



**TÉCNICO**  
LISBOA

# **Estimação de Topologia de uma Rede de Câmaras Usando Métodos Esparsos**

**Isabel Queirós da Silva**

Dissertação para obtenção do Grau de Mestre em

**Engenharia Electrotécnica e de Computadores**

Orientadores: Prof. Alexandre José Malheiro Bernardino  
Prof. Jorge dos Santos Salvador Marques

## **Júri**

Presidente: Prof. João Fernando Cardoso Silva Sequeira  
Orientador: Prof. Alexandre José Malheiro Bernardino  
Vogal: Prof. Pedro Manuel Quintas Aguiar

**Novembro 2017**



## Agradecimentos

Este trabalho foi parcialmente financiado pela FCT ao abrigo do projeto SPARSIS PTDC/ EEIPRO/ 0426/ 2014.

Em primeiro lugar quero agradecer aos meus orientadores, os professores Alexandre Bernardino e Jorge Salvador Marques, pela sua dedicação e pelos ensinamentos.

Agradeço também à minha família e amigos, de quem tanto apoio recebi ao longo desta etapa. Em particular, ao meu irmão e ao Vicente, pela paciência e pela força que me deram.

A todos, o meu mais sincero obrigada.



## Resumo

Este trabalho estuda a representação topológica de sistemas de vigilância multi-câmara, sem sobreposição, através de grafos, com base nos dados de trajetórias de peões. A informação da topologia de um sistema é muito útil nos problemas de re-identificação. Sabendo quais os percursos possíveis num certo ambiente, quando um peão desaparece da imagem de uma câmara, o número de locais onde poderá reaparecer diminui, tornando o problema mais simples de resolver. Os nós dos grafos que queremos obter serão as zonas de entrada e saída dos campos de visão e as ligações representam os percursos existentes no ambiente considerado. A tese propõe um método automático de estimação de grafos esparsos. Para tal utiliza-se a informação dos percursos verificados no ambiente, que poderá conter erros e, consequentemente, nem todas as ligações verificadas nos dados corresponderão à realidade. O método proposto utiliza estes dados e atribui pesos às ligações, com base no número de ocorrências verificadas em cada percurso e na semelhança das características associadas a cada pessoa, nos momentos de entrada e saída dos campos de visão das câmaras. O método proposto foi testado usando dados gerados por um simulador. Realizámos testes na presença de um ruído fixo nos valores das características e para vários valores de percentagem de erros de falhas de deteção e de falsos positivos.

**Palavras-chave:** sistemas multi-câmara, campos de visão não sobrepostos, topologia, grafos esparsos



## Abstract

This work studies the topological representation of multi-camera, non-overlapping, vigilance systems, through graphs, using data from pedestrian's trajectories. The topological information of a system is very useful for the re-identification problems. By knowing the possible paths in an environment, when a pedestrian disappears from the image of a camera, the number of possible places where he could reappear decreases, making the problem easier to solve. The nodes of the graphs that we want to obtain will be the entry and exit zones of the fields of view of the cameras and the links represent the possible paths in the considered environment. The thesis proposes an automatic estimation method of sparse graphs. For that, we use the data of the verified paths in the environment. These data may contain errors and, consequently, not all the links verified in it will be true. The proposed method uses these data and assigns weights to all the links verified in it, based on the number of occurrences for each path and the similarity of feature vectors associated with each pedestrian, at the moments of entry and exit of the fields of view of the cameras. The proposed method was tested using data generated by a simulator. We performed tests in the presence of a fixed noise in the feature vectors and for several values of the percentage of errors of missed detections and of false positives.

**Keywords:** multi-camera network, non-overlapping fields of view, topology, sparse graphs



# Conteúdo

Agradecimentos . . . . .	iii
Resumo . . . . .	v
Abstract . . . . .	vii
Lista de Tabelas . . . . .	xi
Lista de Figuras . . . . .	xiii
Lista de Abreviações . . . . .	xvii
<b>1 Introdução</b>	<b>1</b>
1.1 Motivação . . . . .	1
1.2 Objetivos gerais . . . . .	2
1.3 Contribuições . . . . .	3
1.4 Estrutura . . . . .	4
<b>2 Estado da arte</b>	<b>5</b>
<b>3 Métodos de estimação de grafos</b>	<b>9</b>
3.1 Formulação do problema . . . . .	9
3.2 Método básico de estimação de topologia . . . . .	11
3.3 Método de estimação de topologia proposto . . . . .	12
3.4 Probabilidades de transição . . . . .	13
3.5 Distribuições temporais . . . . .	14
3.5.1 Modelo das distâncias . . . . .	15
3.5.2 Modelo dos <i>outliers</i> . . . . .	15
3.5.3 Modelo dos pesos . . . . .	16
3.5.4 Modelo dos segmentos . . . . .	16
<b>4 Geração de eventos de vídeo</b>	<b>17</b>
4.1 Geração de eventos através de vídeo . . . . .	17
4.1.1 Descrição da base de dados etiquetada . . . . .	17
4.1.2 Super-câmaras . . . . .	18
4.1.3 Obter entradas e saídas . . . . .	18
4.1.4 Obter os percursos através de uma base de dados etiquetada . . . . .	19

4.2	Geração de eventos através de um simulador . . . . .	20
4.2.1	Gerar percursos sem erros . . . . .	20
4.2.2	Gerar falhas de deteção . . . . .	22
4.2.3	Gerar ruído no vetor de características . . . . .	22
4.2.4	Gerar falsos positivos . . . . .	23
<b>5</b>	<b>Experiências</b>	<b>25</b>
5.1	Características do sistema de câmaras . . . . .	25
5.1.1	Base de dados . . . . .	25
5.1.2	Super-câmaras . . . . .	26
5.1.3	Zonas de entrada e saída . . . . .	27
5.1.4	Grafos reais . . . . .	30
5.1.5	Parâmetros de modelação do sistema . . . . .	30
5.1.6	Grafos ideais . . . . .	36
5.2	Validação . . . . .	37
5.3	Experiências . . . . .	38
<b>6</b>	<b>Resultados</b>	<b>39</b>
6.1	Casos sem erro . . . . .	39
6.2	Casos com ruído nos vetores de características . . . . .	39
6.3	Casos com variações de erros . . . . .	40
6.3.1	Casos com erros de falhas de deteção . . . . .	41
6.3.2	Casos com erros de falsos positivos . . . . .	43
6.3.3	Casos com erros de falhas de deteção e de falsos positivos . . . . .	47
6.4	Considerações gerais . . . . .	50
<b>7</b>	<b>Conclusões e trabalho futuro</b>	<b>53</b>
	<b>Bibliografia</b>	<b>56</b>
<b>A</b>	<b>Conhecimentos teóricos</b>	<b>A.57</b>
A.1	O algoritmo Esperança-Maximização . . . . .	A.57
A.2	Homografia . . . . .	A.60
A.3	Método de medição da altura de uma pessoa através de uma imagem . . . . .	A.61
A.4	Método de estimação da topologia . . . . .	A.63
<b>B</b>	<b>Representação da posição de peões no referencial do mundo real</b>	<b>B.65</b>
B.1	Método de estimação da posição dos pés de um peão . . . . .	B.65
<b>C</b>	<b>Estimação da homografia do plano dos pontos da cabeça</b>	<b>C.69</b>
<b>D</b>	<b>Características do sistema de câmaras</b>	<b>D.71</b>

# Lista de Tabelas

2.1	Comparação da nossa solução com a de três artigos apresentados . . . . .	7
3.1	Exemplo das listas de percursos visíveis e invisíveis . . . . .	11
5.1	Super-câmaras definidas e respetivas câmaras constituintes . . . . .	27
5.2	Dados de percursos usados para obter o modelo das distâncias . . . . .	33
6.1	Valores testados para a probabilidade de falhas de deteção e respetivos valores médios de número de percursos não detetados: valores totais e de percentagem em relação ao número de percursos gerados sem erro . . . . .	41
6.2	Valores testados para a probabilidade de falsos positivos e respetivos valores médios de número de falsos percursos gerados: valores totais e de percentagem em relação ao número de percursos gerados sem erro . . . . .	44
D.1	Dados dos percursos a que vamos aplicar o modelo das distâncias . . . . .	D.73
D.2	Segmentos de percursos escolhidos, respetivas descrições e exemplos. . . . .	D.74



# Lista de Figuras

1.1	Exemplo de um corredor com uma câmara e do respetivo grafo . . . . .	3
3.1	Exemplo de (a) um percurso e dos respetivos (b) grafos dos percursos visíveis e (c) grafo dos percursos invisíveis . . . . .	10
3.2	Histograma dos tempos de transição para um determinado percurso invisível . . . . .	14
4.1	Esquema representativo do referencial da imagem e das coordenadas de posição de uma pessoa . . . . .	18
4.2	Exemplo da representação dos aparecimentos e desaparecimentos na imagem e no mundo real . . . . .	19
4.3	Esquema representativo dos acontecimentos e das respetivas listas dos percursos visíveis e invisíveis que pretendemos obter . . . . .	20
5.1	Planta do 7.º e 8.º andar do Instituto de Sistemas e Robótica, em Lisboa, com a representação das câmaras utilizadas neste trabalho . . . . .	26
5.2	Exemplos de imagens adquiridas pelas câmaras 17, 18, 19 e 40 do <i>HDA dataset</i> . . . . .	26
5.3	Referencial do mundo real com a representação dos polígonos de visibilidade de cada câmara . . . . .	27
5.4	Representação da posição dos peões, no referencial do mundo real, nos momentos em que aparecem e desaparecem dos campos de visão as câmaras 40 e 54 . . . . .	28
5.5	Representação dos pontos centrais das <i>bounding boxes</i> , no referencial do mundo real, nos momentos em que aparecem e desaparecem dos campos de visão as câmaras 17 e 19 . . . . .	29
5.6	Representação das ZES das câmaras 18, 19, 40 e 59 . . . . .	29
5.7	Representação das ZES para a super-câmara <i>S1</i> . . . . .	30
5.8	Grafo real dos percursos invisíveis . . . . .	31
5.9	Grafo real dos percursos visíveis . . . . .	31
5.10	Exemplos de distribuições temporais, obtidas com o EM . . . . .	32
5.11	Modelo das distâncias: ajustes obtidos para os valores da média e da moda dos parâmetros da distribuição Gamma, em função da distância percorrida . . . . .	33
5.12	Modelo dos outliers . . . . .	34
5.13	Exemplo da distribuição temporal, obtidas com os modelos de distâncias, dos <i>outliers</i> e dos pesos . . . . .	35
5.14	Exemplos das distribuições temporais obtidas com o modelo dos segmentos . . . . .	35

5.15	Grafo ideal dos percursos invisíveis . . . . .	36
5.16	Grafo ideal dos percursos visíveis . . . . .	37
6.1	Curvas PR e F obtidas para as ligações invisíveis, com probabilidades de erro $P_{FD} = 0.8$ e $P_{FP} = 0.08$ . . . . .	40
6.2	Curvas PR e F obtidas para as ligações visíveis, com probabilidades de erro $P_{FD} = 0$ e $P_{FP} = 0.08$ . . . . .	41
6.3	Curvas PR e F médias para as ligações invisíveis, obtidas para várias condições de erros de falhas de deteção . . . . .	42
6.4	Curvas PR e F médias para as ligações visíveis, obtidas para várias condições de erros de falhas de deteção . . . . .	42
6.5	Grafo dos percursos invisíveis resultante dos dados das simulações com $P_{FD} = 0.3$ e limiar $L = 1.5$ . . . . .	43
6.6	Grafo dos percursos invisíveis resultante dos dados das simulações com $P_{FD} = 0.8$ e limiar $L = 2$ . . . . .	44
6.7	Curvas PR e F médias para as ligações invisíveis, obtidas para várias condições de erros de falsos positivos . . . . .	45
6.8	Curvas PR e F médias para as ligações visíveis, obtidas para várias condições de erros de falsos positivos . . . . .	45
6.9	Grafo dos percursos visíveis resultante dos dados das simulações com $P_{FP} = 0.025$ e limiar $L = 5$ . . . . .	46
6.10	Grafo dos percursos visíveis resultante dos dados das simulações com $P_{FP} = 0.08$ e limiar $L = 5$ . . . . .	46
6.11	Grafo dos percursos invisíveis resultante dos dados das simulações com $P_{FP} = 0.08$ e limiar $L = 2.5$ . . . . .	47
6.12	Curvas PR e F médias para as ligações invisíveis, obtidas para 9 combinações de erros de falsos positivos e de falhas de deteção . . . . .	48
6.13	Curvas PR e F médias para as ligações visíveis, obtidas para 9 combinações de erros de falsos positivos e de falhas de deteção . . . . .	48
6.14	Grafo dos percursos visíveis resultante dos dados das simulações com $P_{FD} = 0.3$ , $P_{FP} = 0.025$ e limiar $L = 4.5$ . . . . .	49
6.15	Grafo dos percursos visíveis resultante dos dados das simulações com $P_{FD} = 0.8$ , $P_{FP} = 0.08$ e limiar $L = 4.5$ . . . . .	49
6.16	Grafo dos percursos invisíveis resultante dos dados das simulações com $P_{FD} = 0.3$ , $P_{FP} = 0.025$ e limiar $L = 2$ . . . . .	50
6.17	Grafo dos percursos invisíveis resultante dos dados das simulações com $P_{FD} = 0.8$ , $P_{FP} = 0.08$ e limiar $L = 3$ . . . . .	50
A.1	Geometria do cálculo de rácios de comprimentos entre linhas de cena paralelas . . . . .	A.62

B.1	Geometria para estimar posição dos pés através de imagem . . . . .	B.66
D.1	Exemplos de imagens adquiridas pelas câmaras 50, 52, 53, 54, 56, 57, 58, 59 e 60 do <i>HDA</i> <i>dataset</i> . . . . .	D.71
D.2	Representação das ZES de cada câmara . . . . .	D.72
D.3	Representação das ZES das super-câmaras <i>S1</i> e <i>S2</i> . . . . .	D.72



# Lista de Abreviações

**EM** Esperança-Maximização.

**ZES** Zonas de Entrada e Saída.



# Capítulo 1

## Introdução

### 1.1 Motivação

A videovigilância é uma tecnologia que está presente em situações muito variadas [1]. Em ambientes grandes ou complexos é necessário a utilização de múltiplas câmaras. Chamamos a um conjunto de câmaras que monitorizam o mesmo ambiente com um objetivo em comum, *rede* ou *sistema de câmaras* [2]. Estas estão normalmente espalhadas ao longo de uma grande área geográfica, em que o ambiente não é controlado e em que podem ser, intencional ou acidentalmente, movidas, o que torna a sua configuração desconhecida. Na maioria dos casos não há sobreposição dos campos de visão da rede e as observações podem ser largamente separadas no tempo e no espaço. Devido a estas condições o controlo manual do sistema é uma forma ineficiente e não confiável de se monitorizar uma grande região [3], especialmente quando é necessário seguir um objeto. Em resposta a isto, vários sistemas têm sido desenvolvidos de modo a automatizar a tarefa de seguimento de objetos.

A tarefa de seguimento de objetos num ambiente extenso observado por múltiplas câmaras pode ser dividida em três problemas. O primeiro é o de seguir múltiplos objetos dentro do campo de visão de uma só câmara, trabalhando com oclusões parciais ou totais, que podem estar associadas a objetos estáticos ou dinâmicos. O segundo é a combinação das deteções de objetos que estejam a ser simultaneamente observados por várias câmaras, com sobreposição. O terceiro é o de seguimento de objetos quando estes transitam de uma zona observada pelo sistema de câmaras para uma zona não observada, reaparecendo numa outra câmara passado algum tempo, sendo necessária a re-identificação de objetos. A resolução destes problemas permite ao sistema o seguimento contínuo de um objeto à medida que este se move num ambiente, desde o momento em que entra até ao momento em que sai, criando um registo completo das suas atividades. Neste trabalho vamos concentrar-nos no terceiro problema mencionado.

O problema de *re-identificação* é definido como o processo de estabelecer correspondências entre imagens de um mesmo objeto, adquiridas por diferentes câmaras sem sobreposição, em instantes de tempo distintos [4]. Isto permite manter a identidade dos objetos e realizar o seguimento contínuo, mesmo em situações em que o objeto transita temporariamente para zonas não observadas pelo sistema de câmaras. Este problema constitui ainda um grande desafio pois as condições de aquisição das câmaras têm muitas

variações, por exemplo, a nível de iluminação, posição e cor. Todas estas combinadas, traduzem-se em grandes mudanças de aparência nos objetos detetados, o que torna o problema muito difícil de resolver.

Se não tivermos acesso a nenhuma informação sobre o sistema e o ambiente observado, consideramos que um objeto quando desaparece do campo de visão de uma câmara poderá reaparecer num qualquer outro campo de visão, em qualquer instante de tempo. Uma forma de diminuirmos o número de possibilidades nestes casos, e consequentemente facilitar o problema de re-identificação, é adquirirmos conhecimento sobre as relações espaciais entre as várias câmaras. Assim, a informação da topologia de uma rede tem um papel muito importante no seguimento de objetos de um sistema de câmaras sem sobreposição. Nos sistemas de vigilância mais antigos a informação sobre a topologia da rede era especificada manualmente ou derivada da calibração de câmaras, o que constitui um processo bastante demorado e propenso a erros. Para além disso, temos de ter atenção à possibilidade de as câmaras poderem ser movidas ou desligadas durante o período de observação, o que tornaria a informação da topologia obsoleta. Por isso, existe uma necessidade de desenvolver métodos automáticos de estimação da topologia de uma rede de câmaras.

Para aprendermos a topologia de uma rede é necessário termos um grande conjunto de dados sobre o sistema com que estamos a trabalhar. Queremos que o método de estimação de topologia seja capaz de lidar com erros provenientes de algoritmos de seguimento pois pretende-se que estes sistemas funcionem de forma autónoma. Assim, é importante trabalhar com um algoritmo cujo erro se conheça ou que se consiga variar de modo a testar o desempenho do algoritmo. Para conseguirmos calcular o erro de um algoritmo de seguimento para uma determinada sequência de imagens é necessário conhecermos a posição, nas imagens, dos objetos a detetar e a identificação dos mesmos. Isto implica, na maioria dos casos, a deteção manual dos objetos de interesse. Para uma base de dados de tamanho considerável este é um trabalho moroso e pouco motivador.

Outra opção será desenvolver um simulador dos eventos gerados pelo sistema de câmaras. Para construir o simulador é necessário termos um número considerável de deteções, sem erros, de objetos de interesse, de modo a conseguirmos obter modelos do sistema que queremos simular. Assim, um desafio nesta área é ter acesso a bases de dados de tamanho considerável e com objetos de interesse detetados e identificados sem erro, de modo a poder comparar resultados com os dados reais e avaliar os algoritmos desenvolvidos.

## 1.2 Objetivos gerais

O objetivo deste trabalho é desenvolver um método de estimação de topologia de uma rede de câmaras sem sobreposição. Esta será representada por um grafo que descreve o comportamento dos objetos a detetar, observado pela rede. Os nós representam câmaras ou zonas de entrada e saída e as ligações entre eles representam os percursos que os objetos podem percorrer entre os nós representados [5]. O grafo pode ter, ainda, associado probabilidades de transição entre os nós e as distribuições temporais de cada ligação.

Vamos definir os nós como as zonas de entrada e saída (ZES) dos campos de visão das câmaras. Um

exemplo de um grafo possível é apresentado na figura 1.1, onde se representa uma câmara e três ZES. Na situação apresentada as ZES são duas portas e a zona mais perto da câmara, por onde as pessoas podem entrar ou sair do seu campo de visão, cada uma delas correspondendo a um nó do grafo. As ligações representadas entre os nós são os percursos possíveis que a câmara pode observar.

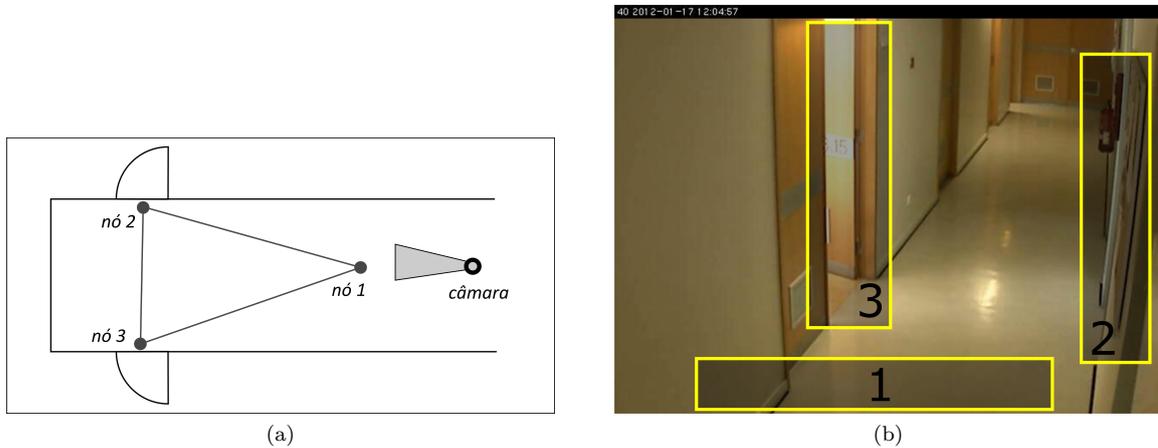


Figura 1.1: Exemplo de uma (a) uma imagem adquirida por uma câmara num corredor e do (b) respetivo grafo, representado sobre a planta do respetivo corredor. Os nós do grafo representam as ZES e as respetivas posições na imagem estão representadas com retângulos amarelos e dos respetivos números identificativos. As ligações do grafo representam os percursos possíveis entre essas zonas.

### 1.3 Contribuições

Uma das principais contribuições deste trabalho é o desenvolvimento de um sistema para estimação automática da topologia de uma rede de câmaras. O sistema foi testado para deteção de pessoas e usou imagens de vídeo capturadas por 13 câmaras e com 83 pedestres, extraídas da base de dados *HDA Person Dataset* [6].

