação de modelos 3D densos e precisos a partir de um conjunto de imagens. Inicialmente, este software foi desenvolvido como resposta a necessidades internas, contudo, atualmente encontra-se como *open source*. Comparando com algumas outras iniciativas de software livre, é provavelmente a mais complexa mas também mais completa, sendo os seus utilizadores-alvo profissionais como arquitetos, arqueólogos e geomorfologistas [79]. O utilizador deve estar ciente que estas ferramentas não 'trocam' a precisão pela flexibilidade de criar modelos 3D a partir de imagens desordenadas, e portanto, este deve seguir as regras de aquisição fotogramétrica para obter os melhores resultados.

A ferramenta **Apero** permite determinar a orientação das imagens e a ferramenta **MICMAC** utiliza as orientações e calibrações processadas pela ferramenta anterior, para criar mapas de profundidade e nuvens de pontos 3D densas. Ambas encontram-se disponíveis para plataformas *Linux* e *MacOS*.

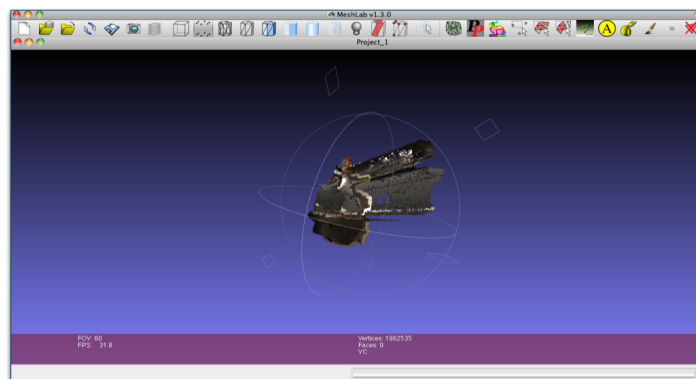
Este conjunto de ferramentas mostrou-se muito técnico e pouco interativo. A interface disponibilizada é ainda uma versão beta, e como tal, não funciona muito bem, o que levou ao uso da linha de comandos para poder correr aplicação. Tal facto tornou todo o processo mais complexo, e consequentemente, os resultados

⁹www.tapenade.gamsau.archi.fr/TAPEnADe/Tools

obtidos não foram tão bons quanto o esperado (ver Figura 3.6). Além disso, o facto destas ferramentas exigirem requisitos especiais na aquisição das imagens (os quais não foram tidos em conta) influenciou em certa parte os resultados obtidos.



(a)



(b)

Figura 3.6: Exemplo teste no Apero/MICMAC: (a) e (b) visualização no MeshLab do modelo obtido.

3.3 Visualização dos Modelos

O próximo passo, após adquirir os modelos 3D, consiste na visualização e navegação dos mesmos. Isto é possível a partir da utilização de um número diverso de software profissional que atualmente se encontra disponível, como: Blender, ccViewer, Meshlab, CloudCompare e Geomagic [85]. A propósito, é necessário ter

um rato de computador conetado para se conseguir trabalhar com modelos 3D em qualquer uma destas aplicações, pois apenas um *touch-pad* ou *touch-screen* não permitirão navegação nos modelos.

Para os testes realizados, o software de visualização utilizado foi o Meshlab¹⁰, que é talvez a aplicação de nuvem de pontos 3D e *meshing* mais popular disponível como *open-source*. MeshLab teve a sua origem em Itália, na Universidade de Pisa em 2005, e desde então, tem percorrido um longo caminho em relação às suas características. O sistema apresenta como principal objetivo auxiliar o processamento dos típicos modelos não estruturados provenientes do 3D *scanning*, proporcionando um conjunto de ferramentas para edição, limpeza, recuperação, inspeção, processamento e conversão deste tipo de *meshes*. Encontra-se disponível para os diferentes sistemas operativos (*MacOS*, *Windows* e *Linux*), e possibilita importar uma vasta lista de possíveis formatos, no entanto, o formato padrão para esta aplicação é o formato PLY.

Embora o MeshLab tenha requerido uma aprendizagem inicial acerca de como utilizar plenamente as suas funcionalidades, este mostrou-se muito simples de usar apenas como visualizador de nuvens de pontos 3D. A Figura 3.7 ilustra um exemplo teste da aplicação do MeshLab para a visualização de uma nuvem de pontos, que representa um departamento da universidade.

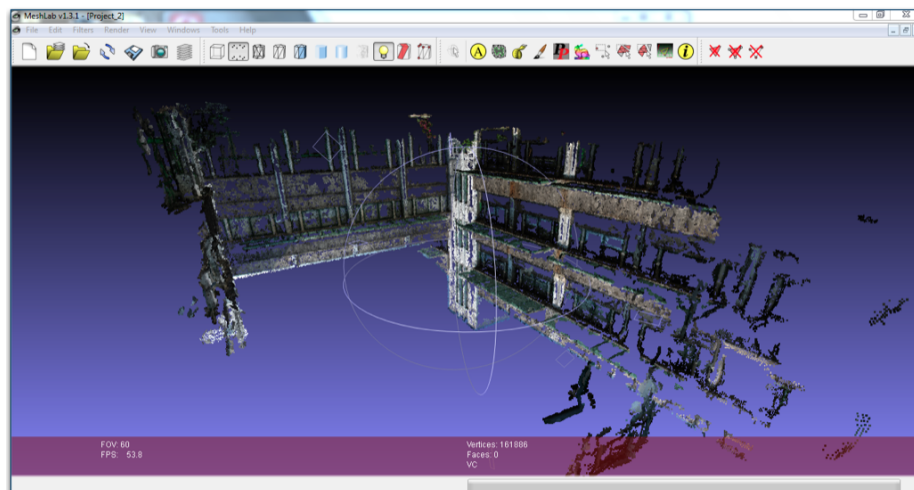


Figura 3.7: Exemplo teste do importe de um ficheiro PLY no MeshLab.

¹⁰meshlab.sourceforge.net

A construção de modelos 3D a partir de imagens, embora seja uma técnica relativamente recente, mostrou ser acessível e promissora. Foram conseguidos resultados razoáveis neste estudo, contudo verificou-se que a utilização desta técnica para qualquer aplicação exige sempre o cuidado na captura das imagens assim como na qualidade das mesmas, de forma a se conseguir alcançar um bom ou muito bom resultado. Este método de reconstrução 3D, apesar todas estas vantagens, não vai ser aplicado ao caso em estudo, à presente dissertação. Isto porque foi possibilitado para o estudo um modelo 3D em *Revit*, mais pormenorizado, e o qual será utilizado e aplicado na criação da solução *indoor* proposta, como se irá verificar no capítulo seguinte, Capítulo 4.

Capítulo 4

NavIOS: Arquitetura e Implementação

Navigation Indoor OSM System (NavIOS) é uma aplicação *web* desenvolvida para solucionar os problemas característicos do *indoor*. A aplicação permite a visualização e a exploração 2D da informação *indoor*. Proporciona também uma solução de *routing* multi-modal, que tira vantagem da informação *indoor* disponível. A solução de *routing* foi desenvolvida sobre a plataforma [Open Trip Planner \(OTP\)](#). A arquitetura escolhida assim como o tratamento de dados necessário e todo o processo na implementação desta aplicação serão descritos em maior detalhe nas seções seguintes.

4.1 Arquitetura

Estruturar uma aplicação é mais importante do que programá-la. Por vezes torna-se impossível compreender como uma aplicação se encontra implementada devido à sua má estrutura, o que leva a uma arquitetura complicada. Por outro lado, existem também algumas aplicações que apresentam uma boa estrutura, mas falham na utilidade das suas características. Portanto, é necessário manter um equilíbrio entre estes dois fatores (estrutura e características) aquando do desenvolvimento de uma aplicação.

A arquitetura da aplicação proposta na presente dissertação, NavIOS, encontra-

se alinhada com o modelo MVC. O **Model View Controller (MVC)** consiste num modelo de desenvolvimento de software, que foi descrito pela primeira vez em 1979 por *Trygve Reenskaug* [86], e atualmente é considerada uma "arquitetura padrão" aplicada na engenharia de software. O modelo isola a "lógica" (a lógica da aplicação) da interface do utilizador (inserir e exibir dados), permitindo assim desenvolver, editar e testar separadamente cada parte, e conseqüentemente, aumentar a flexibilidade e o reuso. Apresenta três camadas básicas (ver Figura 4.1), cada uma delas com as suas características e funções bem definidas, sendo estas:

- ◇ **Modelo** (*model*) - camada que possui a representação detalhada de toda a informação que a aplicação opera. Utilizada para definir e gerenciar o domínio da informação de forma consistente, e notificar observadores (camada Visão) sobre alterações nos dados. Tem acesso a toda e qualquer informação, sendo esta vinda de uma Base de Dados ou um arquivo XML por exemplo. O modo como os dados são armazenados ou acedidos não é de interesse do MVC, mas sim da responsabilidade da camada Modelo, que persiste os mesmos sem se preocupar como serão mostrados.
- ◇ **Visão** (*view*) - camada responsável pela construção de uma interface para interação com o utilizador e pela a apresentação das várias visões dos dados. Foca-se apenas na apresentação dos dados, e não no modo como eles foram obtidos, ou seja, apresenta o modelo num formato adequado ao utilizador.
- ◇ **Controlador** (*controller*) - camada que gere o fluxo de apresentação das informações efetuando a intermediação entre as camadas de Visão e de Modelo. Cada requisição do utilizador é inicialmente interpretada pela camada de controlo que determinará qual a ação que deverá ser efetuada no modelo, invoca os objetos do modelo, e por fim, encaminha a respetiva resposta para a camada visão. Esta camada é também responsável pela validação e filtragem da entrada de dados.

Hoje em dia, os sistemas/plataformas *web* desenvolvidos apresentam-se cada vez mais complexos. Desta forma, a aplicação do modelo MVC torna-se muito vantajosa, uma vez que facilita a reutilização do código, qualquer tipo de alteração numa das camadas não afeta as outras simplificando assim a atualização de *layouts*,

a alteração de regras, assim como a manutenção e inclusão de novos recursos. No caso de grandes projetos, o MVC facilita muito a divisão de tarefas entre equipa.

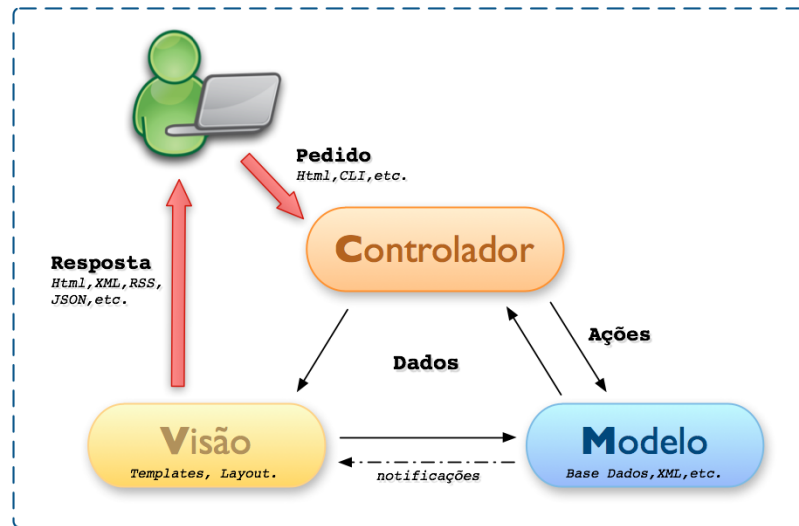


Figura 4.1: Diagrama representativo do modelo MVC, exemplificando a relação entre as três camadas da arquitetura (modelo, visão e controlador).

4.2 Tratamento de Dados

A informação espacial/geográfica é essencial no desenvolvimento de qualquer aplicação *web* baseada em localização. Quanto mais precisa é a informação fornecida, mais precisos serão os resultados proporcionados pela aplicação. Esta informação é utilizada no processo de geração do mapa para navegação e posicionamento *indoor*.

Para o desenvolvimento da aplicação proposta (NavIOS) foi cedida pela Universidade toda a informação espacial necessária, em forma de plantas arquitetónicas, relativa ao edifício em estudo - Departamento de Informática (DI). Estas plantas foram originalmente recebidas (em papel) no formato CAD, e posteriormente transformadas num modelo 3D utilizando o software Revit¹. O Revit consiste num software BIM para arquitetos e engenheiros, que permite aos utilizadores

¹usa.autodesk.com/revit

projetar um edifício e os seus componentes em formato 3D, anexar ao modelo elementos auxiliares 2D (plantas digitalizadas, por exemplo) e aceder a toda informação do edifício a partir da base dados do modelo.

O principal objetivo é transformar o modelo 3D concebido em objetos sofisticados no OSM. Portanto, algumas transformações dos dados foram necessárias (ver Figura 4.2). O acesso ao modelo 3D do edifício (neste caso, em Revit) deu origem a três diferentes componentes do sistema:

1. Dados 3D para visualização do modelo;
2. Dados para Geoserver;
3. Dados para JOSM.

Estes três componentes possibilitam que toda a informação relativa ao edifício em estudo esteja disponível na plataforma *web*, para desta forma poder ser visualizada. Encontrar uma estrutura de dados adequada para representar esta informação tornou-se no principal desafio para a conceção da presente aplicação.