O método de estimação utiliza os percursos de cada pessoa, observados por várias câmaras, admitindo que foi feita a re-identificação correta de cada pessoa, em cada câmara. Os dados a utilizar poderão ter erros de deteção e estudámos o impacto dos mesmos no método de estimação de topologia.

As variações dos erros foram feitas com recurso a um simulador que desenvolvemos, que é também um dos principais contributos deste trabalho. Este algoritmo simula o comportamento de peões observado por um certo sistema de câmaras, gerando um conjunto de percursos, definidos com recurso a eventos de entrada e saída. Simula ainda erros de seguimento, nomeadamente de falhas de deteção e falsos positivos, de magnitude variável.

Este trabalho representa ainda uma contribuição para a base de dados utilizada, o *HDA Person Dataset*, pois foi feita a anotação manual da posição e identidade dos peões de uma das sequências de vídeo utilizada e efetuou-se ainda a identificação de todas as zonas de entrada e saída das 13 câmaras utilizadas.

## 1.4 Estrutura

O capítulo 2 contém o estado da arte, onde apresentamos uma revisão da literatura de estimação de topologia aplicada ao problema de seguimento de objetos numa rede de câmaras. O capítulo 3 descreve a formulação do problema e os métodos de estimação de topologia usados. O capítulo 4 descreve os métodos de geração de eventos de vídeo, nomeadamente os percursos obtidos através de uma base de dados etiquetada e obtidos através de um simulador. O capítulo 5 apresentam as experiências que vamos realizar. Primeiro descrevemos o sistema de câmaras para o qual vão ser realizados os testes, de seguida explicamos como pretendemos validar a solução apresentada e por fim apresentamos os testes que vamos realizar. O capítulo 6 apresenta os resultados obtidos para os diferentes testes realizados. Por fim, o capítulo 7 apresenta as conclusões e o trabalho futuro.

## Capítulo 2

# Estado da arte

Existem vários trabalhos que se dedicam ao problema de estimação da topologia de uma rede de câmaras. Pretendem obter as relações entre as várias câmaras, representando-as com recurso a um grafo. Os nós desse grafo podem representar câmaras ou zonas de entrada e saída, que são zonas do campo de visão das câmaras em que os objetos a detetar podem entrar ou sair do mesmo. As ligações entre os nós representam percursos possíveis entre as zonas representadas. As ligações podem ainda estar associadas a probabilidades de transição entre nós e a distribuições de probabilidade de tempos de transição dos percursos.

Em [2] o autor apresenta uma revisão da literatura do problema de estimação de topologia de uma rede de câmaras, dividindo o problema em casos em que existe sobreposição dos campos de visão das câmaras e casos em que não existe sobreposição. No caso de sobreposição de câmaras é assumido que estas observam a mesma parte de um ambiente, de perspetivas diferentes. O autor diz que este tipo de relação pode ser modelada usando um grafo não-direcional que nomeia de *grafo de visão*. Os nós do grafo representam as câmaras da rede e uma ligação entre nós significa que estas câmaras observam a mesma parte de um ambiente. Em sistemas de câmaras não sobrepostas é assumido que as câmaras observam partes diferentes de um ambiente. As relações entre estas câmaras são representadas por um grafo não-direcional, que o autor nomeia de *grafo de comunicação*. Os nós deste grafo representam as câmaras e as ligações entre dois nós representam a possibilidade de um objeto sair de um deles e reaparecer no outro passado um intervalo de tempo. Os pesos das ligações correspondem a probabilidades e tempos de transição. É possível ainda que um sistema tenha, simultaneamente, câmaras com e sem sobreposição dos campos de visão. O autor fala também de métodos que se dedicam a estimar a topologia da rede nestes casos.

Existem métodos que estimam a topologia de um sistema recorrendo a câmaras calibradas [7]. Neste caso, se a calibração for feita num sistema de coordenadas do mundo em 3D, a topologia poderá ser recuperada diretamente através da análise de geometria e dos campos de visão das câmaras.

Outros métodos tentam recuperar a topologia sem calibração de câmaras. Vamos falar de alguns destes trabalhos, nomeadamente de soluções para redes sem sobreposição ou para redes mistas, ou seja, em que existem câmaras com e sem sobreposição. Podemos dividir estes métodos em duas categorias: métodos

baseados em correspondência [8] e métodos sem correspondência [9]. Os métodos sem correspondência não necessitam de saber se os objetos detetados em diferentes câmaras correspondem ao mesmo objeto ou não, enquanto que os métodos baseados em correspondência necessitam [10]. Este conhecimento pode ser obtido manualmente ou de forma automática através de métodos de identificação.

Em [10] os autores pretendem estimar a topologia de uma rede de câmaras sem sobreposição, utilizando um método sem correspondência. Pretendem obter três fatores de informação da topologia: os nós, as distribuições de tempo de transição e as ligações entre os nós. Definem os nós como as zonas de entrada ou saída, ou seja, zonas do campo de visão das câmaras em que as pessoas podem aparecer ou desaparecer do mesmo; as ligações indicam a conectividade entre dois nós, ou seja, se um objeto desaparecer do nó  $i$  e reaparecer no nó  $j$  então existirá uma ligação entre estes nós; a distribuição do tempo de transição é usada para descrever a probabilidade de um objeto demorar uma certa quantidade de tempo ao mover-se entre nós. As zonas de entrada ou saída dos campos de visão das câmaras são identificadas usando um método de *clustering*. As distribuições de tempo de transição são obtidas através de acumulação de correlação cruzada e ajuste a uma função Gaussiana. Por fim, utilizam informação mútua para excluir falsas ligações. Apresentam resultados experimentais para uma rede simulada com quatro zonas de entrada/saída e para uma rede real com cinco zonas de entrada/saída.

Em [9] os autores usam um método não supervisionado que permite a estimação combinada das relações entre câmaras com e sem sobreposição. Pretendem estimar um grafo em que os nós representam as zonas de entrada e saída e em que as ligações representam as transições entre os nós, que podem ser visíveis ou invisíveis. As transições visíveis são realizadas dentro e as invisíveis fora dos campos de visão das câmaras. É considerado ainda um nó adicional  $k$  que será usado para representar casos em que um objeto desapareceu num nó  $i$  e não voltou a reaparecer, o que resulta numa ligação do nó  $i$  para o  $k$ . Do mesmo modo, se um objeto entra no nó  $i$  sem ter aparecido em nenhum outro nó antes, representaremos este percurso com a ligação do nó  $k$  para o nó  $i$ . Os autores obtêm as zonas de entrada e saída através de um método automático, usando o método de Esperança-Maximização (EM). As ligações visíveis são aprendidas automaticamente através de um algoritmo de seguimento de câmara única ou de múltiplas câmaras sobrepostas. Com os dados do algoritmo de seguimento obtêm percursos e utilizam a primeira e a última deteção de cada um, associando cada uma destas a uma zonas de entrada ou saída. As ligações invisíveis são obtidas através da correlação entre os dados referentes às saídas e às entradas, assumindo que não têm acesso à identificação dos objetos que realizaram cada trajetória. As ligações são representadas no mesmo grafo e estão associadas a tempos ao tempo de transição. Se o tempo de transição for zero significa que as zonas de entrada e saída se encontram sobrepostas; se o tempo de transição for positivo a ligação é invisível e se o tempo de transição for negativo trata-se de uma ligação visível. No artigo foram apresentados resultados para uma rede de 6 câmaras, com dados adquiridos durante um período de 13 horas que resultou em mais de 30000 trajetórias. Os resultados foram considerados favoráveis.

Em [8] os autores propõem um método para o seguimento de pessoas numa rede de câmaras sem sobreposição. Primeiro, na fase de treino, o sistema aprende a topologia da rede ao aprender as distribuições de probabilidade de um objeto entrar numa certa câmara, num determinado instante de tempo, dada a sua localização, tempo e velocidade de saída de uma outra câmara. Utilizam janelas de Parzen, também

Tabela 2.1: Comparação da nossa solução com a de três artigos apresentados, em termos de uso de correspondência de identidades de objetos, de sobreposição de campos de visão, de estimação de ligações visíveis e invisíveis.

Método	Usam Correspondência	Campos de visão sobrepostos	Lig. Invisíveis	Lig. Visíveis
[10]	X	X	✓	X
[9]	X	✓	✓	✓
[8]	✓	X	✓	X
Solução proposta	✓	X	✓	✓

conhecidos como estimadores de densidade de Kernel, e os dados obtidos com algoritmos de seguimento de todas as câmaras. Utilizam também dados sobre a aparência dos objetos para modelar as alterações que acontecem de uma câmara para outra. Após o treino estar completo, usam esta informação para efetuar correspondências entre saídas e entradas verificadas, ou seja, para a re-identificação de objetos. As correspondências são feitas através da estimação de *maximum a posteriori*, com base nos dados aprendidos na fase de treino. Apresentam dados para duas experiências. Uma foi realizada com duas câmaras numa sala e outra para um sistema de 3 câmaras num espaço exterior. O treino durou 10 minutos para ambas as experiências e os resultados foram satisfatórios.

A tabela 2.1 compara a nossa solução com a dos três artigos mencionados anteriormente, resumindo algumas características já mencionadas dos métodos usados pelos autores. Neste trabalho pretendemos estimar a topologia de uma rede sem sobreposição de forma automática, com recurso a correspondência. A topologia é representada por um grafo em que os nós são as zonas de entrada e saída existentes e as ligações representam a possibilidade de um objeto realizar o percurso entre os nós em questão. Obtemos as ligações invisíveis e as ligações visíveis do sistema de câmaras e recorremos a grafos distintos para as representar.

De todos os artigos revistos, a nossa solução é a única que utiliza correspondência e estima ligações visíveis e invisíveis, simultaneamente. A nossa solução utiliza um método de estimação de grafos já existente mas aplicado ao problema em particular de estimação de uma rede de câmaras, algo que não foi ainda testado.



## Capítulo 3

# Métodos de estimação de grafos

O objetivo deste trabalho é representar sistemas de vigilância multi-câmara, sem sobreposição, através de grafos estimados com base nos dados de trajetórias de peões. Os grafos pretendem representar os caminhos possíveis entre um conjunto de zonas do ambiente observado.

Se toda a informação sobre as trajetórias estiver correta, a estimação dos grafos passaria apenas por representar todos os percursos verificados pelos peões. No entanto, na maioria dos casos tal não acontece. Pretende-se, por isso, utilizar um método que, apesar da existência de erros, consiga estimar a topologia correta do sistema de câmaras, sem representar necessariamente todos os percursos detetados.

Para além do conjunto de nós e das ligações representadas nos grafos dos percursos visíveis e invisíveis, é importante conhecer as probabilidades de transição entre nós e as distribuições temporais das ligações. Estes dados, tal como a topologia do sistema, permitem-nos diminuir a ambiguidade existente nos problemas de re-identificação.

Este capítulo encontra-se organizado do seguinte modo: na primeira secção vamos apresentar a formulação do problema, em que apresentamos os dados que assumimos conhecer e definimos alguns conceitos necessários. Na segunda secção vamos apresentar o método básico de estimação de topologia, em que todas as ligações verificadas nos dados são consideradas corretas e representadas no grafo que queremos obter. Vamos explicar que tipo de erros podem estar presentes nos dados que vamos receber e discutir o impacto de alguns desses erros. Na secção três apresentamos o método de estimação de topologia proposto e explicamos como o vamos adaptar ao nosso problema. Na quarta secção discutimos os métodos de estimação das probabilidades de transição entre os nós dos grafos e, por fim, na quinta secção apresentamos as distribuições temporais, que queremos obter para cada ligação representada.

### 3.1 Formulação do problema

Consideremos um trajeto descrito por uma pessoa ao circular num ambiente observado por um conjunto de câmaras sem sobreposição, como por exemplo o da figura 3.1a. Ao longo deste trajeto o peão pode estar em zonas observadas pelo sistema de câmaras ou em zonas não observadas. Assim, uma trajetória pode ser segmentada em partes visíveis e em partes invisíveis. Na figura 3.1a as partes a cinzento

representam as zonas observadas por câmaras. Assim, os segmentos  $A - B$ ,  $C - D$  e  $E - F$  são visíveis e os segmentos  $B - C$  e  $D - E$  são invisíveis. Vamos chamar aos primeiros segmentos de *percursos visíveis* e aos segundos de *percursos invisíveis*.

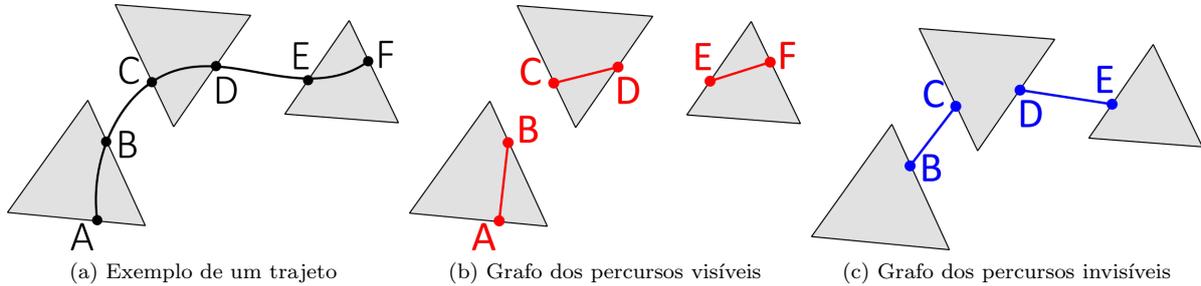


Figura 3.1: Exemplo de (a) um percurso e dos respetivos (b) grafos dos percursos visíveis e (c) grafo dos percursos invisíveis.

O que divide o trajeto nos segmentos apresentados são as entradas e saídas da pessoa dos campos de visão das câmaras. Consideramos que existe um conjunto limitado de zonas do ambiente observado, em que as pessoas podem entrar e sair dos campos de visão. A estas chamamos zonas de entrada e saída. Todas as zonas são simultaneamente consideradas de entrada e de saída, não havendo distinção. É de notar ainda que se uma pessoa aparece ou desaparece da imagem de uma câmara devido a situações de oclusão, permanecendo dentro do campo de visão, estes acontecimentos não constituem entradas ou saídas.

Vamos representar os percursos descritos pelos peões em dois grafos: o grafo dos percursos visíveis e o grafo dos percursos invisíveis. Os nós destes grafos serão as zonas de entrada e saída. As ligações representam os percursos visíveis e invisíveis, respetivamente, verificados entre os nós. Os grafos são unidirecionais pois consideramos que os percursos representados poderão ser feito em qualquer um dos sentidos. Na figura 3.1b e 3.1c estão representados, respetivamente, os grafos dos percursos visíveis e invisíveis para o exemplo da trajetória apresentada na figura 3.1a.

Neste trabalho assumimos que temos acesso ao conjunto de zonas de entrada e saída do sistema com que vamos trabalhar e ainda ao conjunto de percursos visíveis descrito por cada peão. Se um pedestre desaparece do campo de visão de uma câmara, espera-se que reapareça, passado algum tempo, numa câmara do sistema. Assim, com a informação dos percursos visíveis podemos obter também os percursos invisíveis. Cada percurso visível estará associado a uma zona de entrada e a uma zona de saída. Para dois percursos visíveis consecutivos, associados à mesma pessoa, existirá um percurso invisível desde o nó da primeira saída até ao nó da segunda entrada.

Cada percurso visível estará ainda associado ao número de identificação *id* do peão correspondente e ao valor da duração do percurso  $\Delta t$ , que denominamos de tempo de transição. Temos ainda acesso a um vetor de características  $v_{car}$ , representativo do peão em cada nó, ou seja, nos instantes de entrada e saída do percurso. Na prática, o vetor de características poderá ser, por exemplo, representativo da cor do pedestre. Para os percursos invisíveis guardamos a mesma informação.

Na tabela 3.1 estão esquematizadas as listas com os dados dos percursos, para o caso específico dos percursos representados em 3.1, caso tenham sido realizados por um peão com  $id = 1$ . Para cada percurso,

Tabela 3.1: Exemplo das listas de percursos visíveis e invisíveis. Consideramos os percursos representados na figura 3.1, para o caso de terem sido percorridos por um peão de  $id = 1$ .

lista de ligações visíveis						lista de ligações invisíveis					
$id$	nó de entrada	nó de saída	$v_{car}$ de entrada	$v_{car}$ de saída	$\Delta t$	$id$	nó de entrada	nó de saída	$v_{car}$ de entrada	$v_{car}$ de saída	$\Delta t$
1	A	B				1	B	C			
1	C	D	⋮	⋮	⋮	1	D	E	⋮	⋮	⋮
1	E	F									

visível ou invisível, verificado existirá uma ligação na respetiva lista. Se, por exemplo, uma pessoa tiver realizado o mesmo percurso duas vezes, esse aparecerá duas vezes especificado.

O objetivo será utilizar a informação fornecida sobre as zonas de entrada e saída, os percursos visíveis verificados e os percursos invisíveis inferidos para estimar a topologia do grafo dos percursos visíveis e do grafo dos percursos invisíveis. Com os grafos pretendemos representar todos os percursos que possam ser realizados no ambiente observado pelo sistema de câmaras em questão.

Os dados dos percursos visíveis poderão conter erros e o método de estimação de grafos que queremos desenvolver deve conseguir estimar um grafo perfeito, isto é, sem erros, em que todas as ligações representadas correspondem a percursos válidos e reais.

## 3.2 Método básico de estimação de topologia

Nesta secção vamos apresentar o método básico de estimação de topologia, que pode ser usado para dados perfeitos, e discutir qual seria o resultado esperado na presença de erros. Este método limitar-se-ia a aceitar todas as ocorrências presentes nas listas esquematizadas na tabela 3.1 e a representar todas as respetivas ligações nos grafos.

Se os dados a que temos acesso forem perfeitos, nomeadamente as zonas de entrada e saída e as trajetórias dos peões na imagem, então o grafo construído não terá erros. Ou seja, todas as ligações representadas corresponderão a percursos válidos e reais. No entanto, na prática existem erros na obtenção dos dados. O erros podem ocorrer na localização das zonas de entrada e saída, no seguimento, na deteção e na re-identificação de pessoas. Estes geram dados nas tabelas referentes a percursos incorretos.

Vamos discutir o impacto que alguns destes erros teriam nos grafos, caso utilizássemos o método básico de estimação de topologia. Vamos estudar o impacto de três tipos de erros: falhas de deteção e falsos positivos.

As falhas de deteção fazem com que segmentos visíveis do trajeto de um peão não sejam detetados. Assim, não vão ser geradas novas ocorrências de percursos visíveis nem falsas ligações no respetivo grafo. No entanto, ao ignorarmos segmentos visíveis, é possível que novos percursos invisíveis sejam gerados. Vejamos o exemplo da figura 3.1. Se o percurso visível entre os nós  $C$  e  $D$  for ignorado, em vez se obtermos os percursos invisíveis representados na figura, iríamos obter o percurso invisível entre os nós  $B$  e  $E$ .

Os falsos positivos ocorrem quando é feita uma deteção que não corresponde a uma pessoa. Este tipo de erros gera novas ocorrências de percursos visíveis, que podem ou não corresponder a percursos

possíveis. Não correspondendo a um percurso possível, os falsos positivos geram ligações erradas no grafo dos percursos visíveis. Se o percurso gerado devido ao falso positivo for atribuído a uma nova pessoa, este não terá impacto nos percursos invisíveis, visto que essa pessoa só estará associada a uma entrada seguida de uma saída. Se for atribuído a uma identidade existente poderá ser gerado um falso percurso invisível e, conseqüentemente, uma ligação errada no respetivo grafo.

### 3.3 Método de estimação de topologia proposto

Nesta secção vamos apresentar o método de estimação de topologia proposto neste trabalho. O algoritmo proposto permite-nos aprender grafos esparsos com base na comparação de um conjunto de características dos objetos considerados. Este método [11] encontra-se descrito em anexo na secção A.4. Vamos passar a explicar como pretendemos adaptá-lo ao nosso problema, de forma a obter a topologia do sistema de câmaras.

Vamos utilizar os dados das listas representadas na tabela 3.1 para estimarmos a topologia do sistema simulado. O objetivo será obtermos apenas as ligações que representam percursos reais e não representar necessariamente todos os dados presentes nas listas, visto que estas poderão conter informações erradas.

O algoritmo proposto utiliza uma matriz de características  $D$ . As características que vamos utilizar são as fornecidas com os vetores de características. Cada vetor está associado a uma pessoa e a um acontecimento, que pode ser uma entrada ou uma saída, de um determinado percurso. Assim, as características a usar não estão associadas apenas aos objetos considerados, que são os nós, como é feito em [11]. Estão também associadas a uma ligação em particular. Deste modo, é possível estimar as ligações analisando apenas os nós par a par, construindo-se uma matriz  $D$  associada a cada percurso.

Na matriz  $D$  os dados de cada linha são referentes a um nó, para uma ligação em específico. Como estamos a considerar ligações unidirecionais, a direção do percurso não será importante neste caso. Se considerarmos os nós  $A$  e  $B$ , vamos construir a matriz de características com os dados de percursos de  $A$  para  $B$  e de  $B$  para  $A$ . As colunas correspondem aos vetores de características verificados em cada ocorrência, associados ao nó onde foram verificados. Deste modo, um percurso que tenha verificado  $m$  ocorrências e para um vetor de características de dimensão  $j$ , terá uma matriz de características de dimensão  $2 \times jm$ . O algoritmo vai atribuir um peso a cada ligação com base no número de características observadas e na similaridade dos valores de cada característica do vetor, ou seja, de cada coluna.

Como estamos a aplicar o método individualmente a cada ligação, estamos a considerar dois objetos de cada vez, que serão os dois nós do percurso considerado. Assim, a matriz dos pesos  $W$ , calculada com o método considerado, terá dimensão  $2 \times 2$ . Como  $w_{ii} = 0$  e a matriz é simétrica, estamos apenas a encontrar um valor da matriz  $W$  de cada vez que aplicamos o algoritmo. Vamos designar como  $w$  o valor a encontrar. É de notar que os valores dos pesos  $w$  obtidos verificarão sempre  $w \geq 0$ .