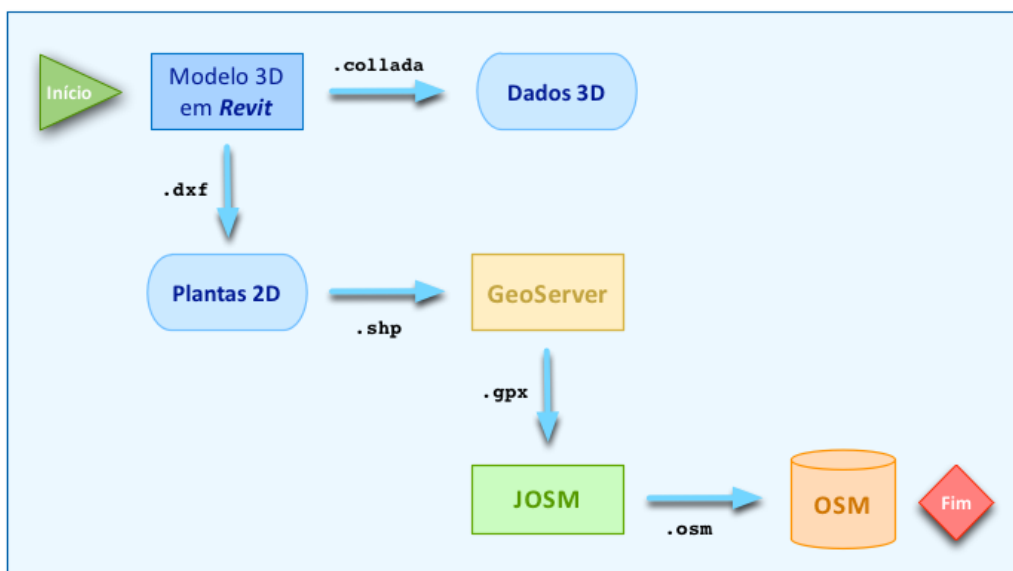


Figura 4.2: Processo geral de tratamento dos dados.

4.2.1 Dados 3D

A primeira transformação necessária dos dados consistiu na conversão do modelo 3D em Revit para ficheiros .dxf (*drawing exchange format*) e .dae (Collada). A conversão para .dxf foi necessária para se conseguir as plantas 2D em formato digital de cada piso do Departamento de Informática, edifício em estudo. Por outro lado, o ficheiro Collada possibilitará a visualização dos dados 3D no OSG earth², e posteriormente na aplicação *web* NavIOS. A Figura 4.3 ilustra o resultado da conversão do modelo 3D do edifício para um ficheiro Collada.

Os dados 3D do edifício em estudo ficam assim previamente preparados. No entanto esta disponibilização de dados 3D para visualização e navegação na plataforma *web* não foi possível ainda ser efetuada. Este será um passo a realizar no futuro, uma vez que nesta fase de estudo, não se tornou possível conjugar o tempo disponível com a execução da tarefa. Visto que os dados já estão preparados, este passo a realizar no futuro será mais rápido, e exigirá menor esforço. Será no entanto necessário perceber a melhor forma de os disponibilizar na plataforma.

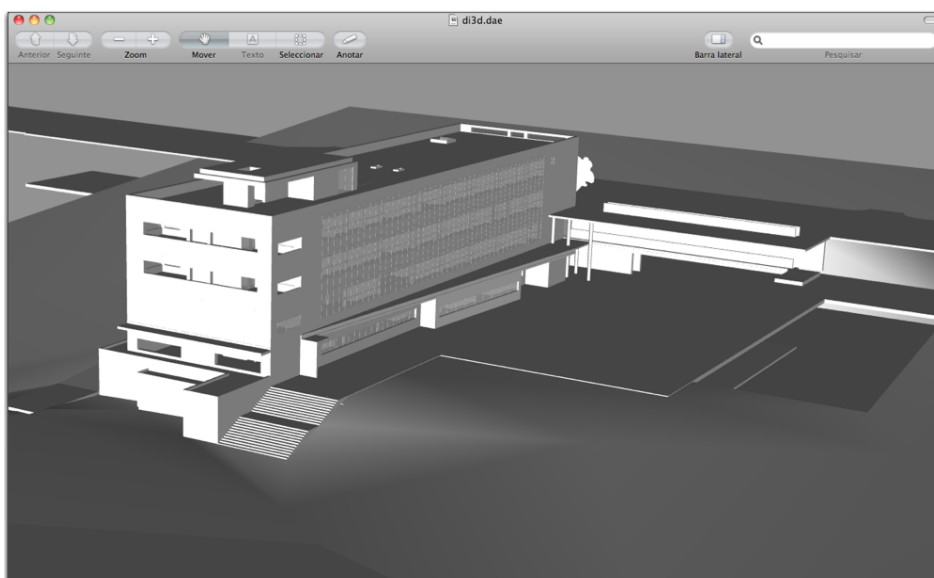


Figura 4.3: Resultado da conversão do modelo 3D do DI para Collada (.dae).

²osgearth.org

4.2.2 Geoserver

Geoserver³ é um software *open source*, desenvolvido em Java, que permite partilhar e editar dados geoespaciais, e desta forma desenvolver soluções de *web mapping*. Assim, para se visualizarem na plataforma *web* proposta as respetivas plantas 2D de cada piso sobrepostas no OSM, assim como selecionar cada uma delas, foi necessário publicar as mesmas no servidor de mapas Geoserver. Para tal, foi preciso primeiro converter os ficheiros *.dxf* de cada piso para o formato *shapefile* (*.shp*). Este passo foi realizado executando o comando, representado no Código 4.1, no terminal. A Figura 4.4 ilustra um exemplo da pré-visualização do *layer*, neste caso relativo ao piso três do DI, no Geoserver.

```
1 ogr2ogr -skipfailures -f "ESRI Shapefile" -a_srs EPSG:3763 -nlt  
  LINESTRING piso_terrace.shp DI_2012-10-21_alt-FloorPlan-  
  Plantadepiso4-terraço-cota15-10.dxf
```

Código 4.1: Comando de conversão de *.dxf* em *.shp*.

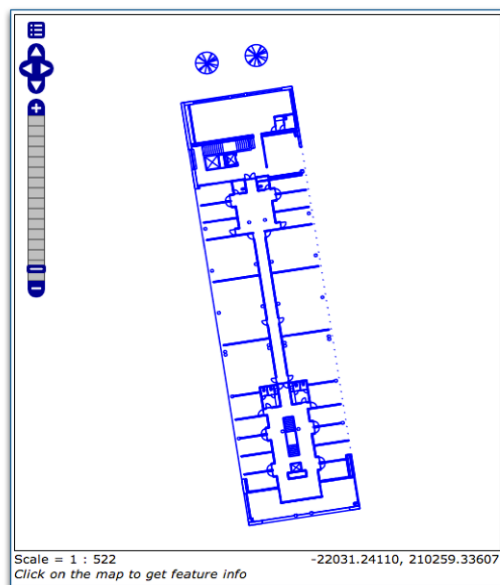


Figura 4.4: Exemplo da pré-visualização no Geoserver do *layer* relativo ao piso três do DI.

³geoserver.org/display/GEOS/Welcome

4.2.3 JOSM

Por fim, os dados necessários para a construção do grafo de routing foram obtidos com a conversão das plantas de cada piso em formato `.shp` primeiramente para o formato `.gpx` e depois para o formato `.osm`. Os comandos executados no terminal que possibilitaram este passo encontram-se definidos no Código 4.2.

```
1 ogr2ogr -skipfailures -f "GPX" -s_srs EPSG:3763 -t_srs EPSG:4326 -nlt  
   LINESTRING piso_gpx_terraco.gpx piso_terraco.shp  
3 gpsbabel -i gpx -o osm piso_gpx_terraco.gpx piso_gpx_terraco.osm
```

Código 4.2: Comando de conversão de `.shp` em `.gpx`, e por fim, para `.osm`.

Geração do Grafo de Routing (2D)

Como pretendido, o grafo de routing foi gerado a partir dos dados do [OSM](#). Para isso, foi utilizado o editor [JOSM](#) (Java Open Street Map - ferramenta de edição [OSM](#) avançada) para marcar explicitamente as rotas para todos os caminhos possíveis dentro do edifício.

Os ficheiros `.osm` de cada piso foram então importados para o [JOSM](#) de forma a auxiliar a geração do grafo. Cada um destes funcionou como imagem de fundo à medida que se foram desenhando os caminhos possíveis para cada piso (ver [Figura 4.5](#), exemplo da construção do grafo no piso 1).

O grafo é composto por nodos e arestas (*ways*). A cada divisão foi atribuído um nodo (definindo um ponto médio em cada divisão), e todos os nodos foram ligados entre si através de arestas. Uma aresta representa um caminho no grafo, por exemplo um corredor dentro do edifício, enquanto que um nodo representa um ponto intermédio do caminho ou uma entrada numa divisão (como ilustrado na [Figura 4.5](#)).

Aos objetos do grafo foram atribuídas diferentes *tags*, que consistem num par `chave/valor` utilizado para os descrever detalhadamente. A semântica seguida para representar o interior do edifício, ou seja, as *tags* utilizadas para representar cada objeto, foram as seguintes:



Figura 4.5: Exemplo da construção do grafo no JOSM para o piso 1, tendo como auxílio a planta do piso (ficheiro .osm).

- **Nodos:**

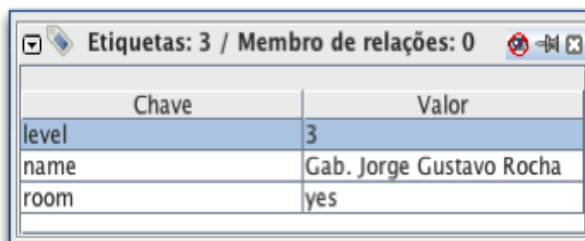
- ◇ *Entrada do edifício:* `room=entrance, level=*, name=*`
- ◇ *Portas:* `room=yes, level=*, name=*`
- ◇ *Ponto intermédio no elevador:* `level=*, name=*`

- **Arestas:**

- ◇ *Corredores:* `highway=footway, level=*, name=*`
- ◇ *Escadas:* `highway=steps, oneway=no, level=*, name=*`
- ◇ *Elevador:* `highway=elevator, name=*`

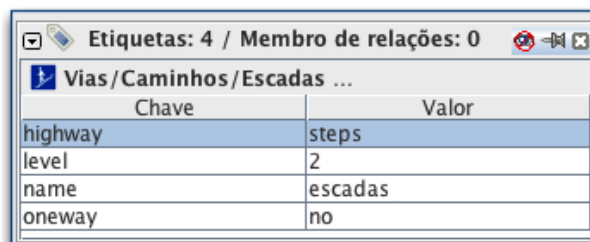
O valor * nas *tags* toma o valor correspondente do piso, no caso em que esteja associado ao valor do piso (p.ex, `level=1`), ou toma o valor do nome do objeto (p.ex, `name=entrada_DI`). A título de exemplo, na Figura 4.6 estão representadas as tags que descrevem um nodo. Neste caso, este nodo representa um gabinete que pertence ao professor Jorge Gustavo Rocha e encontra-se no piso 3 do edifício. A Figura 4.7 apresenta todas as propriedades que descrevem uma aresta, podendo-se constatar que esta aresta representa uma via que é composta por escadas, com o

nome "escadas", não se trata de uma via de sentido único e encontra-se no piso 2 do edifício.



Chave	Valor
level	3
name	Gab. Jorge Gustavo Rocha
room	yes

Figura 4.6: *Tags* utilizadas para descrever um nodo.



Chave	Valor
highway	steps
level	2
name	escadas
oneway	no

Figura 4.7: *Tags* utilizadas para descrever uma aresta.

A representação das escadas e do elevador mostrou-se o maior desafio aquando da geração do grafo. No contexto de semântica para espaços *indoor*, ainda existe presente a dúvida de qual a melhor forma de os representar. Neste trabalho decidiu-se representá-los de uma forma que fosse simples e perceptível para utilizadores comuns.

O edifício do DI é constituído por cinco pisos (0, 1, 2, 3 e 4). O último piso é uma sala/bar com terraço exterior, sendo este apenas acessível por umas escadas. Inicialmente foi tratada a questão do elevador. A cada piso do edifício (do piso 0 ao 3, uma vez que o elevador apenas está acessível para estes pisos) foi associado um nodo a representar a entrada no elevador. O caminho do elevador é composto na íntegra por um conjunto de arestas que definem os trajetos entre os diferentes pisos, sendo atribuída a cada uma das arestas a *tag* 'highway=elevator' e o nome correspondente. Neste caso, foram desenhadas três arestas que são relativas ao caminho entre os quatro pisos do edifício, e entre as arestas encontram-se os nodos

de cada piso, como ilustrado na Figura 4.8. Assim, tornou-se possível representar todos os caminhos entre os diferentes pisos através do elevador. Por exemplo, se alguém entrar no elevador no piso 0 e sair no piso 2, o resultado do routing será a passagem na aresta entre o piso 0 ao 1 e a passagem na aresta entre o piso 1 e 2.

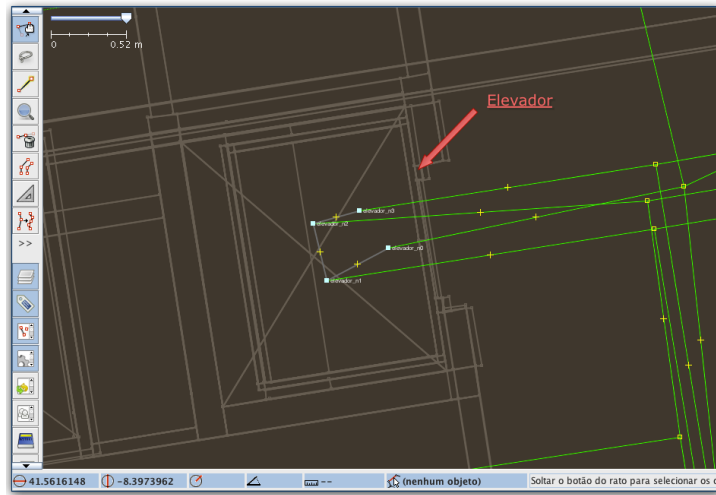


Figura 4.8: Representação do elevador no grafo de *routing*.

Relativamente às escadas, para cada piso foram inseridos nodos representando o seu início e o fim. Estes nodos são ligados por arestas que representam o caminho a percorrer, ou seja, as escadas entre cada piso, sendo atribuída a cada uma destas arestas a tag `'highway=steps'`, a tag `'oneway=no'` e nome correspondente. Uma vez que existem cinco pisos, incluindo o terraço, foram desenhadas quatro arestas para este caso. De forma a obter uma melhor visualização das mesmas no grafo, estas foram desenhadas paralelamente e não sobrepostas, como se pode ver no exemplo da Figura 4.9.

No final de todo este processo, obteve-se o grafo de *routing* completo do edifício, com todos os pisos conectados (ver Figura 4.10). O grafo contém a rede com todos os caminhos possíveis para a navegação no interior do edifício. Toda a informação relativa ao grafo foi importada para um ficheiro em formato `.osm`, e posteriormente os dados foram inseridos no OSM, onde receberam os IDs positivos válidos.

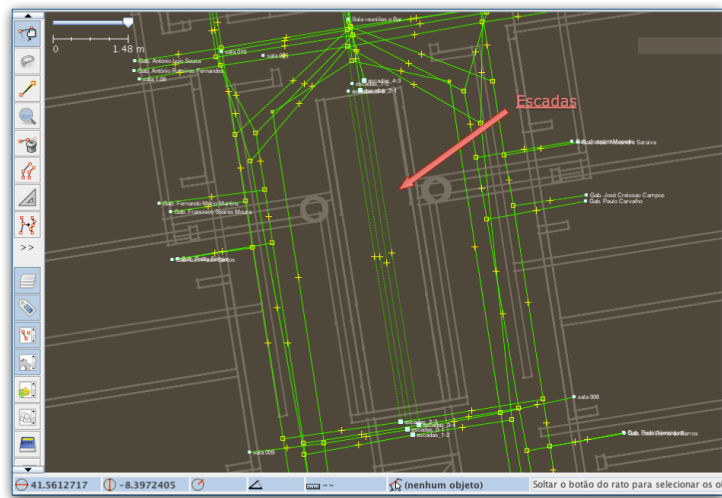


Figura 4.9: Exemplo da representação da escadas no grafo de *routing*.

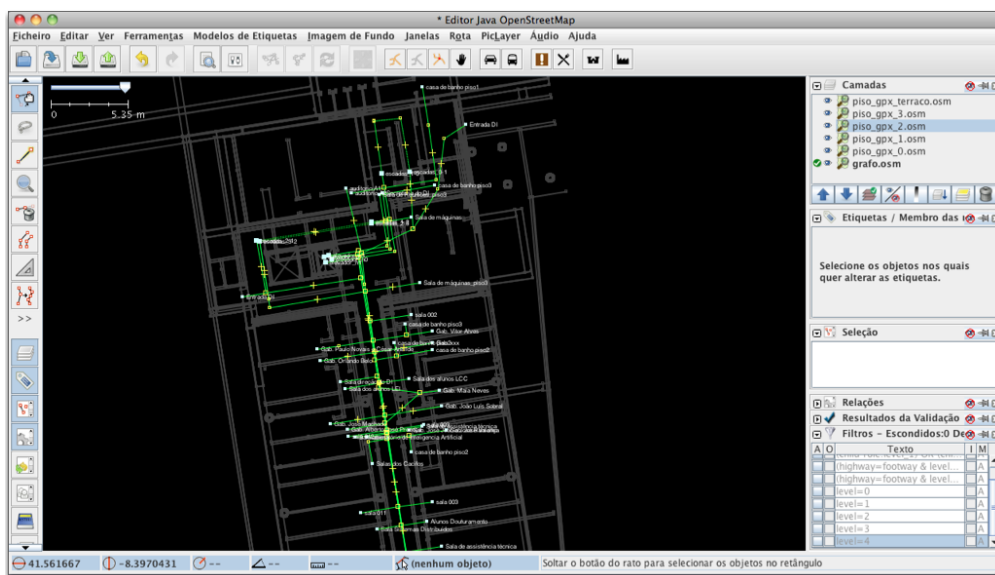


Figura 4.10: Visualização no JOSM do grafo de *routing*.

4.3 Implementação

No desenvolvimento da solução de *routing indoor* proposta nesta dissertação, a escolha de software recaiu na aplicação **OTP**, que devido às suas ferramentas *open source* permitirá a geração do grafo e o cálculo do *routing* (do "melhor caminho" possível entre dois pontos) para ambientes *indoor*.

O **Open Trip Planner**⁴ consiste numa solução *open source* de *routing* multimodal. Isto é, o **OTP** permite calcular o percurso entre uma origem e um destino, recorrendo a vários tipos de transporte, incluindo transportes públicos (como comboios e autocarros, por exemplo). Este projeto foi lançado em 2009, e desde então, o seu sucesso é evidenciado pelo cada vez maior número de utilizadores e colaboradores no seu desenvolvimento. Encontra-se disponível para plataformas *Linux*, *Windows*, ou potencialmente qualquer plataforma com uma máquina virtual Java. O núcleo do **OTP** é composto por três componentes de software básicos: o *graph builder*, um motor de *routing* e uma interface para o utilizador. Quanto à interface para o utilizador (isto é, o mapa e a representação visual das rotas), esta encontra-se desenvolvida em *JavaScript* e os outros dois componentes estão desenvolvidos na linguagem de programação *Java* recorrendo à plataforma *Java Enterprise Edition* (*Java EE*). Este software gratuito e *open source* é otimizado para ser utilizado com fontes de dados abertos como o **OSM** e o **GTFS** (*General Transit Feed Specification*) para os dados do sistema de trânsito. O **OTP** encontra-se implementado já em diversos países, sendo o primeiro Portland (com a *Portland's TriMet transit agency*), seguido de Estados Unidos da América, Canadá, Polónia, Índia, Espanha, Irlanda e Israel. Existe uma boa documentação sobre o **OTP**, estando disponível na página *web* do software vários tutoriais simples e intuitivos.

Inicialmente a *source* do **OTP** foi importada para o **Eclipse** (ferramenta de desenvolvimento de software, uma *Integrated Development Environment* - IDE), para que desta forma fosse possível efetuar todas as alterações necessárias aquando da criação da solução de *routing indoor* proposta. Estas alterações consistiram na geração de um grafo relativo a este caso de estudo, e alterações a nível da interface gráfica. As mesmas serão descritas nos sub-capítulos seguintes.

⁴www.opentripplanner.com

4.3.1 Geração do Grafo

O software de *routing* utiliza um grafo que necessita ser previamente calculado. Este grafo especifica todos os locais de uma determinada região, e como ir de um local para outro. A geração do mesmo poderá demorar algum tempo, e consoante os dados que se queiram incluir, torna-se uma operação mais ou menos complexa.

No presente caso de estudo, foi calculado um grafo para a cidade de Braga. A criação do grafo encontra-se determinada por um ficheiro de configuração XML, designado de `graph-builder.xml`, onde são definidos os diferentes tipos de dados necessários a incluir no grafo e onde este será armazenado. Para o presente caso foram inseridos os dados relativos ao sistema de trânsito (ficheiro GTFS), os dados OSM, e por fim, os dados correspondentes à *altimetria* da cidade de Braga. De seguida, é descrito mais detalhadamente cada um destes dados e o respetivo módulo de importação.

GTFS

A inclusão de transportes públicos, como autocarros, comboios ou barcos, tem que contemplar os horários dos mesmos, tornando o algoritmo de *routing* ainda mais complicado, mas também mais interessante. Esta inclusão é conseguida definindo um ficheiro GTFS (em formato `.zip`) com os dados relativos ao sistema de trânsito (os horários dos transportes e a informação geográfica associada). Isto possibilita, por exemplo, o cálculo de um percurso cuja a origem é a universidade e o destino a casa do utilizador, apanhando um autocarro.

Este ficheiro GTFS, no presente caso designado por `'tub.zip'`, foi previamente preparado e indicado posteriormente no ficheiro de configuração do grafo. O módulo de importação dos dados GTFS necessário inserir no ficheiro de configuração encontra-se representado no Código 4.3. O ficheiro `tub.zip` é composto por oito ficheiros, seis deles requeridos e dois opcionais. Todos os ficheiros apresentam-se em formato texto (`.txt`) e encontram-se relacionados uns com os outros, contendo toda a informação necessária aos transportes públicos, neste caso a autocarros. São eles:

1. `agency.txt` (requerido) - define uma ou mais agências de transporte que fornecem os dados para esta fonte de informação (*feed*).
2. `stops.txt` (requerido) - define as diversas paragens (*stops*) onde os transportes públicos pegam e deixam os passageiros.

3. `routes.txt` (requerido) - define as rotas dos diferentes transportes. Uma rota consiste num grupo de viagens disponíveis para os passageiros como um único serviço.
4. `trips.txt` (requerido) - define as viagens para cada rota. Uma viagem consiste numa sequência de duas ou mais paragens que ocorre num período específico de tempo.
5. `stop_times.txt` (requerido) - define os tempos em que cada transporte chega e parte de uma paragem, para cada viagem.
6. `calendar.txt` (requerido) - define as datas para cada serviço usando o calendário semanal. Especifica quando o serviço começa e acaba, bem como os dias da semana em que o serviço se encontra disponível.
7. `calendar_dates.txt` - (opcional) - define as exceções dos serviços estabelecidos em `calendar.txt`. Por exemplo, os dias de feriado em que determinado serviço não se encontra disponível.
8. `shapes.txt` (opcional) - define as regras para desenhar linhas sobre o mapa, que representam as rotas de uma empresa de transporte.

```

1 <bean id="gtfsBuilder" class="org.opentripplanner.graph_builder.impl.
  GtfsGraphBuilderImpl">
  <property name="gtfsBundles">
3   <bean id="gtfsBundles" class="org.opentripplanner.graph_builder.model.
    GtfsBundles">
    <property name="bundles">
5     <bean class="org.opentripplanner.graph_builder.model.GtfsBundle">
      <property name="path" value="/Users/Nair/Desktop/BUSbraga/tub.zip" />
7     </bean>
    </property>
9   </bean>
  </property>
11 </bean>

```

Código 4.3: Módulo de importação do ficheiro GTFS.

OSM

Relativamente aos dados [OSM](#), é indicado no `graph-builder.xml` um ficheiro local com os dados do [OSM](#) que se pretende e necessita utilizar. Este ficheiro local, relativo

à cidade de Braga, foi obtido como já descrito no sub-capítulo 4.2.3. O módulo de importação dos dados *OSM* que foi necessário inserir no ficheiro de configuração do grafo encontra-se representado no Código 4.4.

```

1 <bean class="org.opentripplanner.graph_builder.impl.osm.
  OpenStreetMapGraphBuilderImpl">
  <property name="provider">
3    <bean class="org.opentripplanner.openstreetmap.impl.
      AnyFileBasedOpenStreetMapProviderImpl">
        <property name="path" value="/Users/Nair/TeseIndoor/pisos_osm/braga.osm" />
5      </bean>
    </property>
7  <property name="defaultWayPropertySetSource">
    <bean class="org.opentripplanner.graph_builder.impl.osm.
      DefaultWayPropertySetSource" />
9  </property>
</bean>

```

Código 4.4: Módulo de importação do *OSM*.

Altimetria

O *Open Trip Planner* pode também utilizar a informação do declive das vias no cálculo de um percurso. Por exemplo, no cálculo de percursos em bicicleta, podem ser preteridas certas vias em detrimento de outras com menor inclinação, mesmo que o resultado implique uma maior distância.

Assim, de forma a se ter acesso a esta informação da elevação da área, foi necessário preparar previamente, usando o software *Quantun Gis*, um *geotif* designado por 'mdt20121021.tif' com os dados da altimetria de Braga, e posteriormente, indica-lo no ficheiro de configuração do grafo. No Código 4.5 é possível observar o módulo de importação dos dados de altimetria necessário inserir no ficheiro de configuração do grafo.

```

<bean id="nedBuilder" class="org.opentripplanner.graph_builder.impl.ned.
  NEDGraphBuilderImpl">
2  <property name="gridCoverageFactory">
    <bean class="org.opentripplanner.graph_builder.impl.ned.
      GeotiffGridCoverageFactoryImpl">
4      <property name="path" value="/Users/Nair/TeseIndoor/mdt20121021.tif" />
    </bean>
6  </property>
</bean>

```

Código 4.5: Módulo de importação do ficheiro de altimetria.

No final, o processo de geração do grafo deve terminar com as mensagens ilustradas no Código 4.6.

```
1 20121120,15:26:08,685 INFO [Graph.java:431]:Main graph size: |V|=6399 |E|=16145
20121120,15:26:08,685 INFO [Graph.java:432]:Writing graph /var/otp/graphs/Graph.
  obj ...
3 20121120,15:26:08,718 DEBUG [Graph.java:446]:Consolidating edges ...
20121120,15:26:08,752 DEBUG [Graph.java:456]:Assigning vertex/edge ID numbers ...
5 20121120,15:26:08,774 DEBUG [Graph.java:458]:Writing edges ...
20121120,15:26:09,865 DEBUG [Graph.java:463]:Writing debug data ...
7 20121120,15:26:09,920 INFO [Graph.java:471]:Graph written .
```

Código 4.6: Mensagens finais no processo de geração do grafo.