De acordo com [11] e como foi mencionado na secção A.4, a matriz  $D$  deve ser escalada de modo a que a média  $D_{media}$  de todos os valores da matriz  $D$  seja zero e que o valor máximo  $cov_{max}$  presente na matriz de covariância  $\frac{1}{m}DD^T$  seja um. Assim, para escalarmos uma matriz  $D$  é apenas necessário substituir-la por  $(D - D_{media})/\sqrt{cov_{max}}$ .

Vamos correr o algoritmo para cada ligação verificada, seja visível ou invisível, e para cada uma vamos obter um valor  $w$  que será o peso da ligação em relação às outras existentes. Uma ligação com um valor de  $w$  maior terá maior peso no grafo respetivo. Os grafos podem ser construídos aceitando ou não uma ligação, tendo em conta o valor  $w$  associado à mesma.

### 3.4 Probabilidades de transição

Queremos ainda calcular as probabilidades de transição entre nós, para um certo grafo obtido com os dados das listas a que temos acesso (esquematisadas na tabela 3.1). Uma probabilidade de transição  $P_{i \rightarrow j}$  representa a probabilidade de uma pessoa, estando no nó  $i$ , transitar para o nó  $j$ . Estes valores, ao contrário das ligações representadas nos grafos, são direcionais. Ou seja, a probabilidade  $P_{i \rightarrow j}$  não será necessariamente igual à probabilidade  $P_{j \rightarrow i}$ . A probabilidade  $P_{i \rightarrow j}$  pode ser calculada através de

$$P_{i \rightarrow j} = \frac{\#i \rightarrow j}{\#i \rightarrow \forall}. \quad (3.1)$$

Em que  $\#i \rightarrow j$  é o número de ocorrências do percurso que começa no nó  $i$  e termina no nó  $j$  e  $\#i \rightarrow \forall$  é o número de ocorrências de todos os percursos que começam no nó  $i$ . Deste modo, a soma das probabilidades de todos os percursos que começam num nó  $i$  será igual a um.

Nas listas a que temos acesso (esquematisadas na tabela 3.1), devido ao número reduzido de dados, existem ligações que só verificaram ocorrências num dos sentidos possíveis. De modo a manter a possibilidade de todas as ligações consideradas poderem ser realizadas em ambos os sentidos, vamos utilizar *add-one smoothing* [12] para o cálculo das probabilidades de transição:

$$P_{i \rightarrow j} = \frac{1 + \#i \rightarrow j}{\#possibilidades + \#i \rightarrow \forall}. \quad (3.2)$$

Em que  $\#possibilidades$  é o número de ligações existentes no grafo cujo nó inicial seja  $i$ . É de notar só vamos calcular a probabilidade de transição do nó  $i$  para o nó  $j$  se a ligação entre estes nós estiver representada no grafo. Caso contrário, a probabilidade será zero. A exceção são as transições para o vazio, ou seja, os casos em que um peão desaparece de um nó e não volta a reaparecer no sistema. Estes casos não estão representados nos grafos mas esta possibilidade vai ser considerada.

Podemos guardar os valores de probabilidades de transição numa matriz esparsa que denominamos de matriz de transição. Teremos uma para os percursos visíveis e outra para os percursos invisíveis. Um valor  $a_{ij}$ , estando na linha  $i$  e coluna  $j$  da matriz de transição, representará a probabilidade de uma pessoa, estando no nó  $i$ , transitar para o nó  $j$ . Se existirem  $N$  zonas de entrada e saída, a matriz de transição dos percursos visíveis terá dimensão  $N \times N$  e a dos percursos invisíveis terá dimensão  $N \times N + 1$ , pois inclui as transições de cada nó para o vazio.

### 3.5 Distribuições temporais

As distribuições temporais fornecem-nos informação sobre a probabilidade de um peão demorar uma certa quantidade de tempo a percorrer um percurso. São representadas com auxílio de funções de distribuição de probabilidade. Cada ligação representada no grafo terá associada uma função, que não dependerá do sentido do percurso. Queremos estimar a função com base nos dados dos tempos de transição verificados. Estes têm uma distribuição semelhante à apresentada na figura 3.2. Estes dados sugerem o uso de uma mistura de densidades.



Figura 3.2: Histograma dos tempos de transição para um determinado percurso invisível.

A maioria das ocorrências estão associadas a tempos de transição menores e com valores próximos uns dos outros. Estas estão associadas a casos em que os peões realizam o trajeto a uma velocidade aproximadamente constante, dependendo a duração do percurso essencialmente da distância percorrida. Consideramos que este é o comportamento *normal*. No entanto, existe sempre a possibilidade de os peões terem outro tipo de comportamento, normalmente associado a valores superiores de tempos de transição. Consideramos estes casos como *outliers*.

As distribuições a usar devem ser definidas apenas para valores positivos, visto que os tempos de transição serão sempre maiores que zero. Para modelar o comportamento *normal* testámos as distribuições Gamma, Poisson, Exponencial e Rayleigh. A Gamma foi a que pareceu fazer um melhor ajuste aos dados e foi portanto a escolhida. Para modelar os *outliers* escolheu-se a distribuição Rayleigh.

Para obtermos a mistura de distribuições vamos usar o algoritmo de Esperança-Maximização (EM). Na secção A.1 apresentamos este método, em particular para o caso de uma mistura de uma Gamma com uma Rayleigh.

Existem alguns percursos, nas listas a que temos acesso, um número muito baixo de ocorrências. Estabelecemos que é necessário termos no mínimo 8 dados para conseguirmos obter resultados fidedignos com o EM. Para todos os percursos com menos de 8 dados será necessário encontrar modelos adicionais de modo a obtermos as funções desejadas.

Os modelos que vamos utilizar baseiam-se na localização dos percursos no ambiente considerado, no comportamento esperado em cada local e ainda nos resultados obtidos com o EM para os percursos com mais de 8 ocorrências. Para alguns percursos existe uma relação linear da distância associada ao percurso com os tempos de transição verificados. Com esta relação conseguimos estimar a distribuição Gamma. Assim, podemos utilizar os dados destas ligações para obter o *modelo das distâncias*. Nos casos

em que queremos utilizar esse modelo para estimar a Gamma, será ainda necessário obter uma estimação da distribuição Rayleigh, que modela os casos *outliers* e ainda atribuir um peso a cada distribuição. A pensar nestes casos desenvolvemos o *modelo dos outliers* e o *modelo dos pesos*. Por fim, existem percursos cuja relação com a distância não é clara. Para esses casos a distribuição é estimada usando o *modelo dos segmentos*.

Nas subsecções seguintes explicamos os métodos desenvolvidos para obter estes quatro modelos.

### 3.5.1 Modelo das distâncias

Verificámos que em alguns percursos os dados referentes ao comportamento *normal* relacionam-se com a distância percorrida. Estes são essencialmente os percursos que passam pelos corredores. Como conseguimos estimar a distância associada a cada um destes percurso, modelámos esta relação, à qual chamámos modelo das distâncias. Para a construção deste modelo vamos considerar apenas as ligações que verifiquem 8 ou mais ocorrências. Utilizamos os valores dos parâmetros  $\alpha$  e  $\beta$  da distribuição Gamma obtida com o EM e também a distância  $d$  associada a cada percurso. Os valores das distâncias são obtidos por observação do referencial do mundo real, com as posições das câmaras e zonas de entrada e saída representadas.

Para cada percurso considerado vamos calcular a média e a moda da distribuição Gamma através das fórmulas:

$$\begin{aligned} \text{Média: } & \alpha\beta, \\ \text{Moda: } & (\alpha - 1)\beta. \end{aligned} \tag{3.3}$$

Cada um destes valores estará associado à distância percorrida pela respetiva ligação. Vamos visualizar a relação entre os valores da média em função da distância percorrida. Com as funções do MATLAB® conseguimos obter o ajuste à reta destes pontos. Obtemos os parâmetros da equação da reta  $m_{média}$  e  $b_{média}$ . O mesmo para os valores da moda, obtendo-se  $m_{moda}$  e  $b_{moda}$ . É de notar que o modelo das distâncias só precisa de ser calculado uma vez.

O modelo será utilizado para obter os parâmetros da distribuição Gamma de todos os percursos que verifiquem menos de 8 ocorrências e para os quais seja possível estimar a distância média percorrida  $d$ . Os parâmetros  $\alpha$  e  $\beta$  que queremos obter são calculados através das equações

$$\begin{aligned} \beta &= (m_{média} - m_{moda})d + b_{média} - b_{moda}, \\ \alpha &= \frac{m_{média}d + b_{média}}{\beta}. \end{aligned} \tag{3.4}$$

### 3.5.2 Modelo dos *outliers*

Nos casos em que pretendemos utilizar o modelo das distâncias, será também necessário aplicarmos um modelo para os *outliers*. Visto que estes não se relacionam com a distância nem conseguimos encontrar

nenhum padrão no seu comportamento vamos obter uma única distribuição Rayleigh, que modele todos os *outliers*.

Para obtermos este modelo utilizamos todos os dados de todas as ligações que tenham verificado 8 ou mais ocorrências. Com estes dados vamos estimar uma distribuição Rayleigh.

Considerando que temos um conjunto  $\mathbf{X} = \{x^1, \dots, x^N\}$ , que resulta da união dos conjuntos de dados de todas as ligações consideradas. Não sabemos exatamente que distribuição gerou cada dado  $x^k$  mas conhecemos as probabilidades  $\pi_1^k$  e  $\pi_2^k$  (dadas pela equação A.4 do EM) de este ter sido gerado pelas distribuições Gamma e Rayleigh, respetivamente.

Queremos utilizar o valores do conjunto  $\mathbf{X}$ , mas queremos dar maior importância aos dados com maior probabilidade de terem sido gerados pela distribuição Rayleigh. Vamos então gerar um novo conjunto de dados  $\mathbf{Y}$  que será utilizado para estimar a nova distribuição. Para cada dado  $x^k$  do conjunto  $\mathbf{X}$  vamos gerar  $100\pi_2^k$  ocorrências desse valor no conjunto  $\mathbf{Y}$ . Sendo  $y^j$  um dado do conjunto  $\mathbf{Y}$  e  $M$  a dimensão desse conjunto, o parâmetro  $\sigma$  da distribuição Rayleigh que queremos obter é calculado através de

$$\hat{\sigma} = \sqrt{\frac{1}{2M} \sum_{j=1}^M y^{j^2}} \quad (3.5)$$

### 3.5.3 Modelo dos pesos

Para obtermos uma mistura utilizando o modelo das distâncias e dos *outliers* é necessário sabermos os pesos a atribuir a cada distribuição e para tal desenvolvemos este modelo. O objetivo é simplesmente calcular o valor  $w_1$  e  $w_2$  dos pesos a atribuir à distribuição Gamma, dada pelo modelo das distâncias, e à distribuição Rayleigh, dada pelo modelo dos *outliers*. Os pesos são calculados através da equação A.6, usando os dados do conjunto  $\mathbf{X}$ , construído na construção do modelo dos *outliers*. Estes pesos são calculados uma única vez e tomam valores fixos para um certo conjunto de dados.

### 3.5.4 Modelo dos segmentos

Em casos em que não é possível obter a distância associada a um percurso ou em que não é esperado que o tempo de transição se relacione com a distância, é necessário utilizar outro modelo. São exemplos casos de percursos nos átrios dos elevadores ou ligações que comecem e terminem no mesmo nó.

Optámos por criar segmentos de percursos, de acordo com a sua localização e do comportamento esperado no local. Estes segmentos vão conter todos os percursos que não se relacionem com a distância, independentemente do número de dados considerado. Um exemplo de segmentos encontrados são todos os percursos que ocorrem no interior de gabinetes. Ou ainda, todos os percursos que ocorrem numa sala de convívio.

Se um segmento tiver 8 ou mais dados, obtemos a função de distribuição de probabilidade do segmento aplicando o EM. Caso contrário não conseguimos concluir nada sobre o comportamento nestes percursos e vamos modelar o segmento utilizando a distribuição Rayleigh obtida com o modelo dos *outliers*.

O modelo de um segmento só será aplicado aos percursos que o constituem que tenham menos de 8 dados. Os restantes serão modelados com o EM, tal como explicado no início desta secção.

## Capítulo 4

# Geração de eventos de vídeo

Para obtermos os eventos das câmaras (ver tabela 3.1) é necessário fazermos o processamento direto do vídeo. Este não é o objetivo do nosso trabalho, portanto obtemos eventos de duas formas distintas: através de uma base de dados etiquetada manualmente e a partir de um simulador.

Com a informação extraída de uma base de dados etiquetada manualmente conseguimos obter os grafos dos percursos visíveis e invisíveis sem erros, aos quais chamamos de grafos ideais. A estes dados chamamos de *ground truth*.

Queremos ainda utilizar a informação da base de dados para modelar o sistema em questão e utilizar esses modelos para gerar mais dados, com um simulador. Este vai também permitir gerar percursos com diferentes tipos de erro. Com os dados obtidos do simulador podemos testar o algoritmo de estimação de topologia proposto. Podemos comparar os grafos obtidos com os grafos ideais e assim avaliar a prestação do algoritmo em diferentes condições de erro.

Neste capítulo primeiro explicamos como obtemos os percursos realizados através da base de dados etiquetada com que vamos trabalhar e, na secção seguinte, explicamos como gerar os eventos através de um simulador.

### 4.1 Geração de eventos através de vídeo

#### 4.1.1 Descrição da base de dados etiquetada

A base de dados com que vamos trabalhar é constituída pelas sequências de imagens de 13 câmaras, em que algumas destas têm campos de visão sobrepostos. Os dados etiquetados manualmente fornecem-nos as deteções dos peões em cada *frame* das sequências. Essas são dadas através de uma *bounding box*, que se encontra ilustrada na figura 4.1, juntamente com o referencial da imagem e os pontos representativos dos pés  $(x_p, y_p)$  e da cabeça  $(x_c, y_c)$  do peão. Cada *bounding box* tem ainda associada o número de identificação do pedestre.

É-nos fornecido o mapa 2D da cena, nomeadamente a posição das câmaras e uma planta simplificada do andar onde estas se localizam. Temos também acesso à homografia do plano do chão para cada câmara. Com esta é possível, dado um ponto do chão no referencial da imagem, obter a sua posição no referencial

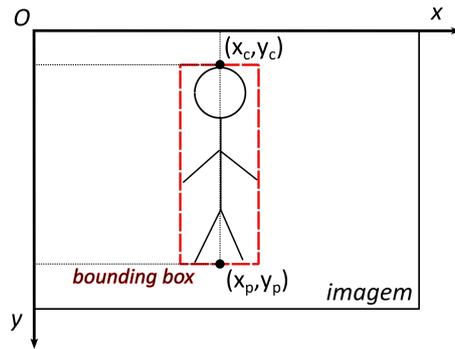


Figura 4.1: Esquema representativo do referencial da imagem e das coordenadas de posição de uma pessoa.

do mundo real. Em anexo, na secção A.2, apresentamos os conhecimentos teóricos sobre a homografia que foram necessários para o desenvolvimento deste trabalho, bem como um método de estimação da homografia, para o caso desta não ser fornecida para alguma câmara.

#### 4.1.2 Super-câmaras

Como queremos trabalhar com um sistema de câmaras sem sobreposição, vamos considerar coleções de câmaras com campos de visão sobrepostos como sendo uma única câmara, que denominamos de super-câmara. Para identificarmos as super-câmaras existentes basta encontrar as câmaras com sobreposição dos campos de visão. Para tal considerámos útil visualizar, no mapa da cena, a zona do chão visível na imagem de cada câmara. A estas zonas chamamos de polígonos de visibilidade e podem ser encontradas aplicando a homografia a pontos que correspondam a vértices da área visível do chão.

Admitimos que, se duas câmaras tiverem os polígonos de visibilidade sobrepostos, farão parte da mesma super-câmara. Podem, no entanto, existir câmaras que constituam uma mesma super-câmara e que não tenham polígonos de visibilidade sobrepostos. É por exemplo o caso quando um peão, num determinado instante de tempo, está completamente visível numa câmara e numa outra tiver apenas a cabeça visível. Nestes casos, os polígonos terão uma localização próxima no mapa da cena e será necessário a visualização das sequências de imagens para se concluir se existe ou não sobreposição.

#### 4.1.3 Obter entradas e saídas

Para obtermos os percursos visíveis e invisíveis verificados na base de dados etiquetada é preciso encontrar e definir as zonas de entrada e saída. Estando as ZES definidas é necessário obter os acontecimentos de entradas e saídas observados e associá-los às respetivas zonas.

Estando a base de dados etiquetada é fácil encontrar as deteções referentes a aparecimentos e desaparecimentos de pessoas e marcar esses pontos na imagem e no mundo real. A representação na imagem é feita utilizando o ponto médio da *bounding box* respetiva. A representação da posição dos peões no referencial do mundo real é feita com recurso à matriz de homografia e à posição dos pés dos pedestres no referencial da imagem. No caso em que os pés dos peões não estão visíveis na imagem é necessário estimá-los. No capítulo em anexo B é feita a descrição do método usado para este efeito.

Na figura 4.2a apresentamos um exemplo da representação dos aparecimentos e desaparecimentos na imagem e na figura 4.2b a representação dos mesmos peões no mundo real.

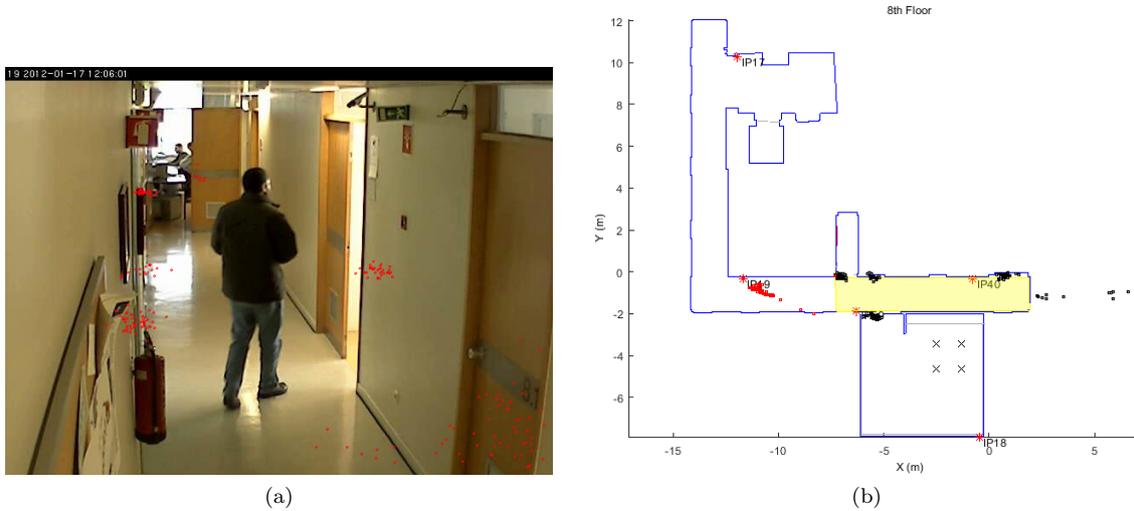


Figura 4.2: Exemplo da representação dos aparecimentos e desaparecimentos na (a) imagem e no (b) mundo real. (a) Os pontos a vermelho representam a posição média das *bounding boxes* referentes a aparecimentos e desaparecimentos. (b) Os pontos a vermelho representam a posição dos peões em situações em que os pés não estavam visíveis na imagem e os pontos a preto representam a posição em situações em que estavam visíveis na imagem; a amarelo está representado o polígono de visibilidade da respetiva câmara.

A partir desta informação definem-se as zonas de entrada e saída manualmente. As zonas de entrada e saída são definidas através de retângulos nos referenciais das respetivas imagens. As ZES das super-câmaras são definidas usando as zonas das suas câmaras constituintes.

Os acontecimentos de entradas e saídas são detetados sempre que uma pessoa apareça ou desapareça numa câmara e esteja numa ZES. Cada entrada estará associada a uma zona de entrada e saída, a um peão e ao número do *frame* em que ocorreu. Estes dados serão guardados numa lista de entradas, uma para cada câmara existente. O mesmo para os dados das saídas. Partimos do princípio que todos os dados etiquetados manualmente estão bem identificados e, conseqüentemente, todas as entradas e saídas obtidas estão corretas. Nos casos em que surjam dúvidas, estas serão esclarecidas por observação das sequências de vídeo.

#### 4.1.4 Obter os percursos através de uma base de dados etiquetada

Com a informação de todos os acontecimentos de entrada e saída, cada uma associada à ZES, ao número de identificação *id* do peão e ao *frame* correspondente, é simples obter as listas pretendidas (ver tabela 3.1). Associamos primeiro cada acontecimento ao instante de tempo em que ocorreu, em segundos, e ordenamos os acontecimentos cronologicamente. Um esquema é representado na figura 4.3 onde é possível visualizar a representação dos acontecimentos num referencial do tempo *t*.

Conhecendo a ordem cronológica com que os acontecimentos sucederam basta associar, para o mesmo peão, uma entrada à saída seguinte, de forma a obtermos um percurso visível. Para os percursos invisíveis associamos uma saída à entrada seguinte, para o mesmo peão.

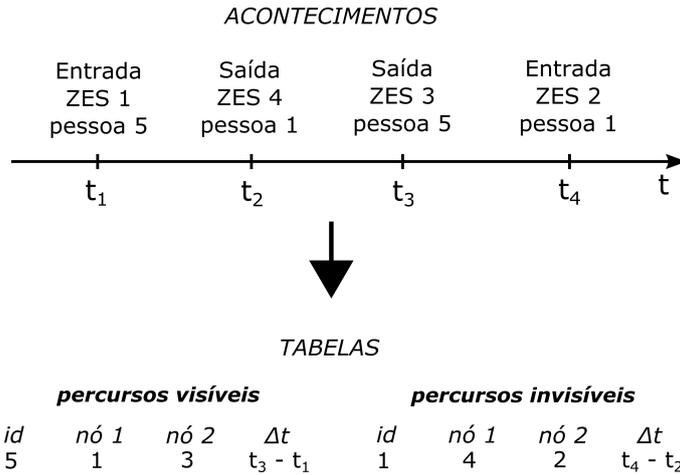


Figura 4.3: Esquema representativo dos acontecimentos e das respetivas listas dos percursos visíveis e invisíveis que pretendemos obter.