4.3.2 Interface

É necessária uma nova interface *web* para explorar na totalidade a informação adicional do *indoor*, ou seja, a nova interface deve estar ciente dos dados *indoor* existentes. Desta forma, a nível da interface da aplicação, foram realizadas inicialmente algumas transformações mais simples, com intuito de se conseguir não só uma interface amigável ao utilizador como adequada às características *indoor*.

Em primeiro lugar foi efetuada a mudança da língua da página *web* para Português, e para isso, foi criado um ficheiro *javascript* designado **Portuguese.js**, contendo todas as traduções necessárias. Também foi criado um logo da aplicação **NavIOS**, que se encontra incorporado na página *web* (ver Figura 4.11). A questão da deteção de edifícios com pisos numa certa área, assim como a forma de indicar o ponto de origem e de destino em espaços *indoor*, exigiu alterações mais complexas. Estas alterações serão descritas de uma forma mais detalhada nos sub-capítulos seguintes.

Como se pode ver na Figura 4.11, a interface gráfica principal da aplicação é composta por um painel central onde apresenta o mapa (e posteriormente, a representação visual das rotas), uma barra lateral à esquerda onde o utilizador define os parâmetros para o planeamento da sua viagem, e por fim, o logo da aplicação no canto superior esquerdo.

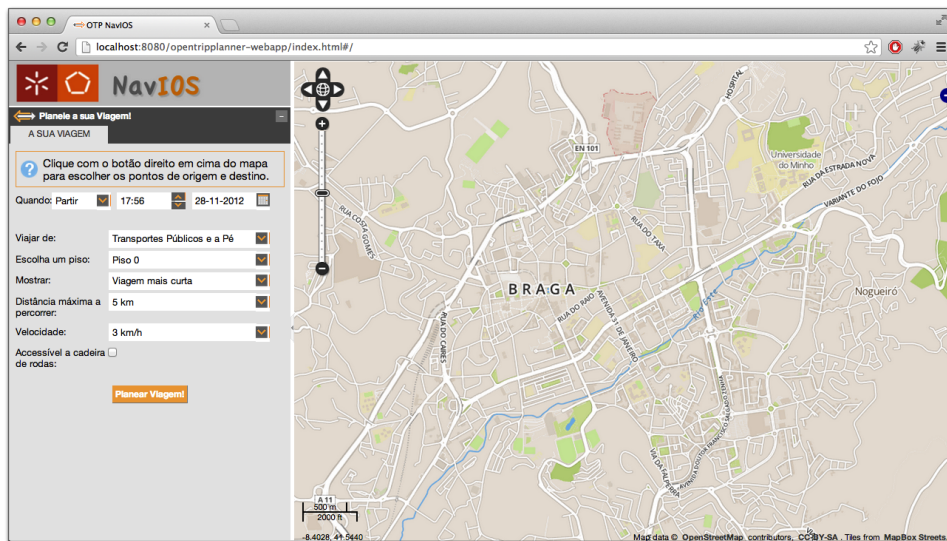


Figura 4.11: Interface gráfica principal da aplicação proposta - NavIOS.

Deteção de edifícios com pisos

Para ser possível identificar os edifícios navegáveis através de diferentes pisos, decidi-se arranjar uma forma de distinguir estes edifícios dos demais existentes. Por um lado, nos dados *OSM*, identificaram-se estes edifícios através da atribuição das seguintes *tags*:

```
building=yes, building:levels=*, building:max_level=*,
building:min_level=*, name=* .
```

Na Figura 4.12 observam-se as *tags* que foram atribuídas para caracterizar o edifício em estudo: o Departamento de Informática. Por outro lado, no código fonte foi necessário efetuar determinadas alterações, para assim se conseguirem identificar no mapa estes edifícios.

Etiquetas: 5 / Membro de relações: 0		
Construção/Construção/Edifício ...		
Chave	Valor	
building	yes	
building:levels	5	
building:max_level	4	
building:min_level	0	
name	Departamento de Informática	

Figura 4.12: *Tags* utilizadas para descrever um edifício.

Estas alterações tornaram então possível verificar, a um determinado nível de *zoom*, se existem numa certa área edifícios com pisos, ou seja, se existe disponível informação *indoor* relativa a esses edifícios. Isto é conseguido realizando uma *query* à XAPI⁵ (*OSM Extended API*), que por sua vez retorna para uma determinada *bounding box* todas as *ways* com as *tags*: `building=yes` e `building:levels=*`. No Código 4.7 observa-se a função que executa a *query* à API e retorna a informação necessária dos edifícios.

```

1 var createUrlBuildings = function(extent) {
    var bbox = extentToBbox(extent);
3  var path = 'http://jxapi.openstreetmap.org/xapi/api/0.6/';
    return path + "way[building=yes][building:levels=*][bbox=" + bbox.left + "," +
        bbox.bottom + "," + bbox.right + "," + bbox.top + " ]";
5 };

```

Código 4.7: Função que efetua a *query* à XAPI para carregar a informação dos edifícios com pisos.

Caso existam edifícios com informação *indoor*, estes são representados no mapa por uns retângulos a cor laranja (ver Figura 4.13), e um novo controlo adicional é criado e ativado na interface da aplicação. Neste caso, o controlo trata-se de uma *combobox* que contém todos os pisos disponíveis relativos ao edifícios identificados na área. Os pisos disponibilizados nesta *combobox*, resultam do cálculo do número mínimo e máximo de pisos entre o edifício ou os edifícios identificados. Por exemplo, dois edifícios são detetados, um com quatro andares (do piso -1 ao piso 2) e outro com seis andares (do piso 0 ao piso 5), assim na *combobox* são disponibilizados os pisos desde o piso -1 (mínimo para este caso) ao piso 5 (máximo para este caso). No Código 4.9 está representada a função responsável pela criação do *layer* relativo aos edifícios (caso estes existam), assim como da criação da *combobox* com os respetivos pisos.

Após ativada a *combobox* torna-se possível selecionar um piso, e consequentemente, visualizá-lo no mapa. Contudo, só após um nível de *zoom* ainda maior é que são carregados para o mapa os nodos relativos ao piso escolhido. A informação de todos os nodos de um determinado piso é conseguida através de uma *query* feita à XAPI, tal como no caso da identificação dos edifícios. Neste caso, a *query* retorna para uma determinada *bounding box* todas os nodos com a *tag*: `level="+ piso + "` (ver Código 4.8). O valor `'piso'` toma o valor do piso anteriormente selecionado na *combobox*. Na Figura 4.13 está representado o resultado obtido para o edifício em estudo, o DI, ilustrando todos nodos

⁵wiki.openstreetmap.org/wiki/Xapi

relativos ao piso 3.

```

1 var createUrlFootway = function(extent, piso) {
    var bbox = extentToBbox(extent);
3   var path = 'http://jxapi.openstreetmap.org/xapi/api/0.6/';
    return path + "node[level=" + piso + "][bbox=" + bbox.left + "," + bbox.bottom +
        "," + bbox.right + "," + bbox.top + " ]";
5 };

```

Código 4.8: Função que efetua a *query* à XAPI para carregar os nodos de um determinado piso.

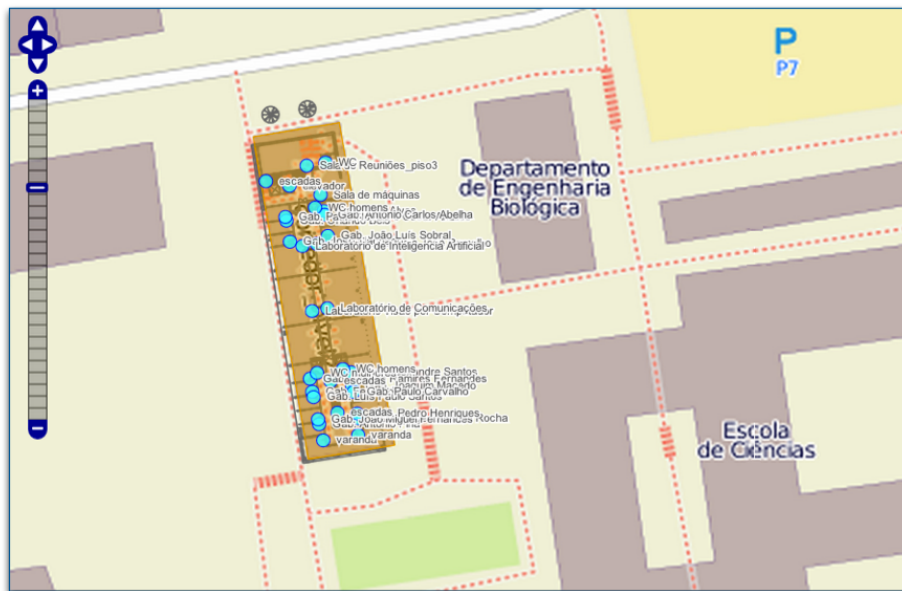


Figura 4.13: Representação da informação *indoor* relativa ao Departamento de Informática, com a visualização dos nodos do piso 3.

A *combobox* mantém-se por defeito seleccionada no piso escolhido, mesmo que o utilizador mova o mapa. Contudo, se este mover o mapa para uma nova área em que não existam edifícios (com a informação dos pisos) a *combobox* fica desativada, até serem novamente identificados edifícios.

```

1 var createLayerBuildings = function(extent, name) {
    var layer = new OpenLayers.Layer.Vector(name, {
3      strategies : [new OpenLayers.Strategy.Fixed()],
      protocol : new OpenLayers.Protocol.HTTP( {
5        url : createUrlBuildings(extent),
        format : new OpenLayers.Format.OSM()
7      } ),
      projection : new OpenLayers.Projection("EPSG:4326")
9    });
    layer.events.register("loadend", layer, function(e) {
11      edificios = e.object;
      var min = 1000;
13      var max = -1000;
      var sel = document.getElementById('selectFloor');
15      if (sel.options.length > 1) {
        sel.options.length = 1;
17        sel.disabled = true;
      }
19      if (edificios.features.length > 0) {
        f = edificios.features[0];
21        min = f.attributes["building:min_level"];
        max = f.attributes["building:max_level"];
23        for ( i = 1; i < edificios.features.length; i++) {
          f = edificios.features[i];
25          x = parseInt(f.attributes["building:min_level"]);
          if (x < min) {
27            min = x;
          }
29          y = f.attributes["building:max_level"];
          if (y > max) {
31            max = y;
          }
33        }
        var opt = null;
35        for ( i = min; i <= max; i++) {
          opt = document.createElement('option');
37          opt.value = i;
          opt.innerHTML = 'Piso ' + i;
39          sel.appendChild(opt);
        }
41        sel.disabled = false;
        if (pisoEscolhido != -1000) {
43          sel.value = pisoEscolhido;
        }
45      }
    }); return layer;
47 };

```

Código 4.9: Função 'criar *layer* dos edifícios e *combobox* com os pisos'.

Identificar os nodos de origem e destino

Como já foi referido, em contexto *indoor*, o processo de *routing* apresenta-se completamente diferente do *outdoor*. Basta pensar que uma pessoa dentro de um edifício pode começar e terminar o seu percurso exatamente no mesmo ponto, mesma latitude e longitude, mas por exemplo três andares acima.

Desta forma, o objetivo principal foi conseguir pôr o *OTP* ciente dos espaços *indoor*. Para tal, foi necessário obter os pontos de origem e destino dentro dos edifícios, isto é, obter os ID's dos nodos em vez da latitude e longitude. Isto é conseguido através da ação do clique do rato feita pelo utilizador. Este interage com a plataforma clicando no ponto de origem, ou seja, no ponto de onde ele pretende partir, e no ponto de destino, o ponto onde ele pretende chegar. São captadas as informações sobre os pontos que o utilizador definiu, e é calculado qual o nodo mais próximo a cada um desses pontos, sendo estes definidos como os nodos de origem e de destino. No Código 4.10 está demonstrada a função que possibilita o cálculo do nodo mais próximo.

```
1 map.events.register('click', map, handleMapClick);
  function handleMapClick(e) {
3     var lonlat = map.getLonLatFromViewPortPx(e.xy);
     var ponto = new OpenLayers.Geometry.Point(lonlat.lon, lonlat.lat);
5     var layer = map.getLayersByName("Footway")[0];
     if (layer) {
7         if (layer.features.length > 0) {
             var min = 1000;
9             var maisperto = null;
             for ( i = 0; i < layer.features.length; i++) {
11                 node = layer.features[i];
                 distancia = ponto.distanceTo(node.geometry);
13                 if (distancia < min) {
                     min = distancia;
15                     maisperto = node;
                 }
17             }
             origem = "osm " + maisperto.fid.replace(".", " ");
19         }
21     }
```

Código 4.10: Função 'cálculo do nodo mais próximo'.