Com as listas dos percursos visíveis e invisíveis obtidas podemos visualizar o grafo dos percursos visíveis e dos percursos invisíveis. Como a base de dados etiquetada de onde extraímos estes dados não tem erros, os grafos obtidos só terão ligações correspondentes a percursos reais. Vamos chamar a estes de grafos ideais.

Não vamos extrair da base de dados etiquetada valores correspondentes a vetores de características. Com os dados do *ground truth* (listas e grafos ideais) podemos calcular as probabilidades de transição (ver secção 3.4) e as distribuições temporais (ver secção 3.5).

## 4.2 Geração de eventos através de um simulador

Nesta secção explicamos como são gerados eventos através de um simulador. O objetivo é gerar um conjunto de percursos visíveis e invisíveis, tal como apresentado na tabela 3.1. Queremos que simular o sistema de câmaras considerado e portanto vamos utilizar os valores de probabilidades de transição e as distribuições temporais obtidos com os dados da base de dados etiquetada. Com o simulador pretendemos ainda gerar erros e variar os seus valores de probabilidade. Os erros que queremos gerar são falsos positivos, falhas de deteção e ainda ruído nos vetores de características.

Nas subsecções seguintes vamos explicar como conseguimos simular o sistema considerado para gerar percursos sem erro; depois discutimos a introdução de erros de falhas de deteção; de seguida introduzimos o ruído nos vetores de características e por fim explicamos como geramos erros de falsos positivos.

### 4.2.1 Gerar percursos sem erros

A primeira tarefa do simulador é gerar percursos sem erro, com comportamento semelhante ao do sistema real com que estamos a trabalhar. Para além dos nós existentes no sistema considerado, das probabilidades de transição e das distribuições temporais, o simulador recebe ainda como parâmetros o tempo máximo de entrada  $t_{max}$  e a probabilidade uma nova pessoa entrar no sistema de câmaras por segundo  $p_{nova\ entrada}$ . Neste simulação consideramos que um *frame* equivale a um segundo de aquisição.

Simulamos primeiro a entrada de novas pessoas no sistema: atribuímos a cada pessoa um número de identificação único  $id$  e guardamos os dados relativos às primeiras entradas numa lista chamada de *novas entradas*. Consideramos que a probabilidade de uma nova entrada no sistema é igual para todos os nós. O ciclo encontra-se esquematizado de seguida:

$id = 0$

*novas entradas*  $\rightarrow$  Inicializamos uma lista vazia

**ciclo**  $t = 1 : 1 : t_{max}$

$x \rightarrow$  Gerado aleatoriamente com distribuição uniforme de valores entre 0 e 1

**se**  $x \leq p_{novas\ entrada} \rightarrow$  Geramos uma entrada de uma nova pessoa

$id = id + 1$

$nó \rightarrow$  Selecionado aleatoriamente do conjunto de nós existentes

*novas entradas*  $\rightarrow$  Guardamos os valores de  $id$ ,  $nó$  e de  $t$

Ficamos assim com a informação referente à primeira entrada de todas as pessoas existentes na simulação. Para cada pessoa geramos os restantes acontecimentos:

**Passo 1. Saída** Escolhemos o nó de saída com base nas probabilidades de transição do nó onde o peão se encontra. Considerando que as probabilidades de transição para os nós 1, 2, ..., até ao nó  $n$  são dadas pelos valores  $p_1, p_2, \dots, p_n$ , o nó de saída será selecionado do seguinte modo:

$x \rightarrow$  Gerado aleatoriamente com distribuição uniforme de valores entre 0 e 1

**se**  $0 < x \leq p_1 \rightarrow$  O nó de saída será o 1

**ou se**  $p_1 < x \leq p_1 + p_2 \rightarrow$  O nó de saída será o 2

**ou se**  $p_1 + p_2 < x \leq p_1 + p_2 + p_3 \rightarrow$  O nó de saída será o 3

$\vdots$

Selecionando o nó de saída gerámos um percurso visível. O tempo de transição associado é escolhido com base na distribuição temporal da ligação obtida.

**Passo 2. Entrada** Escolhemos o nó de entrada com base nas probabilidades de transição do nó em que o peão deu saída, da mesma forma que fizemos no passo anterior.

As probabilidades de transição dos percursos invisíveis contam com a possibilidade de um peão, quando realiza uma saída, não voltar a reaparecer no sistema. Se for este o caso selecionado com a escolha do nó, então a simulação para o peão considerado termina, não sendo gerados mais

acontecimentos associados ao mesmo. Se selecionarmos um nó presente no grafo, vamos selecionar também um tempo de transição, tal como fizemos para o passo anterior. A simulação de dados para este peão continua, recomeçando-se o ciclo no *passo 1*.

Vamos ainda simular, para cada pessoa, um vetor de características  $v_{car} = (x_{car}, y_{car})$ . Estes serão gerados aleatoriamente usando uma distribuição uniforme e podem tomar valores entre  $-0.5$  e  $0.5$ . O vetor de características é suposto representar a aparência de cada peão, que poderão ser, por exemplo, características de cor. Como simulamos o caso sem erro consideramos que a característica de cada peão se mantém constante ao longo de todos os acontecimentos. Os vetores de características são importantes pois será usando estes valores que vamos aplicar o método de estimação da topologia proposto.

Vamos guardar todos os dados referentes aos percursos visíveis simulados de acordo com as listas dos percursos visíveis apresentada na tabela 3.1. Estes correspondem aos percursos simulados sem erro. Será a estes dados que vamos aplicar os erros que iremos discutir nas secções seguintes. A partir da lista dos percursos visíveis será sempre possível obtermos a lista dos percursos invisíveis, tal como explicado na secção 3.1.

#### 4.2.2 Gerar falhas de deteção

Num sistema real este erro ocorre quando uma pessoa, presente numa imagem, não é detetada. As falhas de deteção podem gerar erros nos dados com que estamos a trabalhar caso um peão não tenha sido detetado ao longo de todo um percurso realizado. Consideramos pouco provável que o sistema não detete uma pessoa se esta estiver muito tempo presente na imagem, ou seja, se realizar um percurso longo. Assim, as falhas de deteção só vão afetar percursos de curta duração. A duração máxima escolhida de um percurso para que este possa não ser detetado é de dois segundos, que corresponde a três *frames* na nossa simulação.

Recebemos como variável a probabilidade  $P_{fd}$  de um percurso de um *frame* não ser detetado. Um percurso de dois *frames* tem probabilidade  $P_{fd}^2$  de não ser detetado e um de três terá  $P_{fd}^3$ .

Para cada percurso visível com tempo de transição inferior a três segundos vamos sortear, consoante a probabilidade respetiva, se sofrerá ou não do erro de falha de deteção.

#### 4.2.3 Gerar ruído no vetor de características

Queremos também simular ruído nos vetores de características. Recebemos como parâmetro o valor  $r_{car}$ , que será a magnitude do ruído que vamos impor a cada coordenada do vetor de características.

Para cada percurso visível vamos gerar o mesmo ruído nos dois vetores de características associados ao mesmo. Ou seja, mesmo após aplicarmos o ruído, para um percurso visível, o vetor de características associado à entrada será igual ao associado à saída.

Para cada um dos percursos e para cada coordenada do vetor, vamos selecionar aleatoriamente, e com igual probabilidade, o sinal do ruído. Ou seja, se somamos ou se subtraímos o valor  $r_{car}$  a cada coordenada do ponto de cor.

#### 4.2.4 Gerar falsos positivos

Num sistema real por vezes existem deteções que não correspondem, na realidade, a uma pessoa. Para ser gerado um percurso visível devido a um falso positivo, seria necessário que este fosse detetado de forma consecutiva, de uma zona de entrada até uma zona de saída, pois é apenas nestes locais que os percursos se podem iniciar e terminar. No ambiente em que estamos a trabalhar não consideramos a existência de outros objetos móveis a não ser os peões. É então pouco provável que uma falsa deteção se desloque entre zonas de entrada e saída distintas, gerando um percurso visível. Assim, os falsos percursos que vamos simular vão terminar sempre no nó onde começaram. Assumimos também que os falsos positivos não são detetados durante muitos *frames* consecutivos.

Recebemos como parâmetro a probabilidade  $P_{fp}$  de existir um falso positivo no sistema de câmaras, por cada segundo.

Para cada o tempo total em que decorreu a simulação, cujo valor é dado como parâmetro, vamos sortear, por cada segundo, a possibilidade de ser gerado um falso positivo, com probabilidade  $P_{fp}$ . No máximo poderá ser gerado um falso positivo por segundo. Se forem gerados falsos positivos em segundos sucessivos, consideramos que esses correspondem à mesma deteção e resultarão num falso percurso, com a mesma duração que a deteção em questão. Um falso positivo isolado corresponderá a um percurso com duração de apenas um *frame*.

Para cada percurso falso gerado vamos ainda escolher o nó em que ocorreu. Selecionamos um dos nós existentes, todos com igual probabilidade.

É necessário ainda atribuímos uma identidade e um vetor de características ao falso percurso. O vetor de características é gerado de forma aleatória, tal como explicado anteriormente, obtendo-se  $v_{car}^{novo}$ . A identidade do peão será atribuída com base na semelhança deste vetor com os de todos os peões gerados com a simulação sem erros. Para obtermos a semelhança calculamos primeiro a distância entre o vetor  $v_{car}^{novo}$  e todos os restantes considerados. A distância entre  $v_{car}^{novo} = (x^{novo}, y^{novo})$  e um vetor  $v_{car}^i = (x^i, y^i)$ , associado a um peão  $i$ , é dada por

$$d_i = \sqrt{(x_{cor}^{novo} - x_{cor}^i)^2 + (y_{cor}^{novo} - y_{cor}^i)^2}. \quad (4.1)$$

Com base na distância  $d_i$  obtida, vamos calcular a probabilidade  $P_{associar}$  de o falso percurso ser atribuído ao peão  $i$  através de

$$P_{associar}^i = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{d_i^2}{2\sigma^2}}. \quad (4.2)$$

O valor  $P_{associar}^i$  é calculado através de uma função Gaussiana, em que  $\sigma^2$  é uma variável escolhida pelo utilizador. Consideramos que se  $P_{associar}^i > 0.1$  existe a possibilidade de atribuímos a identidade  $i$  ao falso positivo. Guardamos todos os valores que satisfazem esta condição e vamos normalizá-los, de modo a que a sua soma seja um, obtendo-se o conjunto  $\{P^1, P^2, \dots, P^K\}$

Geramos um número aleatório  $x$  com distribuição uniforme e valores entre 0 e 1. Se o valor  $x$  estiver compreendido entre  $P^1 + \dots + P^{k-1}$  e  $P^1 + \dots + P^{k-1} + P^k$ , a identidade escolhida para o falso percurso será  $k$ . Mas, antes desta ser atribuída temos de confirmar se, no instante de tempo  $t_i$  em que o falso

percurso considerado se inicia, a pessoa  $k$  não se encontra presente no sistema. Para isto basta consultar a lista dos percursos visíveis.

Se a pessoa  $k$  estiver a realizar um percurso visível no instante  $t_i$ , vamos atribuir uma nova identidade ao falso percurso gerado, usando um número de identificação que não esteja ainda atribuído.

Caso a pessoa  $k$  não esteja visível no instante  $t_i$  atribuímos essa identidade ao falso percurso gerado. É necessário porém verificar se esse peão não dá entrada enquanto o falso positivo estiver ainda a decorrer. Se tal acontecer, não vamos alterar a identidade do falso positivo, visto que esse já foi atribuído. Em vez disso atribuímos uma nova identidade ao percurso que interfere com o falso positivo.

No caso de não existir nenhuma pessoa que satisfaça  $P_{associar}^i > 0.1$ , atribuímos uma nova identidade ao percurso falso positivo gerado.

Após este processo damos a simulação por terminada. Os percursos invisíveis com erro são obtidos através dos percursos visíveis finais. É apenas necessário estruturar todos os dados do modo como está apresentado na tabela 3.1.

# Capítulo 5

## Experiências

Este capítulo apresenta a caracterização do sistema de câmaras com que vamos trabalhar, as experiências realizadas para testar o método proposto e para avaliar o seu desempenho.

### 5.1 Características do sistema de câmaras

Nesta secção apresentamos as características do sistema de câmaras para o qual vamos realizar os testes. Primeiro descrevemos a base de dados de imagens com que vamos trabalhar; depois apresentamos as super-câmaras existentes; na terceira subsecção definimos as zonas de entrada e saída de todas as câmaras e super-câmaras; de seguida, representamos os grafos reais, adquiridos com os dados do *ground truth*; na quinta subsecção apresentamos os parâmetros de modelação do sistema, necessários para a simulação, que são as matrizes de transição e as distribuições temporais; por último, apresentamos os grafos ideais, com apenas algumas alterações em relação aos grafos reais representados anteriormente.

#### 5.1.1 Base de dados

Vamos trabalhar com um conjunto de imagens de 13 câmaras, que foram adquiridas no âmbito do projeto *High Definition Analytics* e designadas por *HDA dataset* [6] [13] [14]. As sequências de imagens foram adquiridas durante cerca de 30 minutos. Esta base de dados fornece-nos o *ground truth* de todas as deteções de peões. Para cada deteção temos acesso à identidade da pessoa, sob a forma de um número de identificação único; a localização da pessoa na imagem; uma etiqueta de oclusão que indica se a pessoa estava ou não totalmente visível, e uma etiqueta de grupo, que indica se a anotação está a ser feita para uma pessoa ou para um conjunto de pessoas. As anotações são feitas com recurso a uma *bounding box* (ver figura 4.1).

As câmaras com que as imagens foram adquiridas estão localizadas em dois andares do Instituto de Sistemas e Robótica, em Lisboa. É possível observar, na figura 5.1, as plantas dos andares e as localizações das câmaras. Na figura 5.2 apresentamos exemplos de imagens adquirida por quatro das treze câmaras com que vamos trabalhar. Na figura D.1 (ver anexo D) é possível observar exemplos de imagens para as restantes dez câmaras.

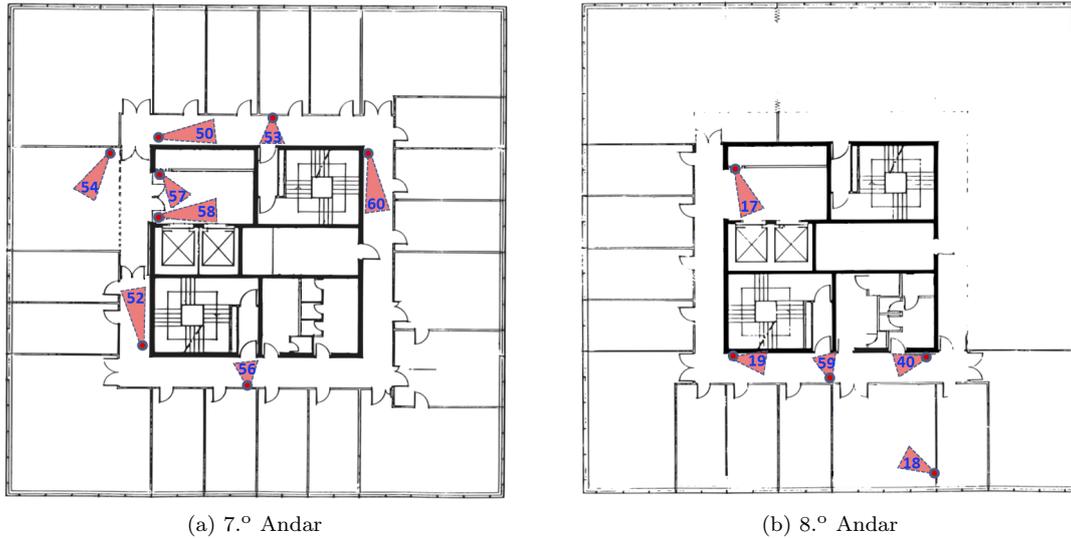


Figura 5.1: Planta do 7.º e 8.º andar do Instituto de Sistemas e Robótica, em Lisboa, com a representação das câmaras utilizadas neste trabalho. A posição das câmaras está representada com um círculo vermelho e é também possível observar uma representação dos seus campos de visão. Cada câmara tem um número de identificação associado. Estas imagens são uma adaptação das plantas disponíveis em [6].

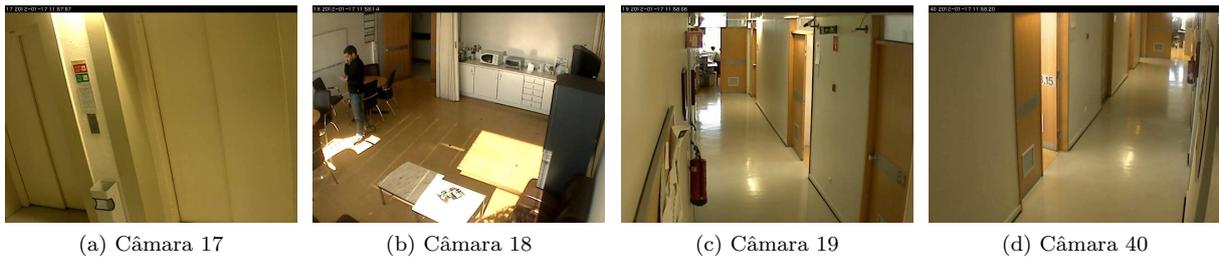


Figura 5.2: Exemplos de imagens adquiridas pelas câmaras 17, 18, 19 e 40 do *HDA dataset*.

A homografia do plano do chão entre o referencial do mundo real e o da imagem foi-nos fornecida para todas as câmaras, à exceção da câmara 52, para a qual foi necessário calcular a homografia.

### 5.1.2 Super-câmaras

O algoritmo proposto nesta tese requer a identificação das câmaras com campos de visão sobrepostos, de modo a definirmos as super-câmaras do sistema, tal como explicado na secção 4.1.2. Para tal, visualizámos os polígonos de visibilidade, que estão representados na figura 5.3. Estes correspondem ao previsto, à parte de pequenos erros. A única exceção é a câmara 56, cujo polígono de visibilidade quase não é visível. Por isso, não iremos realizar mais representações de dados referentes à câmara 56 no referencial no mundo real.

Concluimos, por observação dos polígonos e por consulta das seqüências de imagens, que existem no total oito câmaras com sobreposição dos campos de visão, o que resulta em três super-câmaras, que estão discriminadas na tabela 5.1.

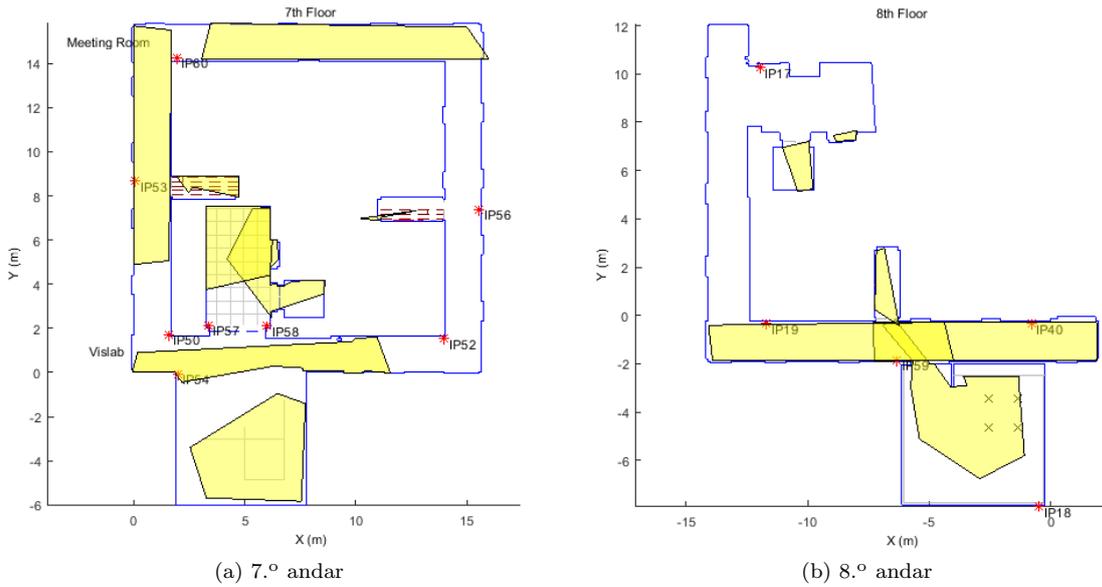


Figura 5.3: Referencial do mundo real com a representação dos polígonos de visibilidade de cada câmara. Estes estão representados a amarelo. As unidades de ambas as coordenadas são metros. É possível também observar uma planta simplificada dos andares e ainda a posição das câmaras, que estão representadas a vermelho e têm associada a inscrição  $IPnn$ , em que  $nn$  é número de identificação respetivo.

Tabela 5.1: Super-câmaras definidas e respetivas câmaras constituintes.

Super-câmara	Câmaras constituintes
S1	50, 53
S2	57, 58
S3	18, 19, 40, 59

### 5.1.3 Zonas de entrada e saída

Vamos passar a caracterizar o sistema de câmaras em relação às suas zonas de entrada e saída, tal como explicado na secção 4.1.3. Para definirmos as zonas de entrada e saída representámos da posição dos peões no referencial da imagem e no do mundo real, nos momentos referentes a aparecimentos e desaparecimentos.