Posteriormente, no processo do pedido do cálculo do *routing*, este é efetuado definindo os nodos (os seus ID's) e não as suas coordenadas, como se pode verificar no pedido exemplificado no Código 4.11. Uma vez tendo todas as informações necessárias definidas

pelo utilizador, a informação é então renderizada no mapa ficando assim disponível para visualização a posição atual do utilizador e todo o caminho a percorrer pelo mesmo até atingir o seu ponto de destino indicado. O resultado da rota é calculado no lado do servidor, e é depois devolvido como uma *WFS* (*Web Feature Service*).

```
1 http://localhost:8080/opentripplanner-api-webapp/ws/plan_dc=1351114743110&arriveBy  
=false&time=22%3A16&ui_date=24-10-2012&mode=WALK&optimize=QUICK&  
maxWalkDistance=5000&walkSpeed=0.833&date=2012-10-24&toPlace=osm%20node  
%201981943264&fromPlace=osm%20node%201981943583
```

Código 4.11: Exemplo de um pedido feito no **OTP**.

Capítulo 5

Testes e Resultados

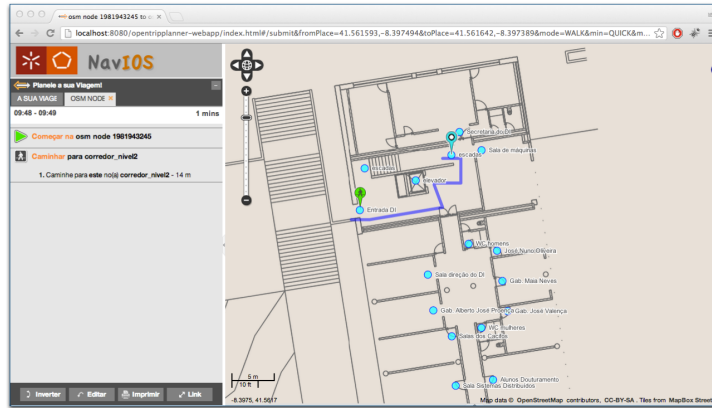
No presente capítulo vão ser apresentados alguns casos práticos com o intuito de comprovar a eficiência da utilização da aplicação NavIOS principalmente a nível de ambientes *indoor*, como também a nível *outdoor* para diferentes planos de viagem. Inicialmente será demonstrado um caso prático para *routing indoor*. De seguida é demonstrado um caso prático mostrando o *routing* com ligação entre *indoor* e *outdoor*. Por fim, será apresentado um caso apenas em *outdoor*. Dependendo de cada situação, vão ser selecionados diferentes tipos de transporte, e para cada um destes, vai ser demonstrada e explicada a rota gerada.

5.1 Caso 1: Perdido no DI

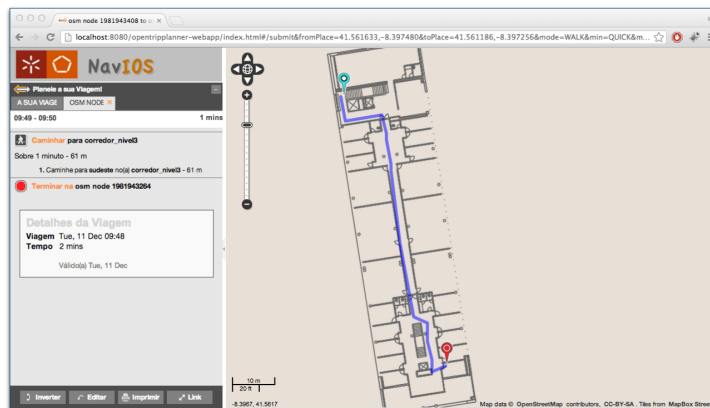
O primeiro caso prático considerado relaciona-se com o Departamento de Informática na Universidade do Minho, tendo como principal objetivo demonstrar o *routing* no interior do edifício, mais precisamente, mostrar as diferenças nas rotas apresentadas entre o tipo de transporte: 'a pé' e em 'cadeira de rodas'.

O Ricardo, um aluno de Sistemas de Informação Geográfica, decidiu tirar uma dúvida com o professor da cadeira, Jorge Gustavo. Uma vez que não conhecia o departamento, o Ricardo não sabia a melhor forma de chegar ao gabinete do professor, apenas sabia que era algures no piso três. Decidiu então utilizar a aplicação NavIOS. Assim, precisou de definir os pontos de origem e destino na aplicação. Definiu o ponto de origem como o ponto onde se encontrava, a entrada do DI (no piso 2), e o ponto de destino como o gabinete do professor, depois de o visualizar no mapa. O resultado que a aplicação

devolveu representa a rota mais curta que o Ricardo pode percorrer para chegar ao gabinete do professor. Este resultado encontra-se demonstrado na Figura 5.1.



(a)



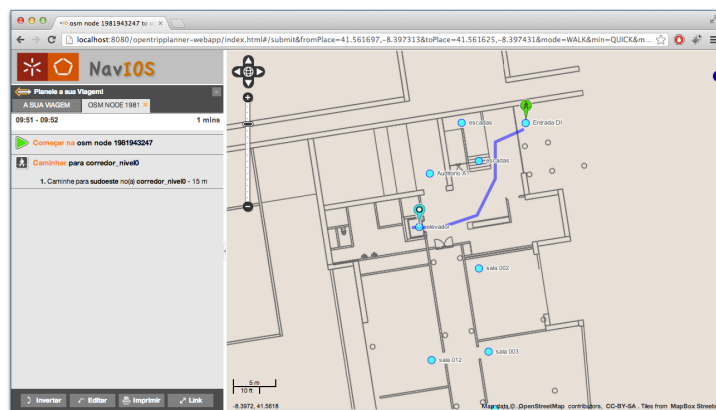
(b)

Figura 5.1: Routing que apresenta o caminho mais curto para o tipo de transporte 'a Pé': (a) visualização da parte da rota relativa ao piso 2; (b) visualização da parte da rota relativa ao piso 3.

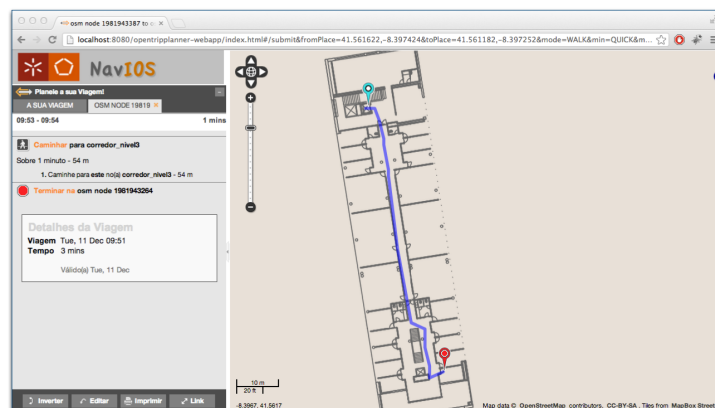
Observando a figura, é possível visualizar que o Ricardo terá que subir até ao piso 3 pelas escadas mais próximas (à sua esquerda). Na rota as escadas são representadas como um ponto intermédio (pino azul) entre os dois pisos. Uma vez no piso 3, o Ricardo segue o trajeto apresentado, percorrendo todo corredor até ao gabinete do professor. A sua viagem demora no total 2 minutos, tendo que percorrer 14 metros no piso 2, e 61 metros no piso 3.

Nesta situação, assumindo que o Ricardo utiliza as escadas, teria dois caminhos

possíveis. Poderia subir as escadas no início do piso 2 e percorrer todo o piso 3 até ao gabinete, ou poderia percorrer todo o piso 2 e só depois subir outras escadas até ao piso 3, ficando assim perto do gabinete. Idealmente, estes dois caminhos seriam equivalentes, ou seja, levariam o mesmo tempo ao Ricardo, e fariam com que ele percorresse a mesma distância. No entanto, na realidade, o grafo não foi construído de forma simétrica, uma vez que foi desenhado manualmente e a simetria se tornava difícil de atingir. Assim, uma das rotas poderá levar mais ou menos tempo, dependendo de alguns nodos do grafo pertencentes ao trajeto que estejam mais longe ou mais perto do destino.



(a)



(b)

Figura 5.2: Routing que apresenta o caminho mais curto para o tipo de transporte 'cadeira de rodas': (a) visualização da parte da rota relativa ao piso 0; (b) visualização da parte da rota relativa ao piso 3.

O professor Jorge tinha também uma outra aluna, a Renata, que possui deficiências motoras, e portanto, ela necessita de utilizar o elevador para se deslocar até ao gabinete. Nesta situação, o ponto de origem está situado na entrada do DI (no piso 0) e o ponto de destino no gabinete do professor Jorge. Uma vez que a Renata não pode subir escadas, escolheu a opção 'Acessível a cadeira de rodas'. A aplicação devolveu a rota mais curta que a Renata tem que percorrer até chegar ao gabinete. Esta encontra-se representada na Figura 5.2.

Observando a figura, é possível visualizar que a Renata após entrar no DI pela entrada do piso 0 terá que se dirigir ao elevador. Na rota o elevador é representado como um ponto intermédio (pino azul) entre os dois pisos. Uma vez no elevador, esta tem que subir até ao piso 3, e posteriormente sair nele. Por fim, terá que percorrer todo corredor do piso 3 até ao gabinete do professor. A sua viagem demora no total 3 minutos, tendo que percorrer 15 metros no piso 0, e 54 metros no piso 3.

5.2 Caso 2: Apanhar autocarro para a Av. Central

Neste segundo caso prático considerou-se uma ida até ao centro comercial Bragashopping, situado no centro de Braga. Por exemplo, a Rita, uma aluna da Universidade do Minho após as suas aulas de informática no DI e a fim de aproveitar o resto do seu dia, decide ir fazer umas compras até ao Bragashopping no centro da cidade. Desta forma, o ponto de origem considerado encontra-se situado na entrada do DI e o ponto de destino no shopping. Na Figura 5.3 é representado o trajeto da rota mais curta usando os tipos de transporte 'a pé' e 'autocarro'.

Observando a figura é possível verificar que a Rita terá que fazer uma parte do seu percurso a pé (partes da rota a azul) e outra apanhando o autocarro (parte da rota a vermelho). Ou seja, a Rita tem que caminhar até à paragem da Universidade do Minho, apanhar o autocarro 43, e sair depois na paragem 'João XXI' perto da escola Carlos Amarante, continuando o resto do percurso a pé. Sai da universidade às 13:07, conseguindo chegar ao centro às 13:45, tendo a viagem uma duração total de 37 minutos. No painel inferior da interface da aplicação pode-se observar, aquando do percurso realizado a pé, os dados de altimetria associados a esse percurso.

Três horas depois, a Rita lembra-se que se esqueceu de tirar umas fotocópias para uma aula do dia seguinte, e por isso, tem que voltar à universidade. Agora, o ponto de origem considerado está situado no Bragashopping e o ponto de destino na universidade (mais

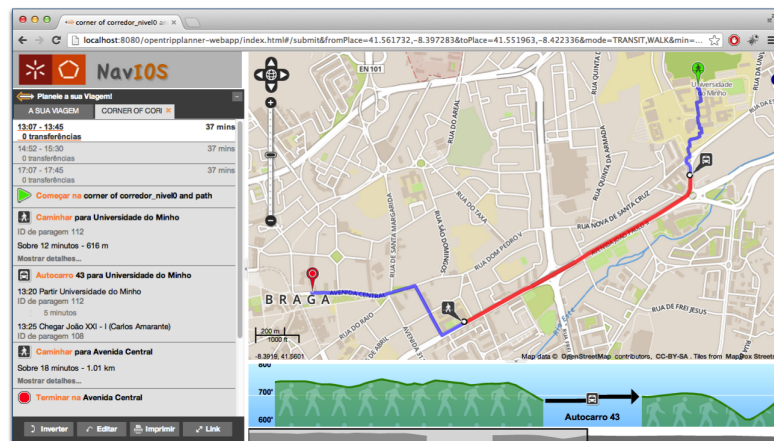


Figura 5.3: Routing que apresenta o caminho mais curto para o tipo de transporte 'Autocarro + a Pé'.

propriamente no DI). Este cálculo do caminho inverso é possível (e mais rápido) com um simples click no botão 'Inverter', situado no canto inferior esquerdo da interface, que faz rapidamente a troca do ponto de origem pelo destino e vice-versa, possibilitando planeamento inverso. A Figura 5.4 ilustra o trajeto inverso da rota mais curta que a Rita tem que percorrer agora para chegar a universidade, utilizando os tipos de transporte 'a pé' e 'autocarro' novamente.

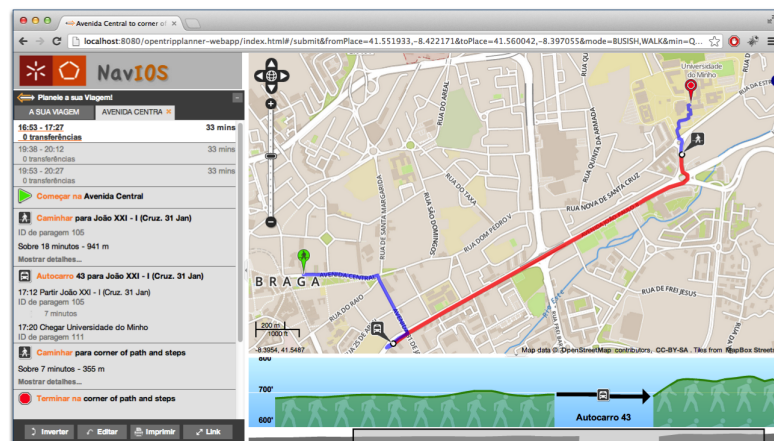


Figura 5.4: Routing que apresenta o caminho mais curto para tipo de transporte 'Autocarro + a Pé'.