A título de exemplo, apresentamos na figura 5.4 a representação da posição dos peões, no referencial do mundo real, nos momentos em que aparecem e desaparecem das câmaras 40 e 54. Esta representação foi feita de acordo com a explicação dada no apêndice B. Representamos a preto os casos em que foi possível obter a posição dos pés diretamente através da *bounding box* e a vermelho os casos em que os pés não estavam visíveis na imagem e foi necessário utilizar o método de estimação dos pés.

Como já mencionámos, nem todos os pontos representados correspondem a entradas e saídas e temos como exemplo a figura 5.4b, onde é possível observar uma concentração de pontos a preto dentro do polígono de visibilidade. Estes dizem respeito a aparecimentos e desaparecimentos de pessoas no meio de um grupo, que se encontra presente na câmara ao longo de toda a aquisição de dados.

A representação da posição dos peões no referencial do mundo real é útil para nos ajudar a identificar as zonas de entrada e saída mas a definição destas é feita com base na representação no referencial da imagem, usando os pontos centrais das *bounding boxes*. Na figura 5.5 apresentamos os resultados obtidos

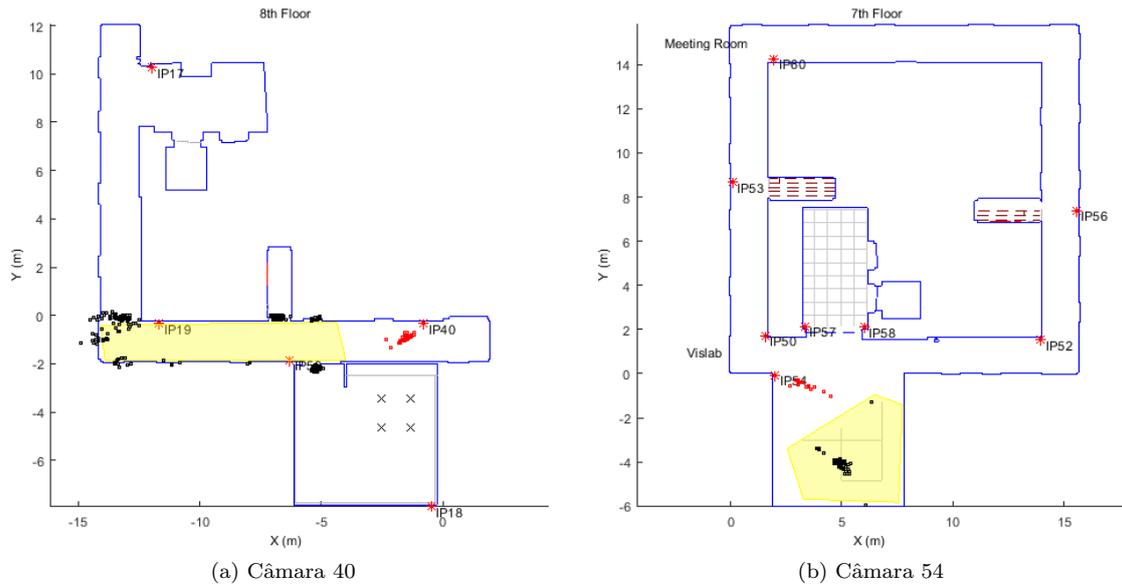


Figura 5.4: Representação da posição dos peões, no referencial do mundo real, nos momentos em que aparecem e desaparecem dos campos de visão as câmaras 40 e 54. É também representado a amarelo o polígono de visibilidade da respetiva câmara. A preto encontram-se representados os casos em que os pés estavam visíveis na imagem e a vermelho estão representados os casos em que foi necessário utilizar o método de estimação dos pés (ver apêndice B).

para as câmaras 17 e 19. Note-se que nestes dois casos apresentados, todos os pontos de aparecimentos e desaparecimentos correspondem a entradas e saídas.

O objetivo é encontrar todos os pontos correspondentes à mesma zona de entrada e saída e definir essa zona através de um retângulo que inclua todos os pontos representados. Na figura 5.5 é possível também observarmos os retângulos obtidos. Em alguns casos isolados não foi possível obtermos os retângulos de modo a que estes incluam todos os pontos pretendidos, sem que haja sobreposição das zonas. Um exemplo dessa situação é apresentado na figura 5.5a. A solução foi utilizar os retângulos que incluam o maior número de pontos, sem sobreposição. Para os pontos que não se localizem no interior da zona correspondente, a associação será feita manualmente.

As zonas de entrada e saída finais das câmaras 18, 19, 40 e 59 encontram-se representadas na figura 5.6. As das restantes câmaras podem ser observadas na figura D.2 (ver anexo D). Cada zona de entrada e saída está associada a um nome identificativo  $nnX$ , em que  $nn$  é um exemplo de número da câmara e  $X$  é o exemplo da letra identificativa da zona.

É necessário realçar que existem casos em que uma zona de entrada e saída pode representar saídas ou entradas para locais distintos. É por exemplo o caso da zona  $40H$ , que representa o gabinete mais próximo dessa zona e também o corredor ao fundo.

Falta ainda definir as zonas de entrada e saída das super-câmaras e para tal vamos utilizar as zonas das suas câmaras constituintes. Existirão zonas limite do campo de visão de uma super-câmara que se encontram apenas observadas por uma câmara, que é por exemplo o caso da zona  $40H$ , da super-câmara 3. Mas outras poderão ser observadas por várias câmaras, por exemplo a porta da casa de banho do 8.º piso, representada pelas zonas  $19B$  e  $40C$ . Este é um caso em que as duas câmaras nos podem fornecer



Figura 5.5: Representação dos pontos centrais das *bounding boxes*, no referencial do mundo real, nos momentos em que aparecem e desaparecem dos campos de visão as câmaras 17 e 19. Nestes casos em particular todas os aparecimentos e desaparecimentos correspondem a entradas e saídas. Os retângulos correspondem a esboços das ZES: na câmara 17 cada cor representa uma ZES distinta enquanto que na câmara 19 cada retângulo corresponderá a uma ZES.

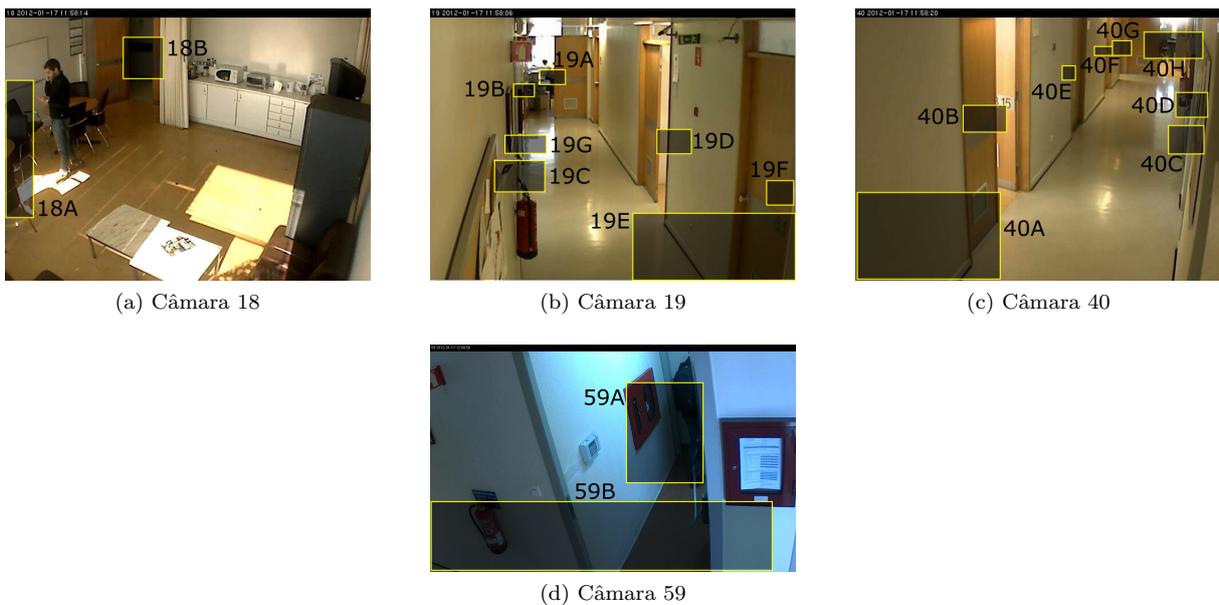


Figura 5.6: Representação das ZES das câmaras 18, 19, 40 e 59. Estas encontram-se delimitadas a amarelo e estão associadas a um nome identificativo de estrutura  $nnX$ , em que  $nn$  é o número identificativo da câmara correspondente e  $X$  a letra identificativa da zona.

informação útil em relação a entradas e saídas e portanto criaremos uma zona de entrada e saída da super-câmara que englobe estas duas zonas e, conseqüentemente, os acontecimentos associados a estas. Quando quisermos utilizar estes dados para construir percursos vamos utilizar sempre, para uma mesma pessoa, a saída associada ao instante de tempo mais tardio ou, no caso de ser uma entrada, a que se verificar primeiro.

As ZES da super-câmara  $S3$  encontram-se representadas na figura 5.7 e as das super-câmaras  $S1$  e  $S2$  na figura D.3, no anexo D. Cada ZES está associada a um nome identificativo de estrutura  $SnX$ , em

que  $n$  é o exemplo do número identificativo da super-câmara e  $X$  é o exemplo da letra identificativa da zona.

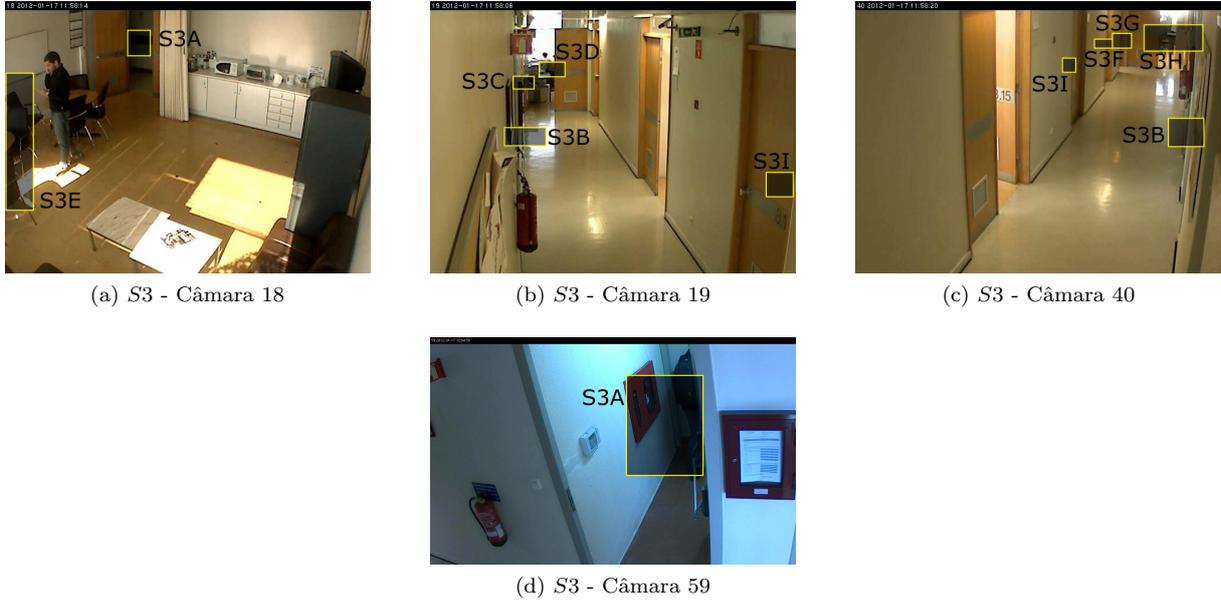


Figura 5.7: Representação das ZES para a super-câmara  $S1$ . Estas encontram-se delimitadas a amarelo e estão associadas a um nome identificativo de estrutura  $S_nX$ , em que  $n$  é o número identificativo da super-câmara correspondente e  $X$  a letra identificativa da zona.

### 5.1.4 Grafos reais

Nesta secção vamos apresentar os grafos reais, obtidos com os dados do *ground truth*. Queremos obter o grafo dos percursos visíveis e o grafo dos percursos invisíveis. Os nós dos grafos são as zonas de entrada e saída, definidas anteriormente. Vamos considerar todas as super-câmaras e todas as restantes câmaras sem sobreposição. Os percursos são obtidos através dos dados etiquetados manualmente, tal como explicado na secção 4.1.4. Verificámos um total de 246 ocorrências de percursos invisíveis e um total de 514 ocorrências de percursos visíveis.

O grafo dos percursos invisíveis está representado na figura 5.8, com um total de 31 ligações, e o grafo dos percursos visíveis na figura 5.9, com um total de 61 ligações. A percentagem de ligações possíveis entre os nós que são, de facto, observadas é de 4.59% para o grafo dos percursos invisíveis e de 5.28% para o grafo dos percursos visíveis.

É importante, para realizarmos a simulação, que os dois grafos tenham todos os nós em comum e é possível perceber, por observação dos mesmos, que existem nós presentes no grafo dos percursos visíveis que não estão presentes no dos invisíveis. É necessário portanto realizar alterações ao grafo, que serão apresentadas nas secções seguintes.

### 5.1.5 Parâmetros de modelação do sistema

Nesta secção vamos apresentar alguns dos parâmetros que caracterizam o *ground truth* e que são necessários para a simulação de dados. Os parâmetros a obter são as probabilidades de transição (ver

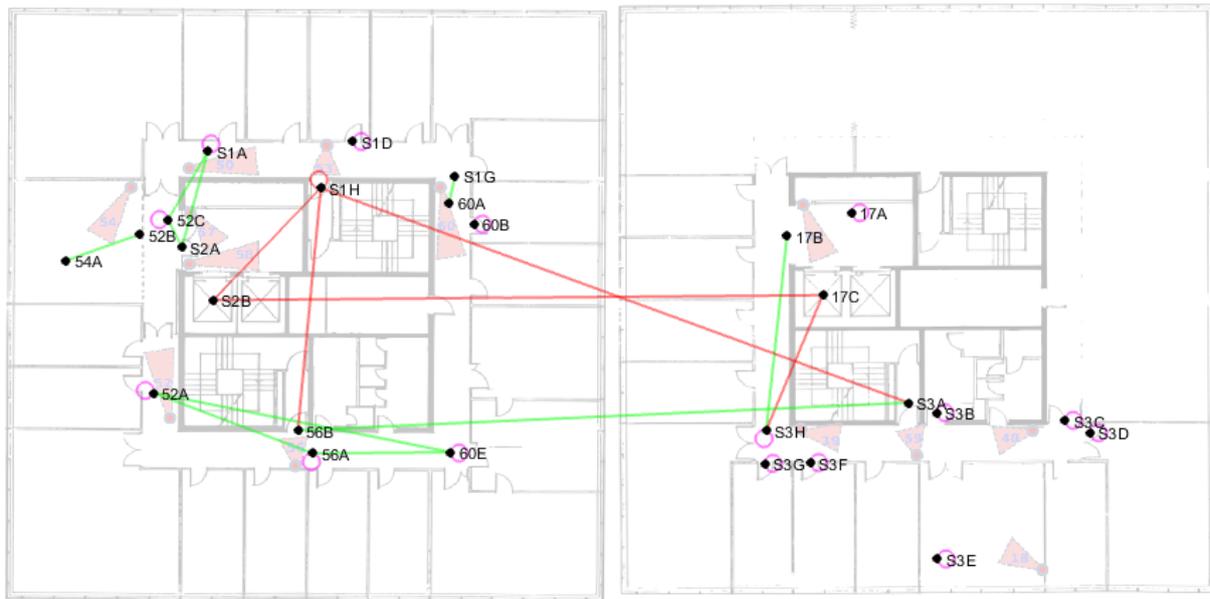


Figura 5.8: Grafo real dos percursos invisíveis. O grafo está representado sobreposto à planta do 7.º (na imagem à esquerda) e do 8.º (na da direita) piso. A posição dos nós na planta pretende dar uma estimativa da sua posição no mundo real, ou seja, da posição das pessoas quando entraram ou saíram pela ZES correspondente. Os percursos cujos dados verificam uma relação linear com a distância estão representados a verde (ver secção 5.1.5 Obter o modelo dos pesos); os percursos circulares a rosa (ver secção 5.1.5 Modelo dos segmentos) e os percursos imprevisíveis a vermelho (ver secção 5.1.5 Modelo dos segmentos).

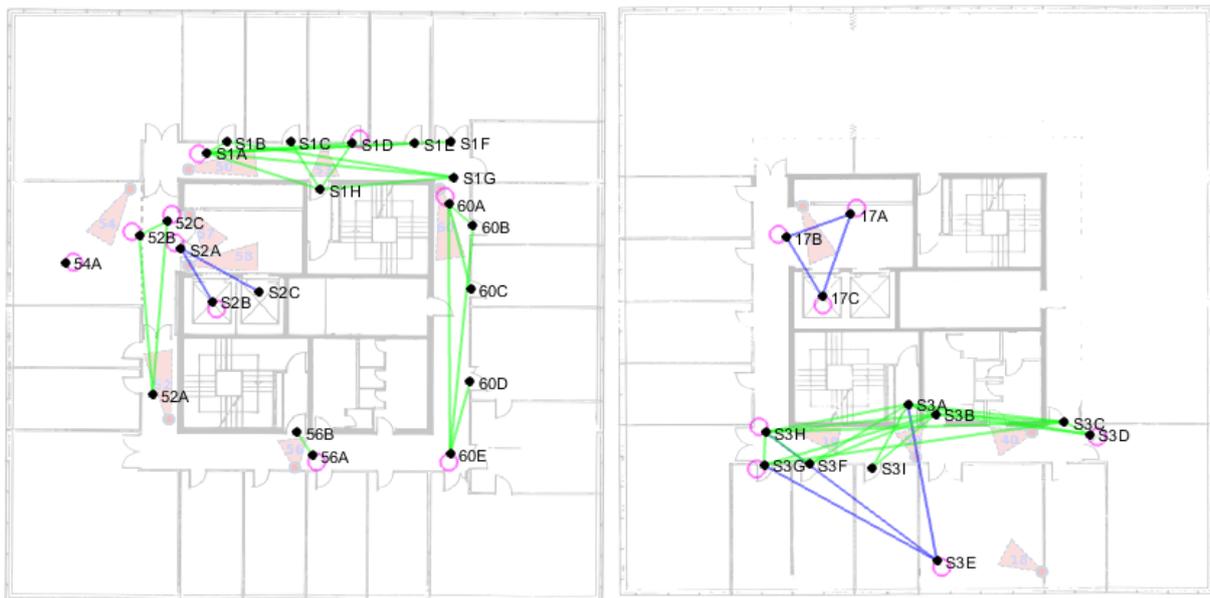


Figura 5.9: Grafo real dos percursos visíveis. O grafo está representado sobreposto à planta do 7.º (na imagem à esquerda) e do 8.º (na da direita) piso. A posição dos nós na planta pretende dar uma estimativa da sua posição no mundo real, ou seja, da posição das pessoas quando entraram ou saíram pela ZES correspondente. Os percursos cujos dados verificam uma relação linear com a distância estão representados a verde (ver secção 5.1.5 Obter o modelo dos pesos); os percursos circulares a rosa (ver secção 5.1.5 Modelo dos segmentos) e os percursos que se localizam em zonas de espera estão representados a azul (ver secção 5.1.5 Modelo dos segmentos).

secção 3.4) e as distribuições temporais (ver secção 3.5). Não vamos apresentar os valores obtidos para as probabilidades, visto que foram obtidas exatamente como descrito. Vamos apenas apresentar os resultados

obtidos para as distribuições.

Para todos os percursos com 8 ou mais ocorrências obtemos as distribuições temporais com recurso ao algoritmo EM. Na figura 5.10 apresentamos dois exemplos de resultados obtidos.

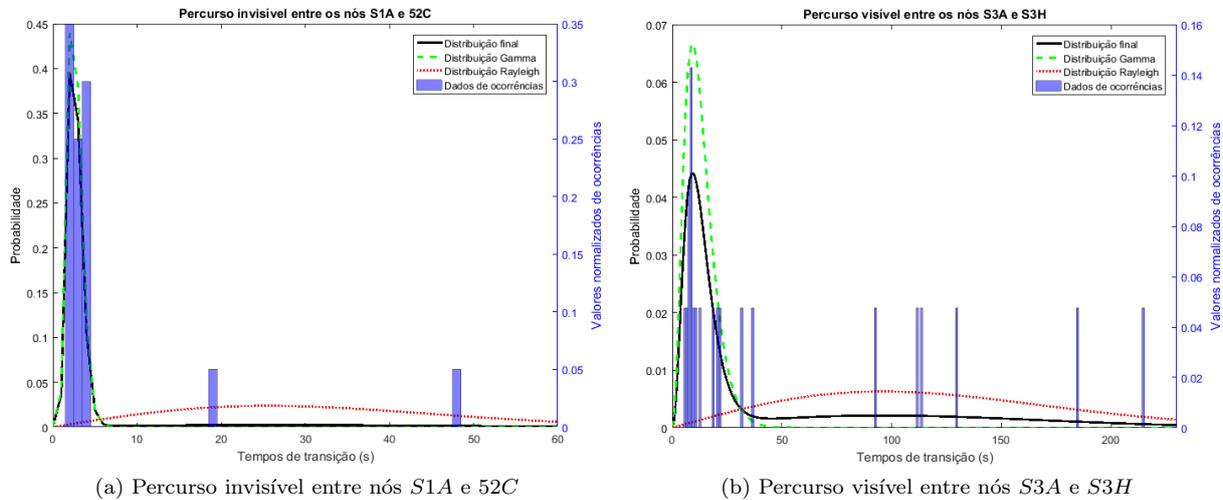


Figura 5.10: Exemplos de distribuições temporais, obtidas com o EM. Os percursos representados têm 20 e 21 dados, respetivamente. A azul representamos o histograma normalizado dos tempos de transição, em segundos. A preto representamos a função de distribuição de probabilidade final, obtida com o EM para os dados de tempos de transição apresentados. Esta resulta de uma mistura de uma distribuição Gamma, representada a verde, e de uma Rayleigh, representada a vermelho.

De todos os percursos obtidos, apenas 12 ligações invisíveis e 21 ligações visíveis verificam 8 ou mais ocorrências. Assim, restam um total de 59 ligações às quais não podemos aplicar o algoritmo EM. São a estes percursos que queremos aplicar os modelos dos tempos de transição apresentados na secção 3.5. Passamos a apresentar os vários modelos e os resultados obtidos.

### Obter o modelo das distâncias

Obtivemos o modelo das distâncias, tal como explicado na subsecção 3.5.1. Para este modelo consideramos todas as ligações cujos tempos de transição se relacionem com a distância do percurso. É o caso das ligações representadas a verde nos grafos das figuras 5.8 e 5.9. Para construir o modelo das distâncias utilizamos os dados dos percursos considerados que tenham verificado 8 ou mais ocorrências, desde que seja possível obter uma boa estimativa da distância associada. Estas ligações e as respetivas distâncias, em metros, estão apresentadas na tabela 5.2.

As distâncias foram obtidas por observação do referencial e da localização das zonas de entrada e saída no mesmo.

Com os dados apresentados na tabela 5.2 e com os valores dos parâmetros da distribuição Gamma, obtidos com o EM para cada um dos respetivos percursos, obtivemos o ajuste linear dos valores da média e da moda dos parâmetros de Gamma em função da distância percorrida. Os pontos considerados e os ajustes obtidos estão apresentados na figura 5.11.

Tabela 5.2: Dados de percursos usados para obter o modelo das distâncias.

Tipo de percurso	Nó 1	Nó 2	Distância (m)
Visível	S1A	S1G	12,7405
	S3A	S3C	11,0849
	S3A	S3H	9,9070
	S3B	S3H	9,6737
	S3C	S3H	15,4999
	60A	60B	2,3440
	60A	60E	13,1740
	60B	60E	12,7100
Invisível	17B	S3H	10,4940
	S1A	52C	2,4525
	S1G	60A	0,8603
	52B	54A	0

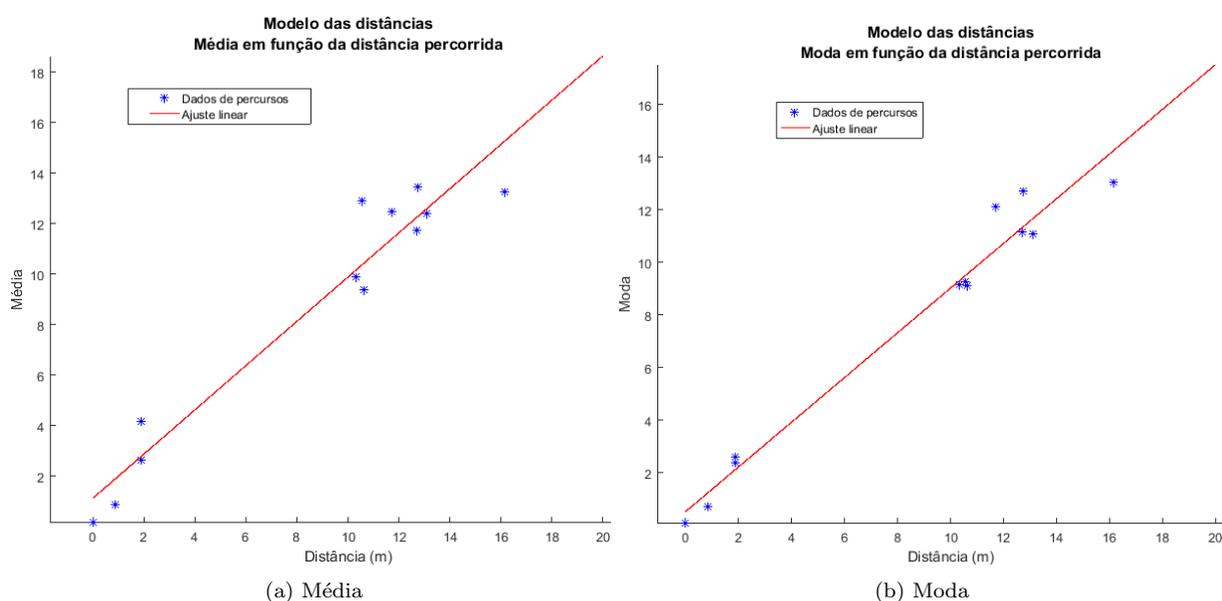


Figura 5.11: Modelo das distâncias: ajustes obtidos para os valores da média e da moda dos parâmetros da distribuição Gamma, em função da distância percorrida. Os pontos a azul são os valores da média (apresentados em (a)) e da moda (em (b)) em função da distância percorrida, em metros, para cada percurso apresentado na tabela 5.2. As retas obtidas pelo ajuste aos pontos estão representadas a vermelho.

### Obter o modelo dos *outliers*

O modelo dos *outliers* foi construído, tal como explicado na subsecção 3.5.2, com os dados de todos os percursos com 8 ou mais ocorrências. O conjunto de dados utilizado está representado no histograma normalizado da figura 5.12 e com estes vamos obter a distribuição Rayleigh, também representada na figura. Esta distribuição é o modelo dos *outliers* que vamos utilizar.

### Obter o modelo dos pesos

Tal como explicado na subsecção 3.5.3, calculamos o modelo dos pesos com os dados de todos os percursos com 8 ou mais ocorrências. Os pesos obtidos foram 0.89 para a distribuição Gamma e 0.11 para a distribuição Rayleigh. Utilizaremos estes valores quando quisermos obter uma mistura com as

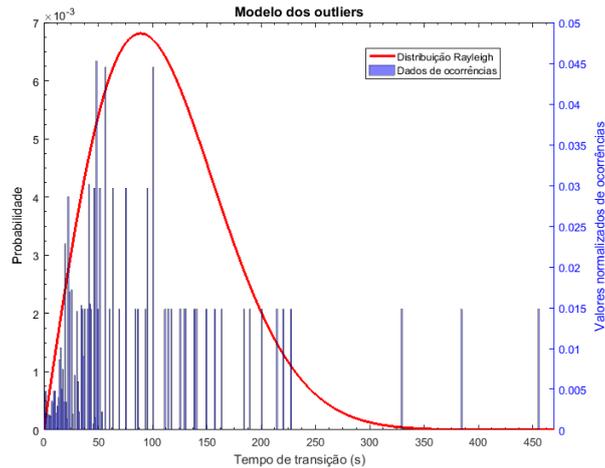


Figura 5.12: Modelo dos outliers: a azul podemos observar um histograma normalizado do número de ocorrências em função do tempo de transição, em segundos. A vermelho representamos a função de distribuição Rayleigh, estimada para os dados apresentados no histograma, e que modela o comportamento dos *outliers*.

distribuições obtidas com o modelo das distâncias e com o modelo dos *outliers*.

### Resultados obtidos com o modelo das distâncias, dos pesos e dos *outliers*

Podemos aplicar o modelo das distâncias a ligações cujos tempos de transição dependam de forma linear da distância associada ao percurso (ligações representadas a verde nos grafos das figuras 5.8 e 5.9) e que verifiquem menos de 8 ocorrências. Os percursos em questão e as respetivas distâncias estimadas estão apresentadas na tabela D.1

Com os modelo das distâncias obtemos os parâmetros da distribuição Gamma de cada um destes percursos. Aplicamos também a cada um destes o modelo dos *outliers*, que nos fornece a distribuição Rayleigh, e o modelo dos pesos. Na figura 5.13 apresentamos um exemplo do resultado obtido.

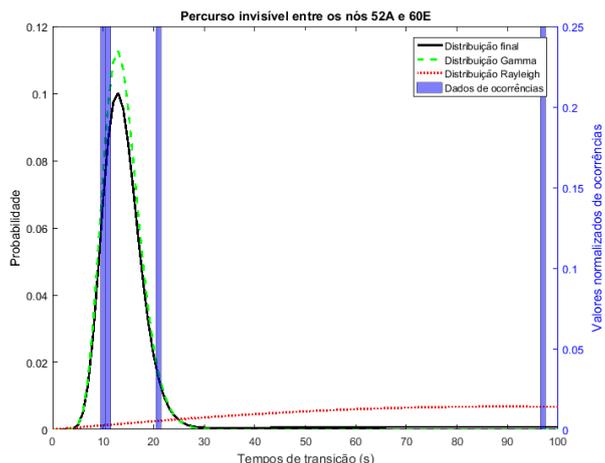
### Modelo dos segmentos

Vamos apresentar alguns dos resultados obtidos com o modelo dos segmentos (ver subsecção 3.5.4).

É possível dividir os percursos em quatro grupos: os que têm uma relação linear com a distância percorrida, os circulares, os imprevisíveis e os que se localizam em zonas de espera. Os primeiros (representados a verde nos grafos das figuras 5.8 e 5.9) são os percursos aos quais podemos aplicar o modelo das distâncias e portanto não serão considerados neste modelo.

A vermelho no grafo 5.8 estão representados os percursos imprevisíveis. Estes são percursos durante os quais é possível os peões saírem do edifício, aumentando muito o espectro de possibilidades de tempos de transição. Estes não verificaram dados suficientes para encontrarmos um padrão dos seus comportamentos. Considerámos os dados de todos os percursos imprevisíveis para construir um segmento. Este verificou menos de 8 ocorrências e assim a distribuição temporal que define cada uma das ligações consideradas é a dada pelo modelo dos *outliers*.

Os percursos circulares são ligações que começam e terminam no mesmo nó, representadas a rosa nos grafos das figuras 5.8 e 5.9. As ligações representadas a azul no grafo 5.9 são percursos que ocorrem em

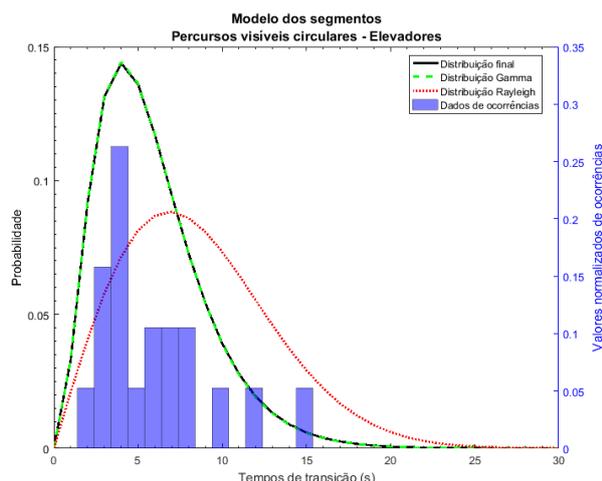


(a) Percurso invisível entre nós 52A e 60E

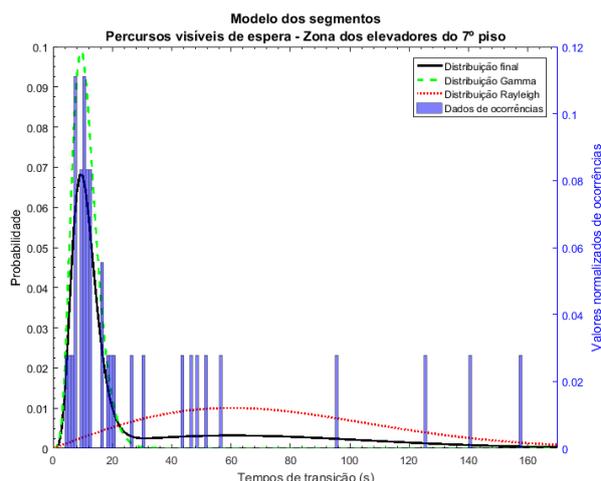
Figura 5.13: Exemplo da distribuição temporal, obtidas com os modelos de distâncias, dos *outliers* e dos pesos. A azul apresentamos um histograma normalizado do número de ocorrências em função do tempo de transição, em segundos. A verde está representada a distribuição Gamma, obtida com o modelo das distâncias. A vermelho representamos a distribuição Rayleigh, obtida com o modelo dos outliers. A preto apresentamos a função de distribuição de probabilidade final, que resulta da mistura das duas outras distribuições apresentadas, com pesos obtidos com o modelo dos pesos.

zonas de espera, como por exemplo, o átrio dos elevadores ou salas de convívio. Com estes dois grupos de ligações vamos criar segmentos, que apresentamos em apêndice na tabela D.2.

Juntamos os dados dos percursos de cada segmento apresentado. Se estes tiverem um total de 8 ou mais ocorrências vamos obter o modelo para esse segmento aplicando o EM ao respetivo conjunto de dados. Dois exemplos de resultados obtidos são apresentados na figura 5.14. Se um segmento tiver menos de 8 ocorrências então a sua função de distribuição será a do modelo dos *outliers*. Tal só se verificou para o segmento da casa de banho.



(a) Visíveis circulares - Elevadores



(b) Visíveis de espera - Zona de espera dos elevadores do 7.º piso

Figura 5.14: Exemplos das distribuições temporais obtidas com o modelo dos segmentos. A azul apresentamos o histograma normalizado do número de ocorrências em função do tempo de transição, em segundos. A preto está representada a função de distribuição final, obtida através do EM, aplicado aos dados apresentados no histograma. A função final é uma mistura das distribuições Gamma, a verde, e Rayleigh, a vermelho.

Todos os percursos constituintes destes segmentos e que tenham verificado menos de 8 ocorrências terão a função de distribuição resultante do modelo do segmento a que pertencem.

### 5.1.6 Grafos ideais

Apresentámos na subsecção 5.1.4 os grafos obtidos com os dados do *ground truth*. Como já mencionado, existem nós presentes no grafo dos percursos visíveis que não estão presentes no dos invisíveis. O número de nós em questão são oito: sete correspondentes a portas de gabinetes ( $S1B$ ,  $S1C$ ,  $S1E$ ,  $S1F$ ,  $S3I$ ,  $60C$ ,  $60D$ ) e um corresponde à porta de um elevadores do 7.º piso ( $S2C$ ). Estas situações verificam-se pois existem poucas ocorrências de entradas e saídas nestes nós.

Como os primeiros sete nós correspondem à porta de gabinetes, podemos criar sete novas ligações circulares para esses nós, em que cada uma terá como distribuição temporal o resultado obtido com o modelo do segmento dos percursos invisíveis circulares, para gabinetes (ver tabela D.2).

A ligação visível que passa no nó  $S2C$  corresponde a uma única ocorrência que se deu para o nó  $S2A$ . Neste caso seria mais complicado modelar os percursos invisíveis possíveis que passam no nó  $S2C$ , visto que este corresponde ao elevador e poderiam existir várias possíveis ligações invisíveis. Assim, vamos ignorar o percurso em questão.

Após estas alterações obtemos os grafos ideais. Na figura 5.15 apresentamos o grafo ideal dos percursos invisíveis, já com as sete novas ligações. Na figura 5.16 apresentamos o grafo ideal dos percursos visíveis, já sem a representação do nó  $S2C$  e da respetiva ligação. Ficamos com 38 ligações invisíveis, 60 ligações visíveis e 33 nós.

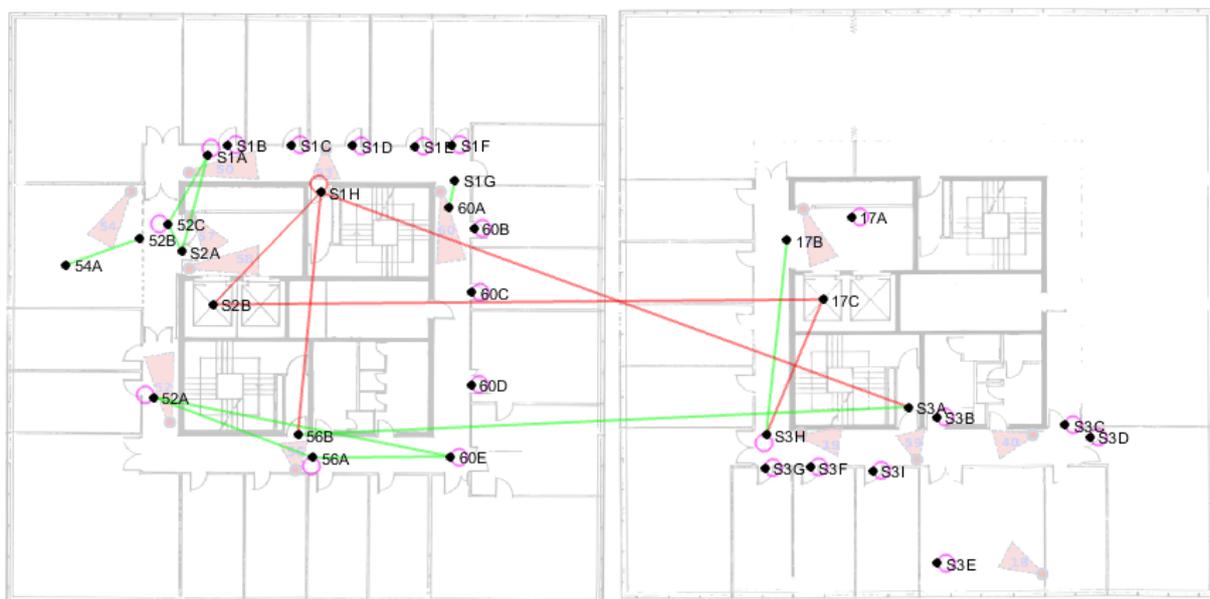


Figura 5.15: Grafo ideal dos percursos invisíveis. O grafo está representado sobreposto à planta do 7.º (na imagem à esquerda) e do 8.º (na da direita) piso. A posição dos nós na planta pretende dar uma estimativa da sua posição no mundo real, ou seja, da posição das pessoas quando entraram ou saíram pela ZES correspondente. Os percursos cujos dados verificam uma relação linear com a distância estão representados a verde; os percursos circulares a rosa e os percursos imprevisíveis a vermelho (ver secção 5.1.5).

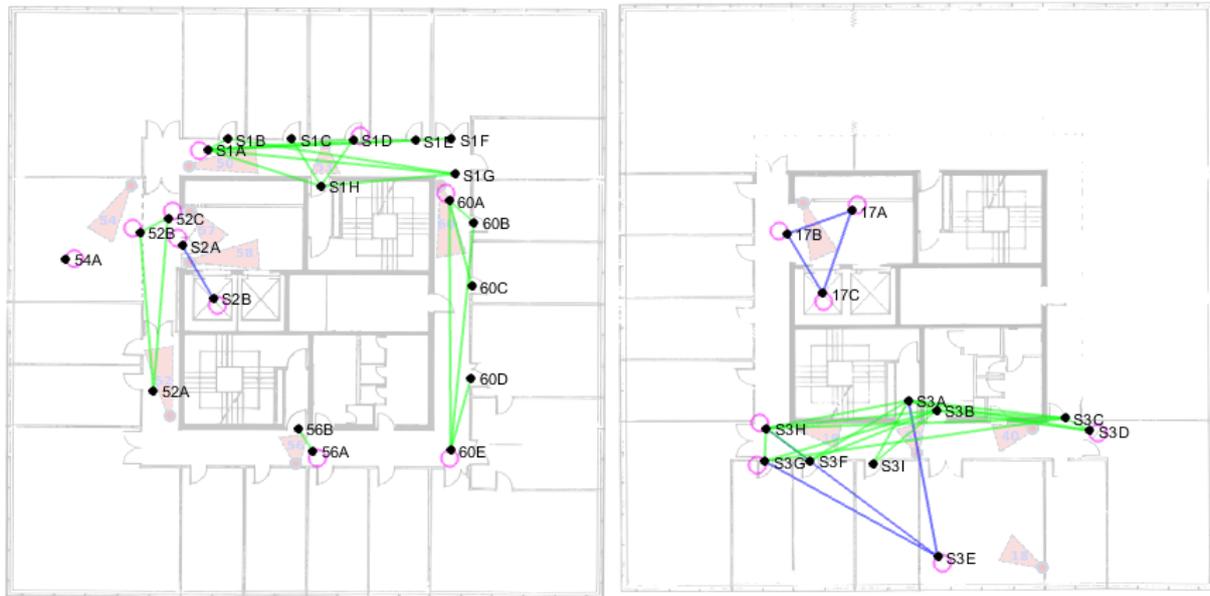


Figura 5.16: Grafo ideal dos percursos visíveis. O grafo está representado sobreposto à planta do 7.º (na imagem à esquerda) e do 8.º (na da direita) piso. A posição dos nós na planta pretende dar uma estimativa da sua posição no mundo real, ou seja, da posição das pessoas quando entraram ou saíram pela ZES correspondente. Os percursos cujos dados verificam uma relação linear com a distância estão representados a verde; os percursos circulares a rosa e os percursos que se localizam em zonas de espera estão representados a azul (ver secção 5.1.5).

A percentagem de ligações possíveis entre os nós que são, de facto, observadas é de 3.49% para o grafo ideal dos percursos invisíveis e de 5.51% para o grafo ideal dos percursos visíveis.

## 5.2 Validação

Nesta secção explicamos como pretendemos avaliar o método proposto para estimação da topologia de uma rede de câmaras.

Vamos utilizar os dados do simulador, gerados tal como explicado na secção 4.2. Com os valores dos vetores de características associados a cada percurso vamos utilizar o método de estimação de topologia proposto (ver secção 3.3) para obter o valor do peso  $w$  atribuído a cada ligação verificada nos dados.

Quanto maior o peso atribuído, maior importância tem a ligação e consideramos que é mais provável que esta corresponda a uma ligação existente no grafo ideal, ou seja, uma ligação correta. De todas as ligações queremos escolher algumas, com base nos pesos  $w$  atribuídos. A seleção das ligações é feita através da escolha de um limiar  $L$  e todos os percursos que verificarem  $w > L$  serão escolhidos.

Vamos avaliar a escolha do limiar  $L$  e o algoritmo de estimação de topologia através da comparação das ligações escolhidas com as ligações dos grafos ideais. Ao conjunto de percursos escolhido chamamos de *acontecimentos selecionados* e ao conjunto de ligações presentes nos grafos ideais chamamos de *acontecimentos relevantes*.