Como demonstra a figura, a Rita terá que realizar o seu percurso mais uma vez a pé e uma parte deste apanhando o autocarro. Sai às 16:53 do centro da cidade, apanhando

pouco tempo depois o autocarro 43 na paragem 'João XXI' perto do cruzamento da Rua 31 Janeiro, e no fim, sai na paragem da universidade chegando por volta das 17:27, demorando assim a sua viagem na totalidade 33 minutos.

5.3 Caso 3: Passeio de bicicleta em Braga

Neste último caso prático foi considerado um passeio de bicicleta por Braga, mais precisamente uma ida ao Santuário do Bom Jesus do Monte. O Rui, um estudante irrequieto que mora perto da universidade, gosta muito de andar de bicicleta e decidiu ocupar a sua tarde indo dar um passeio até ao Bom Jesus. Assim, o ponto de origem considerado situa-se na casa do Rui, perto do campus universitário e o ponto de destino encontra-se no Bom Jesus. O Rui, como gosta muito destes passeios de bicicleta, preferiu personalizar a sua viagem em vez de escolher o caminho mais curto. Como se pode verificar na Figura 5.5, os parâmetros escolhidos pelo Rui foram uma viagem de bicicleta em que esta fosse 60% segura, 30% rápida e 10% sem morros.

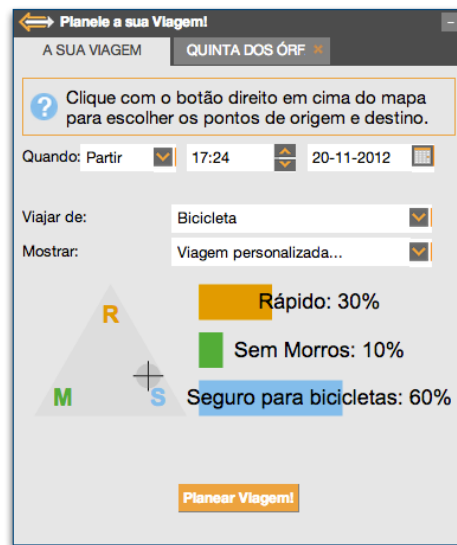


Figura 5.5: Parâmetros escolhidos aquando do planeamento de uma viagem personalizada para o tipo de transporte 'bicicleta'.

O trajeto da rota personalizada, pelo Rui, usando o tipo de transporte 'bicicleta' é então demonstrada na Figura 5.6. Observa-se que para chegar ao seu destino, o Rui terá que percorrer 4.59 km e a sua viagem terá uma duração de 48 minutos. No painel

inferior da interface da aplicação podem-se observar os dados de altimetria associados ao percurso. Verifica-se que o ponto mais alto, em todo caminho, encontra-se quase no final do percurso (ponto com o pino verde no mapa, e risco cinza no painel inferior).

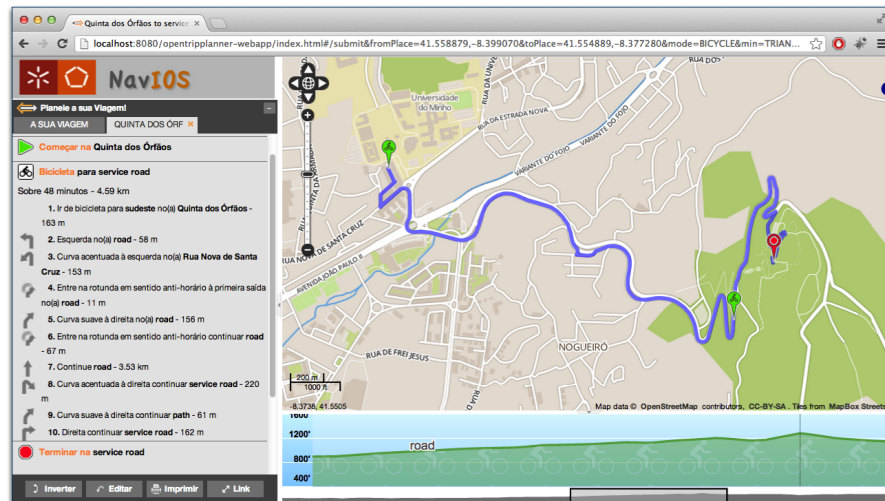


Figura 5.6: Routing que apresenta a viagem personalizada para o tipo de transporte 'bicicleta'.

No fim, o Rui gostou muito do passeio, e por isso, decide partilhar a rota que fez aos amigos e à sua namorada. Para os amigos, este não perde tempo e envia lhes logo um e-mail com o link da rota realizada. Isto é possível com um click no botão 'Link' situado no canto inferior esquerdo da interface, que rapidamente mostra o endereço URL para o respetivo plano de viagem. Para a namorada, como ia estar com ela mais tarde, preferiu imprimir a sua rota para lhe mostrar, conseguindo isto com a simples ação do botão 'Imprimir'. Este botão, também situado no canto inferior esquerdo da interface, como se vê na figura 5.6, possibilita a impressão de uma versão do plano de viagem (sem mapa).

Capítulo 6

Conclusão e Trabalho Futuro

6.1 Conclusão

Esta dissertação focou-se na resolução do problema da navegação em espaços *indoor*. Este é um problema que tem surgido frequentemente nos últimos tempos, não só porque as pessoas ocupam cada vez mais o seu tempo em espaços fechados, como também porque se tem verificado uma contínua urbanização e um crescente número de edifícios de grande dimensões. Por outro lado, ao contrário do *outdoor* que já possui uma solução global, o *GPS*, os espaços *indoor* não têm ainda um método que seja consensual, apesar de já existirem diversas iniciativas, comerciais ou não. Do estudo realizado, foi possível concluir que nenhuma solução se destacava perante as restantes, depois de analisadas várias características de cada uma delas. Desta forma, a chave para a resolução deste problema não passa apenas por um método isolado, mas sim por uma combinação de técnicas, variando consoante o contexto e requisitos de cada caso específico.

Em termos de representação de espaços *indoor*, verificou-se que ainda não existe uma única solução. O modo de representação em *2D* é o mais usual e perceptível ao utilizador, já a representação *3D* tem vindo a aumentar, tendo a ela várias vantagens associadas. Contudo, uma das suas desvantagens prende-se com a sua complexidade para o utilizador comum, uma vez que este não se encontra habilitado para navegar e compreender os modelos *3D*. Foi então necessário avaliar uma forma de como capturar os espaços *indoor*, de como classificá-los (ou seja, definir a sua semântica) e o modo de como obter uma solução de *routing* que seja ciente destes mesmos dados.

Quanto à questão da captura do espaço *indoor* foi disponibilizado para este trabalho um modelo *3D* do Departamento de Informática (edifício em estudo), concebido através

de uma ferramenta profissional: o *Revit*. O passo seguinte foi transformar esta representação em objetos mais ou menos sofisticadas no *OSM*. Para isso, foram importados para o *OSM* os dados *indoor* (número de pisos, divisões, entre outros) relativos ao edifício, com a semântica adotada. Foi também construído o grafo de *routing 2D* a partir dos dados *OSM*, onde se marcaram explicitamente as rotas para todos os caminhos possíveis dentro do edifício.

Após todas estas transformações de dados foi então possível, com base nos dados provenientes do *OSM*, e usando uma solução *open-source* de *routing* multi-modal - o *OpenTripPlanner*, criar uma solução para responder a toda a problemática envolvendo a navegação *indoor*. Esta solução, designada por *NavIOS*, é representada por uma plataforma *web* desenvolvida para possibilitar a navegação do utilizador em espaços fechados, permitindo também a navegação no *outdoor* utilizando diferentes tipos de transporte (*routing* multi-modal).

Conclui-se então que é possível criar soluções para a representação e navegação em espaços *indoor*, objetivo que foi atingido com sucesso neste trabalho. O software de *routing* *OTP* revelou-se uma ferramenta útil e acessível para atingir este objetivo. Mostrou-se uma boa escolha, uma vez que foi possível pô-lo ciente da representação dos espaços *indoor*. Assim, conseguiu-se alcançar o sucesso em relação aos objetivos propostos, com exceção da visualização e navegação em *3D*, uma vez que o tempo para a elaboração desta dissertação não foi suficiente para que esta informação fosse disponibilizada na plataforma criada.

A solução proposta revelou-se viável para o caso de estudo: o Departamento de Informática. No entanto, esta solução não está restrita a este edifício, sendo uma opção válida para qualquer outro tipo de edifícios. Um dos exemplos onde a navegação *indoor* pode ter uma grande utilidade são as instituições de saúde, nomeadamente os hospitais. Estes são edifícios normalmente de grandes dimensões, comparáveis mesmo a pequenas cidades, onde um tipo de solução que permita a pacientes, profissionais da instituição, ou simples visitantes se orientarem, é vista como uma mais-valia. Este tipo de solução poderia assim ser utilizado por exemplo, no novo Hospital de Braga, dadas as suas grandes dimensões e o facto de ser novo, leva a que grande parte das pessoas que lá vão necessitem deste tipo de orientação *indoor*.

Uma vez que esta solução foi baseada completamente em software *open-source*, conclui-se que qualquer pessoa pode construir uma solução para a navegação *indoor*, sendo apenas necessário introduzir os seus dados no *OSM*, conforme os requisitos já descritos neste projeto. Isto traduz-se numa grande vantagem deste tipo de arquitetura, apresentado neste

trabalho, em relação a outras soluções proprietárias.

Por fim, destaca-se o facto de que esta plataforma desenvolvida estará disponível na internet para a navegação dos utilizadores. Esta é outra das vantagens deste tipo de solução, estando disponível em qualquer local onde exista ligação à internet.

6.2 Trabalho Futuro

Como perspectiva de trabalho futuro, apontam-se ainda alguns tópicos a realizar, por forma a aperfeiçoar e otimizar a solução criada. Assim, numa possível continuação deste trabalho, a primeira característica a desenvolver seria a disponibilização dos dados 3D dos edifícios para visualização e navegação dentro do mesmo. A solução deverá apresentar a vista 3D do edifício, assim como as rotas possíveis para percorrer determinado trajeto num grafo em 3D. Assim, todas as vantagens da representação em 3D passariam a estar presentes e disponíveis para os utilizadores, complementando desta forma a informação já disponível em 2D.

Em seguida, seria útil tornar disponível na plataforma uma opção de procura de serviços, tendo em vista a localização rápida e eficiente de serviços de forma automática. Um utilizador poderia procurar, com esta funcionalidade, um serviço dentro do Departamento de Informática, tal como a procura de uma sala de estudo. Generalizando estas funcionalidades a outros edifícios, poderíamos considerar a procura de uma biblioteca dentro de um campus universitário ou a procura de um bar ou serviço multibanco dentro de uma unidade hospitalar de grandes dimensões. É uma opção simples na plataforma mas de extrema utilidade para qualquer utilizador destes sistemas de navegação.

Seria também uma mais-valia para a plataforma melhorar o aspeto visual da mesma, tentando disponibilizar uma melhor organização das funcionalidades com o propósito de a poder tornar ainda mais intuitiva.

Por fim, com o crescente número de smartphones e aplicações *mobile* nos últimos tempos, qualquer desenvolvimento neste tipo de ambientes móveis é também um aspeto interessante, uma vez que tornariam a navegação *indoor* acessível de qualquer parte do edifício, e de forma rápida e cómoda.

Bibliografia

- [1] M. Schafer, “PiNav - Pedestrian Indoor Navigation,” Setembro 2012. [Online]. Available: <http://www.rcs.ei.tum.de/research/pinav-pedestrian-indoor-navigation/>
- [2] S. Beauregard, Haas, and Wpnc, “Pedestrian dead reckoning: A basis for personal positioning,” *Proceedings of the 3rd Workshop on Positioning, Navigation and Communication - WPNC*, 2006.
- [3] M. Le, D. Sarahas, and N. Webb, “Indoor Navigation System for Handheld Devices,” Master’s thesis, Bachelor’s thesis, Worcester Polytechnic Institute, 2009.
- [4] H. Nguyen, “Nokia Indoor Navigation: Next Level of Location-Based Apps,” Setembro 2012. [Online]. Available: <http://www.ubergizmo.com/2011/04/nokia-indoor-navigation/>
- [5] Libramation, “Near Field Communication,” Setembro 2012. [Online]. Available: www.libramation.com/Pages/nfc.htm
- [6] T. N. B. S.A., “NFC: conheça a tecnologia que muda a interação com objetos,” Setembro 2012. [Online]. Available: <http://tecnologia.terra.com.br/noticias/0,,OI4930762-EI15607,00-NFC+conheca+a+tecnologia+que+muda+a+interacao+com+objetos.html>
- [7] H. Kawaji, K. Hatada, T. Yamasaki, and K. Aizawa, “An Image-based Indoor Positioning for Digital Museum Applications,” *16th International Conference on Virtual Systems and Multimedia*, pp. 105–111, 2010.
- [8] G. GIScience, “Indoor OSM - Mapping the Indoor World (2D),” Setembro 2012. [Online]. Available: <http://indoorosm.uni-hd.de/>
- [9] GIScience, “Indoor OSM - Mapping the Indoor World (3D),” Setembro 2012. [Online]. Available: <http://indoorosm.uni-hd.de/3d/>
- [10] OpenGeo, “Open Geo: University of Klagenfurt Campus GIS,” Setembro 2012. [Online]. Available: <http://opengeo.org/gallery/klagenfurt/>
- [11] NTNU, “Campus Guiden, Gløshaugen campus of NTNU,” Setembro 2012. [Online]. Available: <http://www.campusguiden.no/?lang=english>.