O conjunto *acontecimentos selecionados* poderá conter ligações corretas, ou seja, presentes nos grafos ideais. A essas chamamos de *positivos verdadeiros*. Poderá também conter ligações que não estão presentes nos grafos ideais, às quais chamamos de *falsos positivos*. Poderão ainda existir ligações dos

grafos ideais que não foram selecionadas, às quais chamamos de *falsos negativos*. Com estes dados conseguimos calcular os valores de *precision* e de *recall*, que são dados pelas fórmulas:

$$\begin{aligned} precision &= \frac{\text{positivos verdadeiros}}{\text{acontecimentos selecionados}}, \\ recall &= \frac{\text{positivos verdadeiros}}{\text{acontecimentos relevantes}}. \end{aligned} \tag{5.1}$$

Com estes valores conseguimos calcular ainda a *F-measure*, que é dada por:

$$F\text{-measure} = \frac{2 \cdot precision \cdot recall}{precision + recall}. \tag{5.2}$$

Para um conjunto de pesos obtidos podemos variar o limiar  $L$  e obter diversos valores de *precision*, *recall* e *F-measure*. Podemos visualizar os resultados através da curva  $PR$ , em que visualizamos os valores de *recall* em função da *precision*, e também da curva  $F$ , que é dada pelos valores obtidos para a *F-measure* em função do limiar  $L$  escolhido.

Analisando estas curvas podemos encontrar os valores de limiar que nos permitem obter melhores resultados, isto é, um valor máximo para a *F-measure*.

Vamos analisar as curvas obtidas para as diversas simulações que vamos realizar, escolher os valores de limite com maior *F-measure* e obter o grafo com as ligações escolhidas.

### 5.3 Experiências

Nesta secção vamos apresentar as experiências que vamos realizar, nomeadamente os valores escolhidos para os parâmetros da simulação. Vamos estudar o impacto da variação dos erros de falhas de deteção e de falsos positivos, mantendo os outros valores fixos.

Realizámos todas as simulações (ver secção 4.2) para valores de tempo máximo de entrada  $t_{max} = 172800s$ , que equivale a 2 dias. A probabilidade de entrar uma nova pessoa por segundo tomou o valor  $p_{nova\ entrada} = 0.05$ . O valor de ruído nos vetores de características foi de  $r_{car} = 0.1$ . O valor de  $\sigma^2$ , utilizado nos falsos positivos, tomou um valor fixo de 0.001.

Vamos variar os erros de falhas de deteção ao realizarmos testes para diferentes valores de probabilidade  $P_{FD}$  de um percurso com duração de um *frame* não ser detetado. Os erros de falsos positivos são testados variando-se o valor de probabilidade  $P_{FP}$  de ser detetado um falso positivo no sistema de câmaras, por segundo. Os valores testados são  $PP_{FD} = \{0, 0.3, 0.65, 0.8\}$  e  $P_{FP} = \{0, 0.025, 0.05, 0.08\}$ . Vamos realizar simulações para todas as combinações possíveis destes valores, que são 16 casos.

Para o método de estimação de topologia proposto, atribuímos ao parâmetro livre  $\beta$  o valor 20 para os percursos invisíveis e 60 para os percursos visíveis.

## Capítulo 6

# Resultados

Neste capítulo vamos apresentar e analisar o desempenho do sistema, medido através das curvas PR e F, obtidas usando os dados de várias simulações. Vamos escolher alguns valores limiar  $L$  que maximizem a  $F$ -measure, representar os grafos com as ligações selecionadas e compará-los com os grafos ideais. Os testes foram realizados nas condições apresentadas na secção 5.3.

Primeiro consideramos um caso simulado sem erros. Depois analisamos o impacto do ruído nos vetores de características. Para tal, escolheu-se um ruído fixo de 0.1, valor que será mantido nas simulações seguintes. O terceiro passo é estudar o impacto dos erros de falhas de deteção e de falsos positivos. No fim apresentamos uma discussão geral dos resultados obtidos.

### 6.1 Casos sem erro

Nesta secção discutimos os resultados obtidos com os dados de simulações sem erros. Para simulações sem erros todos os dados de ligações estão corretos. Assim, qualquer ligação selecionada, quer seja visível ou invisível estará correta, independentemente do limiar escolhido. O valor de *precision* obtido será sempre igual a um. O valor de *recall* será um para limiares  $L$  suficientemente baixos. À medida que o valor  $L$  aumenta, selecionamos cada vez menos ligações do conjunto de acontecimentos relevantes e o valor de *recall* diminui. Consequentemente, a curva F tem um máximo para limiares baixos. Se escolhermos, por exemplo,  $L = 3$  para os percursos visíveis e invisíveis os grafos resultantes serão idênticos aos grafos ideais.

### 6.2 Casos com ruído nos vetores de características

Nesta secção discutimos os resultados obtidos para o caso com ruído  $r_{car} = 0.1$  nos vetores de características. Realizámos dez simulações e vamos comparar as curvas PR e F obtidas nestas condições com os casos apresentados nas secções seguintes.

O ruído nos vetores de características não vai levar à criação de novas ligações nem apagar ocorrências existentes e os resultados obtidos são idênticos aos do caso sem erro. A única alteração verificada é uma diminuição generalizada dos valores dos pesos associados às ligações invisíveis. Tal é verificado porque,

com a existência do ruído, os vetores de características associados à entrada e à saída de um percurso invisível não são necessariamente iguais. Nos percursos visíveis o vetor de características associado à entrada é igual ao que está associado à saída, pelo que não se verifica uma alteração nos pesos destas ligações.

Em semelhança ao caso anterior, a *F-measure* continua a ter valor máximo para limiares  $L$  baixos. Se escolhermos, por exemplo,  $L = 1$  para os percursos invisíveis e  $L = 2$  para os percursos visíveis os grafos resultantes serão idênticos aos grafos ideais.

### 6.3 Casos com variações de erros

Nesta secção vamos apresentar os resultados obtidos para os dados simulados na presença de erro de falsos positivos e de falhas de deteção. Para todas as simulações considerámos um ruído fixo nos vetores de características de  $r_{car} = 0.1$ .

Na primeira subsecção vamos apenas apresentar os resultados obtidos quando existem erros de falhas de deteção, na segunda para casos em que existem apenas erros de falsos positivos e na terceira para casos em que estão presentes estes dois tipos de erro.

Para cada combinação de erro testada realizaram-se 10 simulações. Os resultados das curvas PR e F que vamos apresentar neste capítulo representam a média de todas as curvas geradas nas mesmas condições. As curvas obtidas para cada simulação não apresentam um desvio significativo face à média. Para ilustrar a variância verificada, apresentamos os resultados para dez simulações e para a média respectiva nas figuras 6.1 e 6.2, para os percursos invisíveis e visíveis, respetivamente.

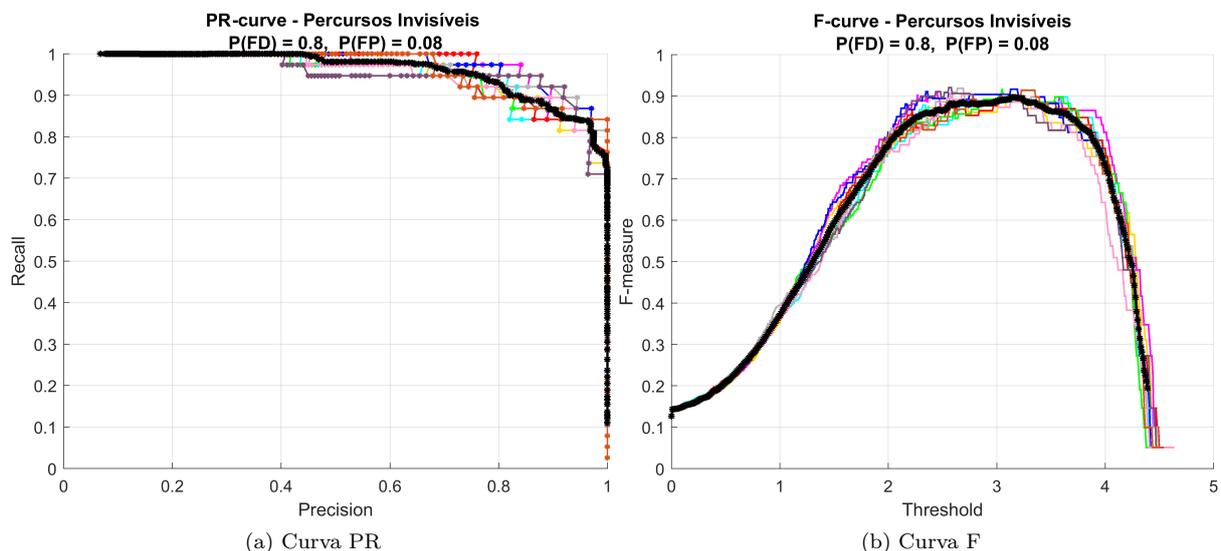


Figura 6.1: Curvas PR e F obtidas para as ligações invisíveis, com probabilidades de erro  $P_{FD} = 0.8$  e  $P_{FP} = 0.08$ . As curvas referentes às 10 simulações realizadas encontram-se representadas em diferentes cores e as curvas médias encontram-se representada a preto.

Nas subsecções seguintes apresentaremos ainda alguns exemplos de grafos obtidos para certos limiares  $L$ . Visto que as curvas PR e F para várias simulações nas mesmas condições de erro não são exatamente iguais, os grafos obtidos poderão ser também distintos. Assim, em vez de apresentarmos o grafo obtido

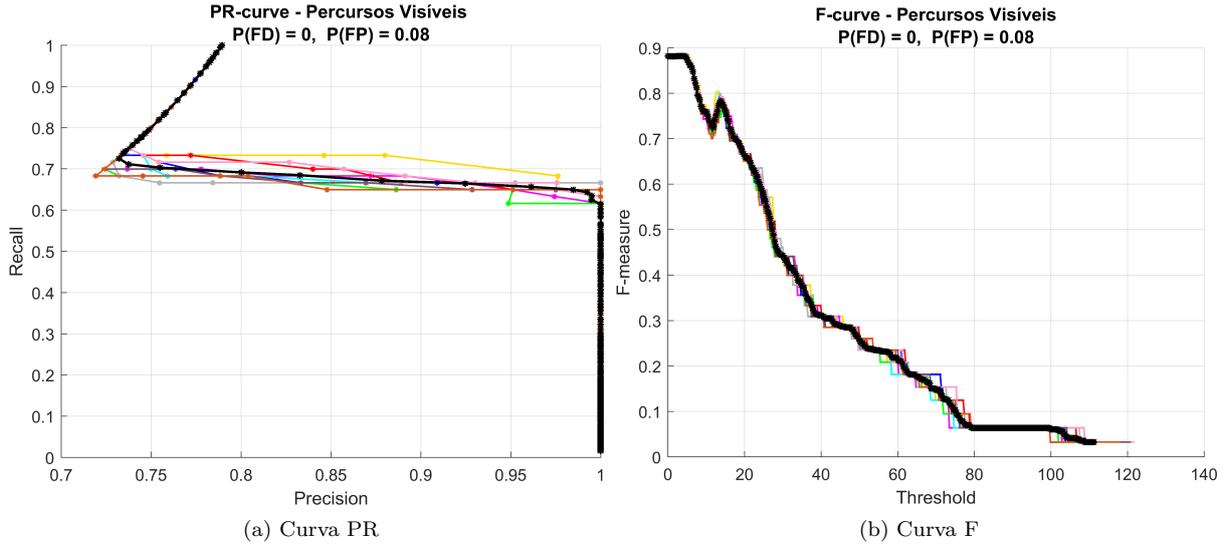


Figura 6.2: Curvas PR e F obtidas para para as ligações visíveis, com probabilidades de erro  $P_{FD} = 0$  e  $P_{FP} = 0.08$ . As curvas referentes às 10 simulações realizadas encontram-se representadas em diferentes cores. As curvas referentes às 10 simulações realizadas encontram-se representadas em diferentes cores e as curvas médias encontram-se representada a preto.

Tabela 6.1: Valores testados para a probabilidade de falhas de detecção e respetivos valores médios de número de percursos não detetados: valores totais e de percentagem em relação ao número de percursos gerados sem erro.

$P_{FD}$	Média de percursos não detetados	
	Número de percursos	Percentagem face ao número inicial de percursos
0.3	648.75	1.49 %
0.65	2508.5	5.77 %
0.8	3788.7	8.71 %

para uma simulação em específico, vamos apresentar a mediana dos grafos obtidos nas mesmas condições de erro, ao qual chamamos de grafo resultante. Ou seja, se uma ligação aparecer representada em 5 ou mais grafos, irá também ser representada no grafo resultante. Caso contrário não será representada.

### 6.3.1 Casos com erros de falhas de detecção

Tal como mencionado na secção 5.3, testámos os erros de falhas de detecção ao variarmos o valor de probabilidade  $P_{FD}$  de um percurso com duração de um *frame* não ser detetado. Os valores testados foram  $P_{FD} = \{0, 0.3, 0.65, 0.8\}$ .

Considerámos útil quantificar os valores testados em termos de impacto nas ligações. Assim, obtivemos o número médio de percursos não detetados para cada valor  $P_{FD}$  testado. Estes valores estão apresentados na tabela 6.1. Para cada um calculámos a percentagem de percursos não detetados face ao número de percursos gerados sem erro.

Todas as curvas PR e F médias obtidas para as ligações invisíveis estão apresentadas na figura 6.3 e para as ligações visíveis estão apresentadas na figura 6.4.

É possível observar na figura 6.4 que as curvas PR e F médias dos percursos visíveis, para os diferentes

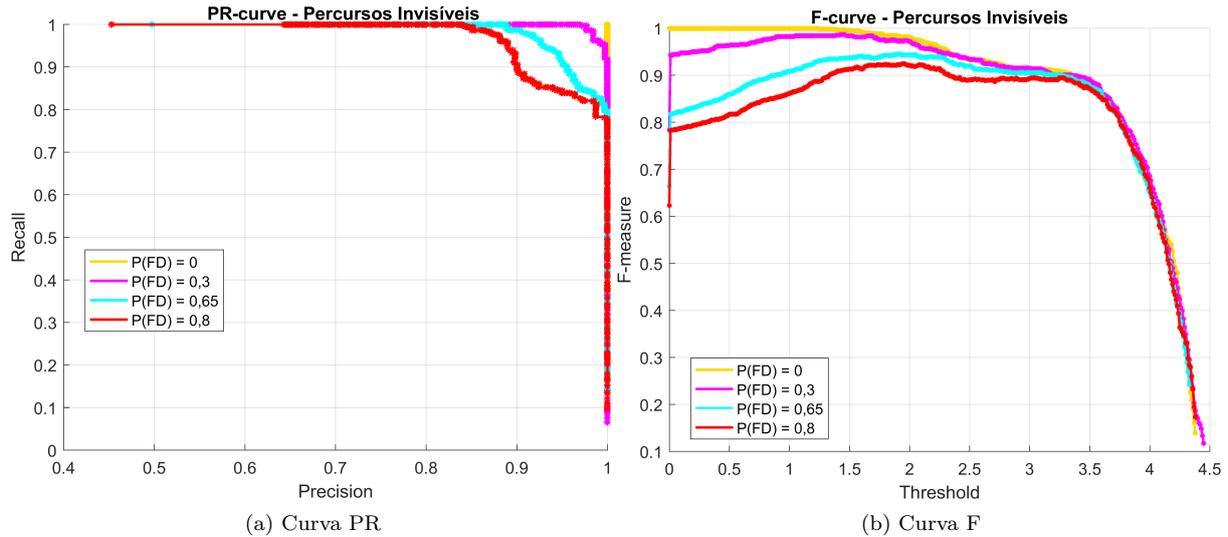


Figura 6.3: Curvas PR e F médias para as ligações invisíveis, obtidas para várias condições de erros de falhas de detecção. A amarelo representamos o caso em que não existem falhas de detecção,  $P_{FD} = 0$ ; a rosa o caso  $P_{FD} = 0.3$ ; a azul o caso  $P_{FD} = 0.65$  e a vermelho  $P_{FD} = 0.8$ . Para todos os casos o valor de ruído nos vetores de características é de 0.1 e a probabilidade de falsos positivos  $P_{FP} = 0$ .

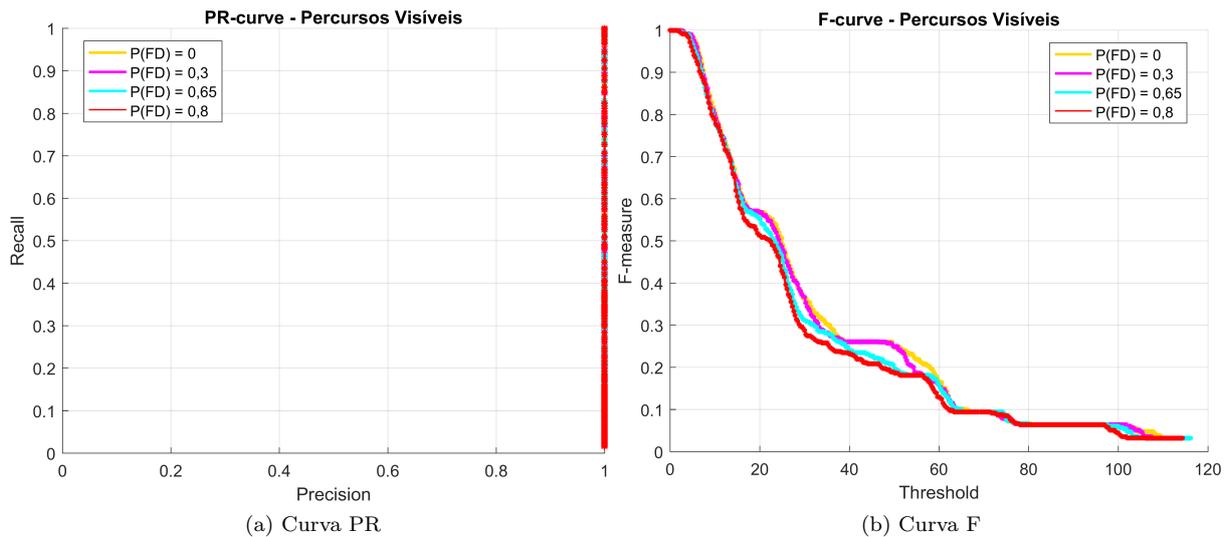


Figura 6.4: Curvas PR e F médias para as ligações visíveis, obtidas para várias condições de erros de falhas de detecção. A amarelo representamos o caso em que não existem falhas de detecção,  $P_{FD} = 0$ ; a rosa o caso  $P_{FD} = 0.3$ ; a azul o caso  $P_{FD} = 0.65$  e a vermelho  $P_{FD} = 0.8$ . Para todos os casos o valor de ruído nos vetores de características é de 0.1 e a probabilidade de falsos positivos  $P_{FP} = 0$ .

valores de erro testados, são muito semelhantes entre si. Tal deve-se ao facto de os erros de falhas de detecção não gerarem novas ligações. Assim, para valores suficientemente baixos de limiar  $L$ , todas as ligações serão escolhidas e farão parte dos acontecimentos relevantes, o que se traduz num valor de  $F$ -measure igual a 1. Se escolhermos, por exemplo,  $L = 1.5$  vamos obter um grafo dos percursos visíveis idêntico ao grafo ideal, para qualquer um dos casos considerados.

O impacto que as falhas de detecção têm nos percursos visíveis é uma diminuição do número de ocorrências de percursos com 3 frames ou menos. Consequentemente, o peso associado a essas ligações também diminui. Tal é visível nas curvas F, principalmente para limiares  $L > 15$ , em que é possível

observar uma translação das curvas F para a esquerda, que será tanto maior quanto o valor  $P_{FD}$  considerado.

Para as ligações invisíveis, os erros de falhas de deteção resultam em novas ocorrências. Isto porque, ao ignorarmos um segmento visível do percurso de um peão, vamos obter um novo percurso invisível para a pessoa considerada. Estes novos percursos terão impacto na curva F, como é possível observar na figura 6.3. Ao contrário do que acontece no caso  $P_{FD} = 0$ , para os restantes casos representados o valor máximo de *F-measure* toma sempre valores inferiores a 1. Ou seja, não vamos conseguir obter um grafo dos percursos invisíveis idêntico ao grafo ideal. Tal acontece pois existirão falsos percursos, ou seja, percursos não verificados no grafo ideal, com um peso associado maior que o peso de algumas ligações verdadeiras.

Vamos apresentar, para dois dos casos considerados, o grafo dos percursos invisíveis obtido. Para cada caso vamos escolher um limiar que maximize o valor de *F-measure*. Na figura 6.5 apresentamos o grafo resultante dos dados obtidos para  $P_{FD} = 0.3$  e com limiar  $L = 1.5$ . Na figura 6.6 apresentamos o grafo resultante para o caso  $P_{FD} = 0.8$  e com limiar  $L = 2$ .