- [12] D. Barber, J. Mills, and E. Heritage, “3D Laser Scanning for Heritage: Advice and guidance to users on laser scanning in archaeology and architecture,” *English Heritage*, 2007.
- [13] R. Desai, “Nokia leads the way with Indoor Mapping,” Julho 2012. [Online]. Available: conversations.nokia.com/2012/07/16/nokia-leads-the-way-with-indoor-mapping
- [14] H. Koyuncu and S. H. Yang, “A Survey of Indoor Positioning and Object Locating Systems,” *Journal of Computer Science*, vol. 10, no. 5, pp. 121–128, 2010.
- [15] Y. Gu, A. Lo, and I. Niemegeers, “A Survey of Indoor Positioning Systems for Wireless Personal Networks,” *IEEE Communications Surveys & Tutorials*, vol. 11, no. 1, 2009.
- [16] N. Kohtake, S. Morimoto, S. Kogure, and D. Manandhar, “Indoor and Outdoor Seamless Positioning using Indoor Messaging System and GPS,” *International Conference: IPIN2011*, pp. 21–23, 2011.
- [17] M. Goetz and A. Zipf, “Open Issues in Bringing 3D to Location Based Services (LBS) - A Review Focusing on 3D Data Streaming and 3D Indoor Navigation,” *Archives*, vol. XXXVIII, pp. 121–124, 2010.
- [18] B. Link, P. Smith, and K. Wehrle, “FootPath: Accurate Map-based Indoor Navigation Using Smartphones,” *International Conference on Indoor Positioning and Indoor Navigation (IPIN’11)*, Guimarães, Portugal, 2011.
- [19] A. Mulloni and D. Wagner, “Indoor Positioning and navigation with Camera Phones,” pp. 22–31, 2009.
- [20] I. Micello, “Micello: mapping the great indoors,” Setembro 2012. [Online]. Available: <http://www.micello.com/>
- [21] S. L. Ting, S. K. Kwok, A. H. C. Tsang, and G. T. S. Ho, “The Study on Using Passive RFID Tags for Indoor Positioning,” vol. 3, no. 1, pp. 9–15, 2011.
- [22] S. House, S. Connell, I. Milligan, D. Austin, T. Hayes, and P. Chiang, “Indoor Localization using Pedestrian Dead Reckoning updated with RFID-based fiducials,” *33rd Annual International Conference of the IEEE EMBS, Boston, Massachusetts USA*, pp. 7598–7601, August 30 - September 3 2011.
- [23] G. Gartner, A. Frank, and G. Retscher, “Pedestrian navigation systems in mixed indoor/outdoor environment - the NAVIO project,” *9th International Symposium on Planning & IT (CORP & Geomultimedia)*, pp. 165–171, 2004.
- [24] F. H. Pinto, “An Indoor Localization Solution for Mobile Devices,” Master’s thesis, FEUP, Faculdade de Engenharia da Universidade do Porto, June 2011.
- [25] O. Mezentsev and G. Lachapelle, “Pedestrian Dead Reckoning - A Solution to Navigation in GPS signal degraded areas?” *GEOMATICA*, vol. 59, no. 2, pp. 175 – 182, 2005.

- [26] J. kim, J. H. J., D. Hwang, and C. Park, "A Step, Stride and Heading determination for the Pedestrian Navigation System," *Journal of Global Positioning System*, vol. 3, no. 1-2, pp. 273–279, 2004.
- [27] J. Collin, O. Mezentsev, and L. G., "Indoor Positioning System using Accelerometry and High Accuracy Heading Sensors," *Proceedings of GPS/GNSS, Conference (Session C3), Portland, The Institute of Navigation*, 2003.
- [28] S. Beauregard, "A Helmet-Mounted Pedestrian Dead Reckoning System," *Proceedings of the 3rd International Forum on Applied*, pp. 1 –11, march 2006.
- [29] M. Kouroggi, T. Ishikawa, Y. Kameda, J. Ishikawa, K. Aoki, and T. Kurata, "Pedestrian Dead Reckoning and its applications," *Proceedings of "Let's Go Out" Workshop in conjunction with ISMAR*, 2009.
- [30] Macslocum, "Mobile accelerometers and gyroscopes explained," Setembro 2012. [Online]. Available: <http://answers.oreilly.com/topic/1751-mobile-accelerometers-and-gyroscopes-explained/>
- [31] Wikipedia, "Gyroscope," Setembro 2012. [Online]. Available: <http://en.wikipedia.org/wiki/Gyroscope>
- [32] J. Saarinen, J. Kappi, and J. Syrjarinne, "Mems-imu based pedestrian navigator for handheld devices," *Technical report, Salt Lake City University of Technology*, 2001.
- [33] R. W. Levi and T. Judd, "Dead Reckoning Navigational System using Accelerometer to Measure Foot Impacts," *U.S. Patent Number 5,583,776*, 1996.
- [34] V. Gabaglio and B. Merminod, "Real-time Calibration of Length of Steps with GPS and Accelerometer," *In Global Navigation Satellite System, GNSS, Genova, Italy*, vol. 2, pp. 599–605, 1999.
- [35] D. Bernstein and A. Kornhauser, "An Introduction to Map Matching for Personal Navigation Assistants," *New Jersey TIDE Center, Institute of Technology Newark*, 1996.
- [36] T. Hollerer, D. Hallaway, N. Tinna, and S. Feiner, "Steps Toward Accommodating Variable Position Tracking Accuracy in a Mobile Augmented Reality System," *AIMS'01: Second Int. Workshop on Artificial Intelligence in Mobile Systems, Seattle*, pp. 31–37, 2001.
- [37] P. Pombinho, A. P. Afonso, and M. B. Carmo, "Indoor Positioning using a Mobile Phone with an Integrated Accelerometer and Digital Compass," *INForum, II Simpósio de Informática*, pp. 443–446, 2010.
- [38] H. Liu, S. Member, H. Darabi, P. Banerjee, and J. Liu, "Survey of Wireless Indoor Positioning Techniques and Systems," vol. 37, no. 6, pp. 1067–1080, 2007.

- [39] J. Torres-Solis, T. H. Falk, and T. Chau, "A review of indoor localization technologies: towards navigational assistance for topographical disorientation," *Bloorview Research Institute & University of Toronto Canada*, 2010.
- [40] F. Evennou, F. Marx, and E. Novakov, "Map-Aided Indoor Mobile Positioning System using Particle Filter," in *Wireless Communications and Networking Conference, IEEE*, vol. 4, march 2005, pp. 2490–2494.
- [41] Z. Xiang, S. Song, J. Chen, H. Wang, J. Huang, and X. Gao, "A wireless LAN - based indoor positioning technology," *IBM J. RES. & DEV.*, vol. 48, no. 56, 2004.
- [42] J. A. Silva, M. J. Nicolau, and A. D. Costa, "WiFi Localization as a Network Service," *International Conference on Indoor Positioning and Indoor Navigation (IPIN), Guimarães, Portugal*, 2011.
- [43] P. Bahl and V. N. Padmanabhan, "RADAR: An In-Building RF based user Location and Tracking System," *Proc. IEEE INFOCOM*, vol. 2, no. 775-784, 2000.
- [44] I. Ekahau, "Ekahau: Wi-Fi Tracking Systems, RTLS and WLAN Site Survey," Setembro 2012. [Online]. Available: <http://www.ekahau.com/>
- [45] W. Chai, J. Zhou, C. Chen, H. Nies, and O. Loffeld, "Continuous Indoor Localization and Navigation Based on Low-cost INS/Wi-Fi Integration," *International Conference on Indoor Positioning and Indoor Navigation (IPIN), Guimarães, Portugal*, 2011.
- [46] A. Pomp, "Indoor Navigation: comparing different indoor location determination approaches," *RWTH Aachen University Germany*, 2011.
- [47] S. Gezici, Z. Tian, G. Giannakis, H. Kobayashi, A. Molisch, H. Poor, and Z. Sahinoglu, "Localization via Ultra-WideBand Radios: a look at positioning aspects for future sensor networks," *Signal Processing Magazine, IEEE*, vol. 22, no. 4, pp. 70–84, 2005.
- [48] F. Elbahhar, B. FALL, M. Heddebaut, A. Rivenq, and J.M. Rouvaen, "Indoor Positioning System based on the UWB technique," *International Conference on Indoor Positioning and Indoor Navigation (IPIN'11), Guimarães, Portugal*, 2011.
- [49] J. Hightower, R. Want, and G. Borriello, "SpotON: An indoor 3D location sensing technology based on RF signal strength," *Univ. Washington, Seattle, Tech. Rep. UW CSE00-02-02*, 2000.
- [50] L. M. Ni, Y. Liu, Y. C. Lau, and A. P. Patil, "LANDMARC: Indoor location sensing using active RFID," *Wireless Netw.*, vol. 10, no. 6, pp. 701–710, 2004.
- [51] S. Ting, S. Kwok, A. Tsang, and G. Ho, "The Study on Using Passive RFID Tags for Indoor Positioning," *International Journal of Engineering Business Management*, vol. 3, no. 1, pp. 9–15, 2011.

- [52] R. Drisgill, “Bluetooth: How it Works,” Setembro 2012. [Online]. Available: <http://www.bluetooth.com/Pages/How-It-Works.aspx>
- [53] Y. Wang, S. Shi, X. Yang, and A. Ma, “Bluetooth Indoor Positioning using RSSI and Least Square Estimation,” *Second International Conference on Future Computer and Communications, Shanghai, China*, 2010.
- [54] A. Bekkelien, “Bluetooth Indoor Positioning,” Master of Computer Science, University of Geneva, March 2012.
- [55] N. Forum, “NFC Forum: About NFC,” Setembro 2012. [Online]. Available: <http://www.nfc-forum.org/home/>
- [56] B. Ozdenizci, K. Ok, V. Coskun, and M. N. Aydin, “Development of an Indoor Navigation System Using NFC Technology,” no. April, pp. 25–27, 2011.
- [57] O. A. Aider, P. Hoppenot, and E. Colle, “A model-based method for indoor mobile robot localization using monocular vision and straight-line correspondences,” *Robotics and Autonomous Systems*, vol. 52, pp. 229–246, 2005.
- [58] J. Krumm, S. Harris, B. Meyers, B. Brumitt, M. Hale, and S. Shafer, “Multi-camera multi-person tracking for EasyLiving,” in *Third IEEE International Workshop on Visual Surveillance*, 2000, pp. 3–10.
- [59] R. Boris, K. Effrosyni, and D. Marcin, “Mobile Museum Guide based on Fast SIFT Recognition,” *6th International Workshop on Adaptive Multimedia Retrieval*, pp. 26–27, 2008.
- [60] H. Bay, T. Tuytelaars, and L. V. Gool, “SURF: Speed-Up Robust Features,” *9th European Conference on Computer Vision*, pp. 404–417, 2006.
- [61] M. Werner, M. Kessel, and C. Marouane, “Indoor Positioning Using Smartphone Camera,” *International Conference on Indoor Positioning and Indoor Navigation (IPIN’11), Guimarães, Portugal*, 2011.
- [62] H. Hile and G. Borriello, “Positioning and Orientation in Indoor Environments Using Camera Phones,” *Computer Graphics and Applications, IEEE*, vol. 28, no. 4, pp. 32–39, 2008.
- [63] G.-H. Kim, J.-S. Kim, and K.-S. Hong, “Vision-based Simultaneous Localization and Mapping with Two Cameras,” *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1671–1676, 2005.
- [64] M. Hofmam, F. G. Carvalho, F. Szenberg, A. Raposo, and M. Gattass, “Um Estudo sobre Marcas Fiduciais em Realidade Aumentada: Combinando Detecção de Linhas com Calibração de Câmera,” 2006.
- [65] C. Owen, F. Xiao, and P. Middlin, “What is the best fiducial?” in *Augmented Reality Toolkit, The First IEEE International Workshop*, 2002, p. 8.

- [66] G. Kim and E. M. Petriu, "Fiducial Marker Indoor Localization with Artificial Neural Network," *IEEE/ASME International Conference on Advanced Intelligent Mechatronics, Montréal, Canada*, pp. 961–966, 2010.
- [67] Piranha-Internet, "What is a QR Code?" Setembro 2012. [Online]. Available: <http://www.whatisaqr.com/>
- [68] L. Y. H. Zizhang, "3D Indoor Navigation: a Framework of Combining BIM with 3D GIS," *44th ISOCARP Congress*, pp. 1–10, 2008.
- [69] L. Liu and S. Zlatanova, "Towards a 3D network model for indoor navigation," *Taylor & Francis Group, London*, 2012.
- [70] W. Yuan and M. Schneider, "inav: An indoor navigation model supporting length-dependent optimal routing," in *Geospatial Thinking*, ser. Lecture Notes in Geoinformation and Cartography, M. Painho, M. Y. Santos, and H. Pundt, Eds. Springer Berlin Heidelberg, 2010, vol. 0, pp. 299–313.
- [71] N. Lenihan, "A Local Optimal User Position System for Indoor Wireless Devices," Degree of Masters of Engineering, University of Limerick, 2004.
- [72] S. Elwood, "Geographic Information Science: new geovisualization technologies - emerging questions and linkages with GIScience research," *Progress in Human Geography*, vol. 33, no. 2, pp. 256–263, 2009.
- [73] M. F. Goodchild, "Citizens as sensors: the world of volunteered geography," *GeoJournal*, vol. 69, pp. 211–221, 2007.
- [74] OSM_wiki, "OpenStreetMap Statistics," Setembro 2012. [Online]. Available: <http://wiki.openstreetmap.org/wiki/Statistics>
- [75] OSM wiki, "OpenStreetMap Indoor Mapping," Setembro 2012. [Online]. Available: http://wiki.openstreetmap.org/wiki/Indoor_Mapping
- [76] OSM Wiki, "3D Development - OpenStreetMap Wiki," Outubro 2012. [Online]. Available: http://wiki.openstreetmap.org/wiki/3D_Development
- [77] M. Goetz, "Using Crowdsourced Indoor Geodata for the Creation of a Three-Dimensional Indoor Routing Web Application," *Future Internet - Open Access Journal*, no. 4, pp. 575–591, 2012.
- [78] C. Halvorsen, "Campusguiden: En navigasjonstjeneste for innendørs bruk," Master's thesis, Norwegian University of Science and Technology (NTNU), Juli 2011.
- [79] M. Pierrot Deselligny and I. Clery, "APERRO, an open source bundle adjustment software for automatic calibration and orientation of set of images," *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 38, no. 5, 2011.

- [80] E. M. Mikhail, J. S. Bethel, and J. C. McGlone, *Introduction to Modern Photogrammetry*. John Wiley & Sons; I.S.ed edition, 2001, no. 496.
- [81] D. Goodman, “What is Digital Photogrammetry?” , wiseGEEK, Outubro 2012. [Online]. Available: www.wisegeek.com/what-is-digital-photogrammetry
- [82] J. Hays and A. A. Efros, “IM2GPS: estimating geographic information from a single image,” *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–8, June 2008.
- [83] A. Georgantas, M. Brédif, and M. Pierrot-Desseilligny, “An Accuracy Assessment of Automated Photogrammetric Techniques for 3D Modeling of Complex Interiors,” *ISPRS Archives, The XXII Congress of the International Society for Photogrammetry and Remote Sensing*, vol. XXXIX-B3, 2012.
- [84] E. Liscio, “Open Source Tools for 3D Forensic Reconstructions - Part 3,” pp. 1–7, Novembro 2011.
- [85] My3DScanner, “Free Point Cloud Viewers,” Outubro 2012. [Online]. Available: http://www.my3dscanner.com/index.php?option=com_k2&view=item&id=3:free-point-cloud-viewers&Itemid=73
- [86] T. Reenskaug, “MVC model,” Xerox Parc 1978-79, Outubro 2012. [Online]. Available: heim.ifi.uio.no/~trygver/themes/mvc/mvc-index.html

Apêndice A

Publicações

A.1 Localização em Espaços Interiores

Artigo de conferência nacional submetido e aceite para a “4^a edição das Jornadas Software Aberto para Sistemas de Informação Geográfica (SASIG 4)”, que se realizou em Guimarães, Portugal, entre 2 e 4 de novembro de 2011.

Acessível em: www.osgeopt.pt/sites/default/files/files/app_Nairalves.pdf

A.2 OSM Indoor: moving forward

Artigo de conferência internacional submetido e aceite para a “Open Source Geospatial Research and Education Symposium (OGRS2012)”, sessão de *Poster*, que se realizou em Yverdon-les-Bains, Suíça, entre 24 e 26 de outubro de 2012.

OSM indoor: moving forward

Nair Isabel Alves and Jorge Gustavo Rocha
Informatics Department
University of Minho
Portugal

Keywords: *OSM, Indoor navigation, Indoor routing, VGI, 3D*

Introduction

The OpenStreetMap (OSM) project started focused on mapping streets. But as many other successful innovations, it created new possibilities. Beyond streets, the community started to capture other information, to create more than just street maps. For example, initiatives like wheelmap.org enhance the OSM with tools and specific data for wheel drivers.

The OSM proved that Volunteered Geographic Information (VGI) [3] can be used in large scale projects. OSM became a school for neogeographers, where they learn how to produce high value user-generated geographic content. Many techniques and technologies were developed specifically to improve the quality of OSM, and to support the OSM mappers.

In this paper we aim to present a proposal to map indoor spaces in OSM. The goal is to map public indoor spaces, like universities, malls and airports. A public hospital, for example, can have many more people moving around than a small city. This is not the first or the only proposal around. Many people already suggested and developed some support to indoor spaces in OSM. This proposal tries to get the best of former initiatives to establish a common path to move forward.

We start by reviewing some important concepts, obviously starting by defining the meaning and the scope of indoor space and its representation. There are representations based on floor footprints and representations more based on the building model. Either can be more syntactical or semantic oriented. The distinction between indoor spaces and 3D models is necessarily discussed, since the distinction between them is somewhat blurred. We also revisit the commercial notion of 3D maps, used by companies like Google, Microsoft or Nokia in their web mapping solutions. These applications are 3D in the sense that they provide terrain elevation plus the 3D building volumes. There are no building models involved. Just the volume (the shell) of the building is modelled and portrayed with textures. For OSM, there are proposals to add this volumetric information to OSM¹, and Heidelberg OSM 3D² [11] is a good example of such proposal. More sophisticated approaches are also emerging [1,2], using the entire building information model (BIM), instead of just its volume. In this scenario, the indoor space can be completely represented in the building information model. Although we strongly believe that is feasible to have BIM shared on OSM in the short time, we argue that OSM indoor mapping still makes sense, and it will take some time to OSM indoor be completely computed from shared BIM. It is worth to remember that OSM indoor has some advantages: it works pretty well on 2D,

¹ http://wiki.openstreetmap.org/wiki/Indoor_Mapping

² <http://wiki.openstreetmap.org/wiki/OSM-3D>

people are already trained to read floor plans, the existent editors like Potlatch³ and JOSM⁴ can be used to draw floor plans with minimal changes and the existent portrayal technology does not require any changes, since multiple floors can be draw on top of the building bounds. 3D building models are more powerful and better represent the building, since they have more information. But the complexity of 3D BIM is its major disadvantage. All the increased complexity in modelling, representation and manipulation can not be easily transformed in a incredible better visualization. 3D navigation is complex for ordinary users. Users are not trained to navigate and understand 3D models. For that reason, we argue that 2D visualizations of indoor spaces are necessary and are a major enhancement over the existent ODM data. It will take more time to change to overall OSM work flow to support 3D BIM. We also argue that both models can coexist, and we can have indoor support and 3D BIM data simultaneously. The indoor can be handfull to improve the 3D models.

Finally, indoor routing and indoor navigation are also reviewed, to highlight the differences and the relations between them. Any kind of routing can always be reduced to a graph representation, but current available algorithms rely on the distance to populate the graph. For indoor spaces, the same location can be used for an elevator that travels through several floors, for example. The graph resulting from the indoor space must be aware of the distance between nodes on different floors, but on almost the same location. Indoor navigation techniques are also reviewed. All techniques are grouped in Pedestrian Dead Reckoning (PDR), Radio Frequency based (WiFi, GPS+beacon, UWB, RFID, Bluetooth and NFC) and Image Analysis based [4].

OSM Indoor related initiatives

After reviewing the main concepts related to indoor spaces, we analyse three known proposals for OSM Indoor. The scientific literature is scarce about the subject. The only publications we know about are [5,6]. Besides these publications, the best starting point for our research was the OSM Wiki⁵, where mostly of the discussions take place, and where pointers to examples are posted.

The methodology adopted for our analysis was to review and compare each proposal against a list of characteristics, group in four categories:

- space representation,
- supporting tools,
- generation of the routing network (graph) and routing calculation,
- interaction, visualization and exploration.

We also surveyed some commercial solutions, mostly to evaluate some HCI issues, since we don't have access to the underlying implementation.

All the three initiatives are related to university buildings or campus: Heidelberg University, Germany [7,8]; Alpen Adria Campus of the University of Klagenfurt, Austria [10]; and Gløshaugen campus of the Norwegian University of Science and Technology [9].

Proposal

³ <http://www.geowiki.com/>

⁴ <http://josm.openstreetmap.de/>

⁵ http://wiki.openstreetmap.org/wiki/Main_Page

The proposal presented for the OSM indoor support take advantage of experiments already mentioned. The goal is to point way a common way to handle indoor spaces in OSM. If we are able to generate some consensus about a common approach, many more mappers will help to improve the tools and share their knowledge about indoor space representation.

The proposal covers the space representation, all necessary tools, and a new web interface, able to explore the indoor space and to provide the necessary controls to enable route calculation.

Space representation

This proposal is for indoor spaces, and not for 3D BIM. The indoor initiatives reviewed proved us that the current OSM data model is able to represent the indoor space. So, the impact on the overall OSM architecture and work flow is minimal, since the data is stored on the same database. There are no modifications needed on the current API. By the contrary, if we want to represent 3D BIM on OSM, we should move these to another database, with another data model and API. The OSM data model is so simple (nodes, ways and relations), that theoretically we can represent any building with that primitives. But the abstraction level of that representation is not suited. 3D BIM requires a separated database, with models and API that manipulate concepts more closed to the building semantics. The recent proposal [1] goes in that direction, since they propose a new parallel project, OpenBuildingModels for 3D BIM data.

Tagging

For OSM Indoor two major proposals are being discussed on the OSM wiki. One, from the University of Heidelberg, tries to represent more than the indoor space, and has more tags to represent the building details⁶. This proposal almost tries to represent the 3D building model using OSM tags. The other proposal, by User:Saerdnaer, is more simple⁷. We propose that this last proposal be used as the starting point. Some tags from the first proposal can be used, not for OSM indoor, but for OSM 3D. By other words, we should distinguished the minimal tags necessary for indoor from tags than can be used to better extrude 3D buildings (just the shell) from the OSM data.

Tools

Editors can be slightly improved to enhance the representation of indoor spaces. Basically, the existent editors can provide an interface where users can edit each level independently. Additionally, an OpenLayers based editor can be developed, using the simple editing primitives of OpenLayers. While this might not be the best solution, it can be an alternative.

Routing

The routing solution is as simple as possible, and only includes routes by foot or by wheel chairs. As we already said, the only specific requirement is related with the distance weight of distinct nodes at the same location.

From our initial tests, the precision of locations is relevant, since locations are more closed. When generating the network graph, the entire precision of the locations must be considered, without any kind of rounding.

Web map

A new web interface is necessary to fully explore the indoor additional information. The

⁶ <http://wiki.openstreetmap.org/wiki/IndoorOSM>

⁷ http://wiki.openstreetmap.org/wiki/Proposed_features/indoor

new interface should be aware of existing indoor data. At certain zoom levels, if indoor data is available, an additional control is displayed. The number of floors will be used to display the floor switch. To calculate routes, two possibilities are provided. The origin and destination can be pointed on the map, or the user can select the explicit rooms or stores, if they were tagged on the data. The result route is calculated on the server side, and it is returned as WFS feature. Since the sources of this alternative site are open source, users can improve these features.

One additional feature is to use such interface within other web sites, for example, in mall or airport web sites, where they can take advantage of the crowded source data and tools.

Conclusions and outlook

This discussion and distinction between OSM Indoor and OSM support for 3D BIM is very important to guarantee that we focus on OSM indoor without the temptation to include BIM features. In this paper we discussed the difference between both approaches and we argue that both models can co-exist in OSM.

Having stated that, the OSM indoor is a wise compromise, to enhance the OSM, without moving right now to fully support 3D BIM. That will take time, and OSM indoor will be around for a while, since users are more familiar with this kind of space representation.

For OSM indoor, we need to create some broad consensus about the tags, but it would be easier if we provide tools to support and explore that additional information. Together is this proposal, we started an open source implementation of all necessary tools to fully explore the indoor data, as described.

References

- [1] - Uden, M., & Zipf, A. (2012). OpenBuildingModels - Towards a Platform for Crowdsourcing Virtual 3D Cities, 1-17.
- [2] - Navigation, I., & Scanner, T. L. (2008). 3D Indoor Navigation : a Framework of Combining BIM with 3D GIS. *Building*, 1-10.
- [3] - Goodchild, M. F. (2007). Citizens as sensors: the world of volunteered geography. *GeoJournal*, 69(4), 211-221. doi:10.1007/s10708-007-9111-y
- [4]_ Hakan Koyuncu and Shuang Hua Yang. A Survey of Indoor Positioning and Object Locating Systems. *Journal of Computer Science*, 10(5):121{128, 2010.
- [5] - Goetz, M., & Zipf, A. (2010). Extending OpenStreetMap to Indoor Environments : Bringing Volunteered Geographic Information to the Next Level, (Hansen 2004).
- [6] - Halvorsen, C. (2011). Campusguiden - En navigasjonstjeneste for innendørs bruk. Master's Thesis, Norwegian University.
- [7] - Indoor OSM, Applications with IndoorOSM data - *2D Indoor Map with Routing*, Heidelberg. <http://indoorosm.uni-hd.de/>
- [8] - Indoor OSM, Applications with IndoorOSM data - *3D Indoor Model with Routing*, Heidelberg. <http://indoorosm.uni-hd.de/3d/>
- [9] - Campus Guiden, Gløshaugen campus of NTNU, Norwegian. <http://www.campusguiden.no/?lang=english>.

[10] - Open Geo, University of Klagenfurt Campus GIS, Austria.
Retrieved from: <http://opengeo.org/gallery/klagenfurt/>

[11] - Str, B., Telephone, G., Response, K., & Date, D. (2010). University of Heidelberg OGC Services Emergency Routing with OpenLS RouteService 3D Visualisation with W3D in the GEOSS Architecture Implementation Pilot – Phase 3. *Response*, 49(March).