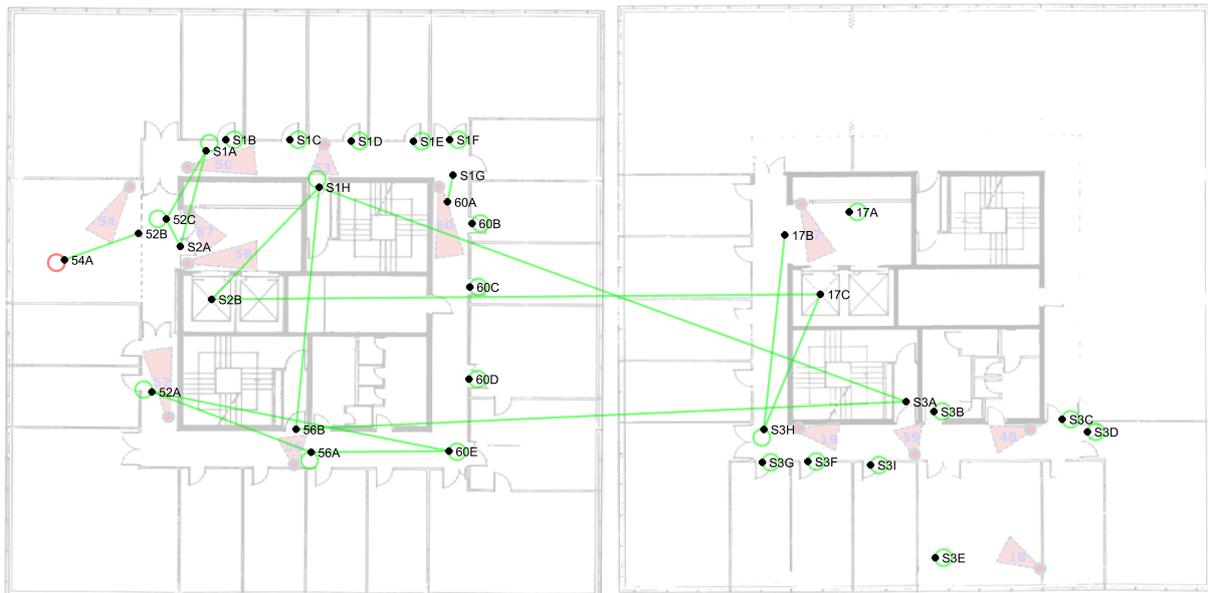


Figura 6.5: Grafo dos percursos invisíveis resultante dos dados das simulações com  $P_{FD} = 0.3$  e limiar  $L = 1.5$ . A verde representamos as ligações correspondentes a positivos verdadeiros e a vermelho as ligações correspondentes a falsos positivos.

### 6.3.2 Casos com erros de falsos positivos

Tal como mencionado na secção 5.3, testámos os erros de falsos positivos ao variarmos o valor de probabilidade  $P_{FP}$  de ser detetado um falso positivo no sistema de câmaras, por segundo. Os valores testados foram  $P_{FP} = \{0, 0.025, 0.05, 0.08\}$ .

Considerámos útil quantificar os valores testados em termos de impacto nas ligações. Assim, para cada valor  $P_{FP}$  considerado obtivemos a média de falsos percursos gerados e a respetiva percentagem, em relação ao número de percursos gerados sem erro. Estes valores estão apresentados na tabela 6.2.

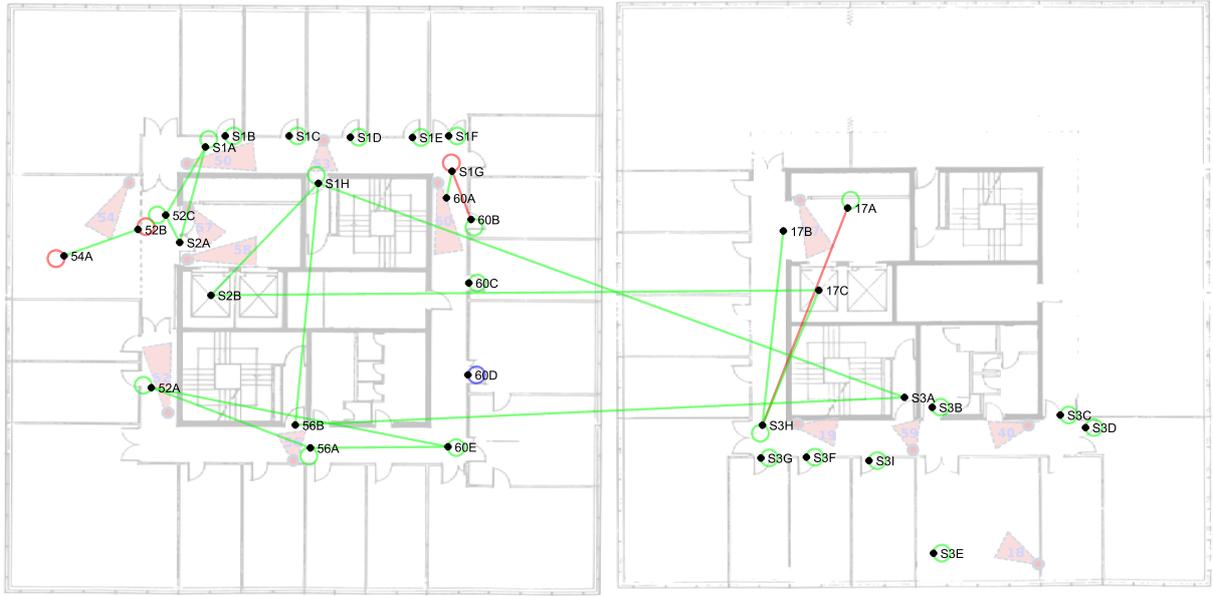


Figura 6.6: Grafo dos percursos invisíveis resultante dos dados das simulações com  $P_{FD} = 0.8$  e limiar  $L = 2$ . A verde representamos as ligações correspondentes a positivos verdadeiros, a vermelho as ligações correspondentes a falsos positivos e a azul as correspondentes a falsos negativos.

Tabela 6.2: Valores testados para a probabilidade de falsos positivos e respetivos valores médios de número de falsos percursos gerados: valores totais e de percentagem em relação ao número de percursos gerados sem erro.

$P_{FP}$	Média de percursos gerados	
	Número de percursos	Percentagem face ao número inicial de percursos
0.025	4211.0	9.64 %
0.05	8221.4	18.88 %
0.08	12728.0	29.31 %

Todas as curvas PR e F médias obtidas para as ligações invisíveis estão apresentadas na figura 6.7 e para as ligações visíveis estão apresentadas na figura 6.8.

Os erros de falsos positivos geram novas ocorrências de percursos visíveis circulares. Em alguns casos estes podem corresponder a ligações verificadas no grafo ideal e noutros não. Nos casos simulados verificámos que cerca de metade das ocorrências de falsos positivos são associadas a novas pessoas e a outra metade a identidades já existentes. Nos casos em que a ocorrência é atribuída a uma identidade existente, poderão ser gerados novos percursos invisíveis associados à pessoa em questão.

É possível observar na figura 6.8 que as curvas PR e F médias dos percursos visíveis para os casos com  $P_{FP} > 0$  apresentam um comportamento distinto em relação ao caso  $P_{FP} = 0$ . Para o caso  $P_{FP} = 0.025$  a curva F apresenta um crescimento inicial, com valor máximo para um valor aproximado de limiar  $L = 5$ . Tal acontece pois as ligações com menores pesos correspondem a ligação não verificadas no grafo ideal. Apresentamos o grafo dos percursos visíveis resultante dos dados das simulações para  $P(FP) = 0.025$  e com  $L = 5$  na figura 6.9.

Para os casos  $P_{FP} = 0.05$  e  $P_{FP} = 0.08$ , as ligações com menor peso correspondem a percursos verificados no grafo ideal. Tal acontece porque, com o aumento do valor de probabilidade de falsos

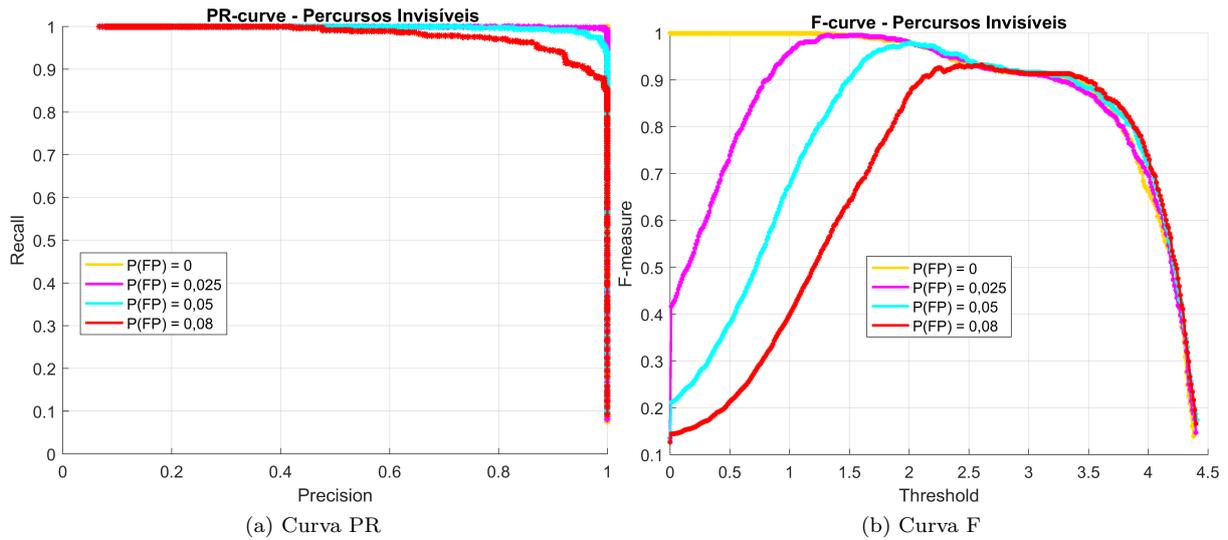


Figura 6.7: Curvas PR e F médias para as ligações invisíveis, obtidas para várias condições de erros de falsos positivos. A amarelo representamos o caso em que não existem falsos positivos,  $P_{FP} = 0$ ; a rosa o caso  $P_{FP} = 0.025$ ; a azul o caso  $P_{FP} = 0.05$  e a vermelho  $P_{FP} = 0.08$ . Para todos os casos o valor de ruído nos vetores de características é de 0.1 e a probabilidade de falhas de detecção  $P_{FD} = 0$ .

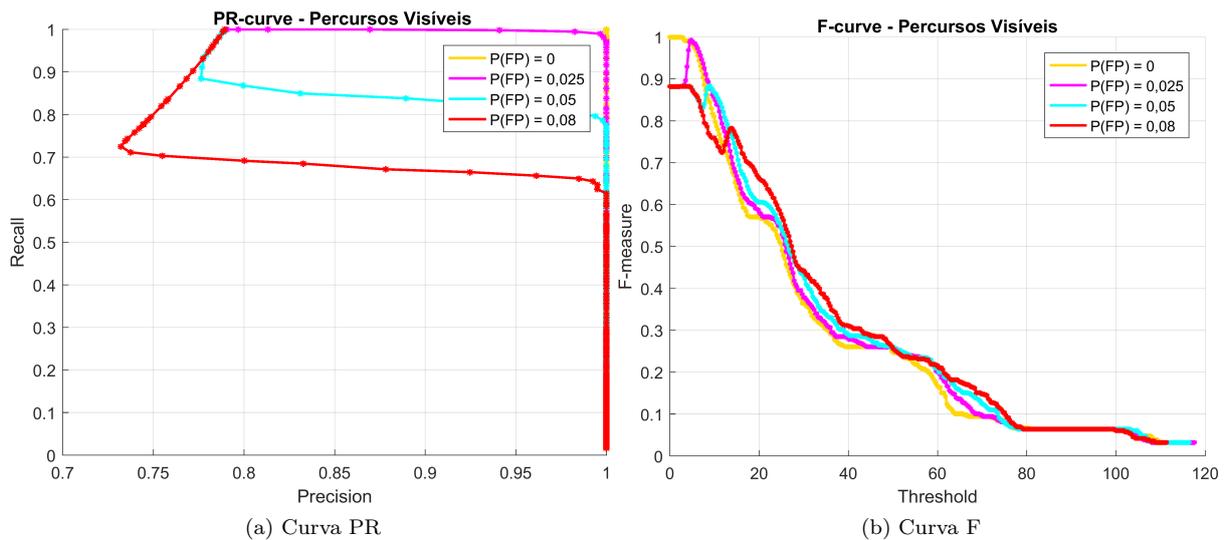


Figura 6.8: Curvas PR e F médias para as ligações visíveis, obtidas para várias condições de erros de falsos positivos. A amarelo representamos o caso em que não existem falsos positivos,  $P_{FP} = 0$ ; a rosa o caso  $P_{FP} = 0.025$ ; a azul o caso  $P_{FP} = 0.05$  e a vermelho  $P_{FP} = 0.08$ . Para todos os casos o valor de ruído nos vetores de características é de 0.1 e a probabilidade de falhas de detecção  $P_{FD} = 0$ .

positivos, são geradas mais ocorrências de percursos incorretos. Assim, para limiares  $L$  baixos os valores de *recall* e de *precision* diminuem, o que leva a um decréscimo da curva F. No caso  $P_{FP} = 0.05$ , o valor máximo da curva F verifica-se para limiares inferiores a 4, aproximadamente, e para um valor próximo de 10. A diferença nos resultados obtidos para estes dois casos é que, no primeiro caso são escolhidas mais ligações, existindo mais falsos positivos, enquanto que, para o segundo caso, é escolhido um número menor de ligações, existindo mais falsos negativos.

Para o caso  $P_{FP} = 0.08$ , o máximo da curva F ocorre para limiares inferiores a 4. O valor de *F-measure* para os casos  $P_{FP} = 0.05$  e  $P_{FP} = 0.08$ , para limiares inferiores a 4 são aproximadamente iguais e os

grafos obtidos são idênticos. A curva F é igual nesta região pois existe um número máximo de ligações que é possível gerar devido a falsos positivos, visto que só podem ser geradas ocorrências de percursos que começam e terminam no mesmo nó.

Apresentamos na figura 6.10 o grafo resultante dos dados da simulações com  $P_{FP} = 0.08$  e limiar  $L = 5$ .

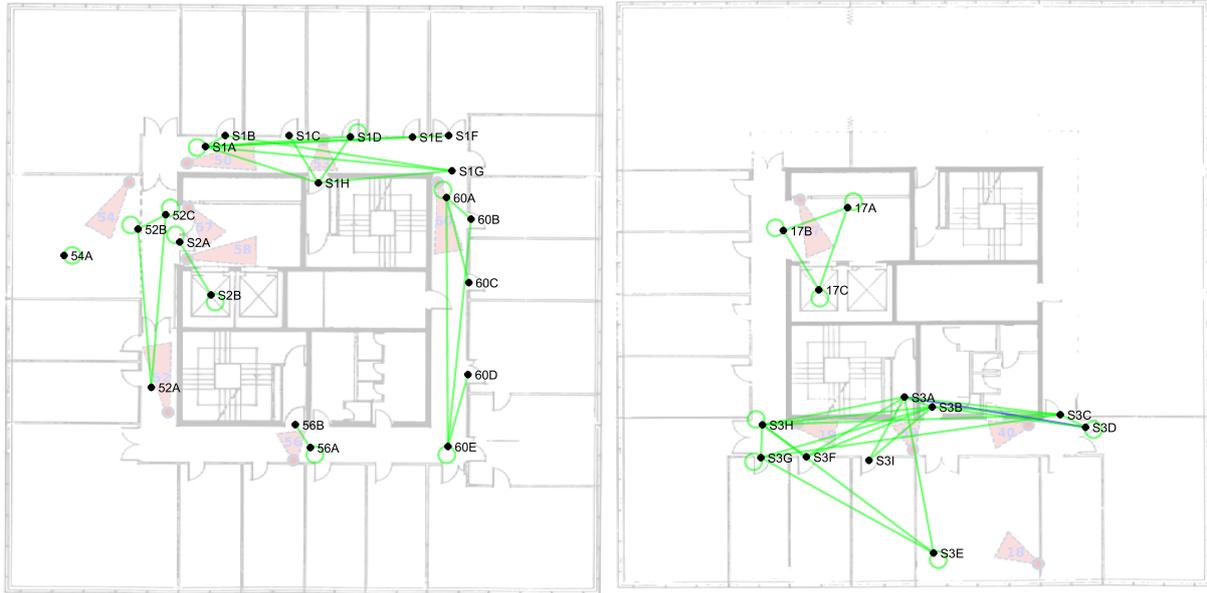


Figura 6.9: Grafo dos percursos visíveis resultante dos dados das simulações com  $P_{FP} = 0.025$  e limiar  $L = 5$ . A verde representamos as ligações correspondentes a positivos verdadeiros e a azul as correspondentes a falsos negativos.



Figura 6.10: Grafo dos percursos visíveis resultante dos dados das simulações com  $P_{FP} = 0.08$  e limiar  $L = 5$ . A verde representamos as ligações correspondentes a positivos verdadeiros e a vermelho as correspondentes a falsos positivos.

Observando na figura 6.7 a curva F para os percursos invisíveis, percebemos que, para o caso



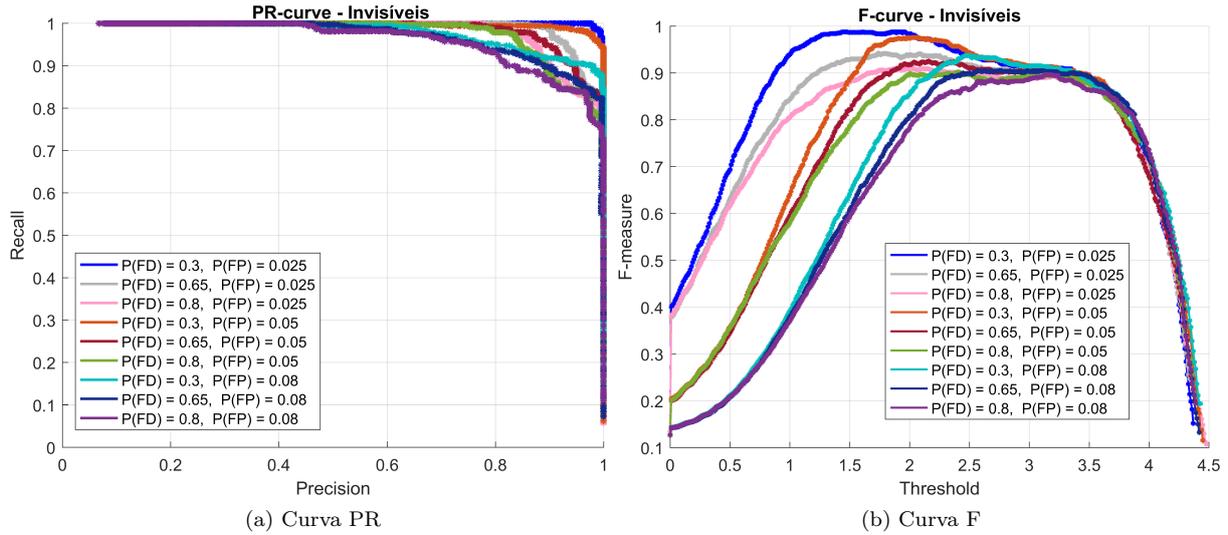


Figura 6.12: Curvas PR e F médias para as ligações invisíveis, obtidas para 9 combinações de erros de falsos positivos e de falhas de detecção. Para todos os casos o ruído nos vetores de características é igual a 0.1 e não existem falhas de detecção.

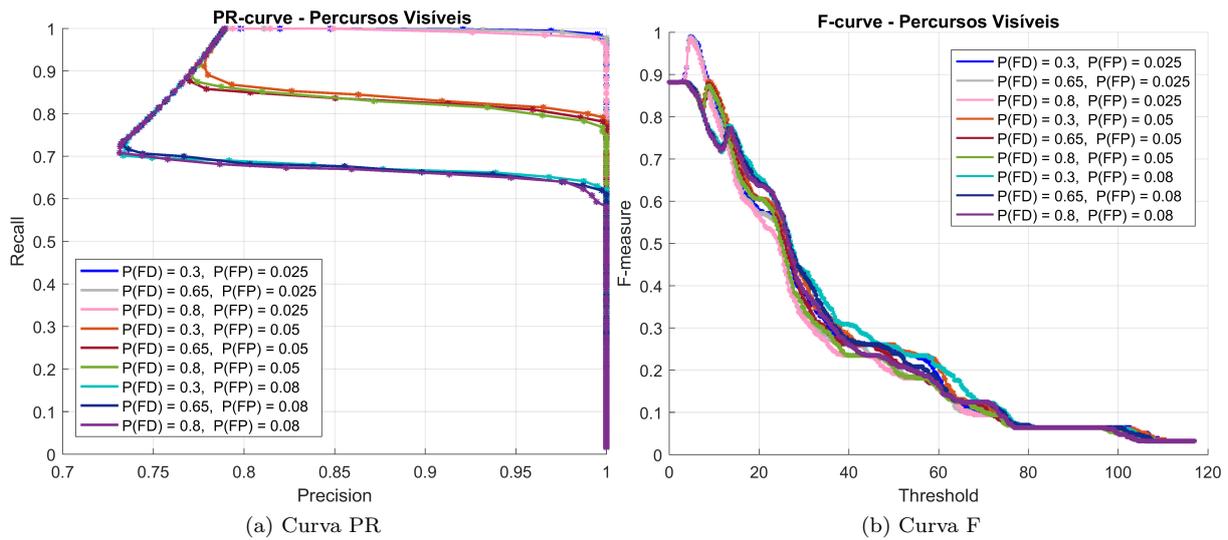


Figura 6.13: Curvas PR e F médias para as ligações visíveis, obtidas para 9 combinações de erros de falsos positivos e de falhas de detecção. Para todos os casos o ruído nos vetores de características é igual a 0.1 e não existem falhas de detecção.

os valores máximos da *F-measure*. Por exemplo, para a curva obtida com os dados das simulações com  $P_{FD} = 0.3$  e  $P_{FP} = 0.025$ , o valor máximo da *F-measure* atinge valores próximos de 1, o que significa que o grafo obtido terá poucos erros. Para o caso simulado com maior erro, ou seja, o caso  $P_{FD} = 0.8$  e  $P_{FP} = 0.08$ , o valor máximo da *F-measure* atinge valores próximos de 0.9.

Os valores do limiar para os quais os máximos da curva F se verificam também dependem dos valores de erros considerados. Para valores menores de erro, o limiar escolhido para obter um máximo da curva F poderá ser  $L = 2$ . Para os casos simulados, seriam todos os casos apresentados na figura 6.12 com  $P_{FP} = 0.025$  ou com  $P_{FP} = 0.05$ . Para as restantes curvas poderíamos escolher um valor entre  $L = 2.5$  e  $L = 3$ .

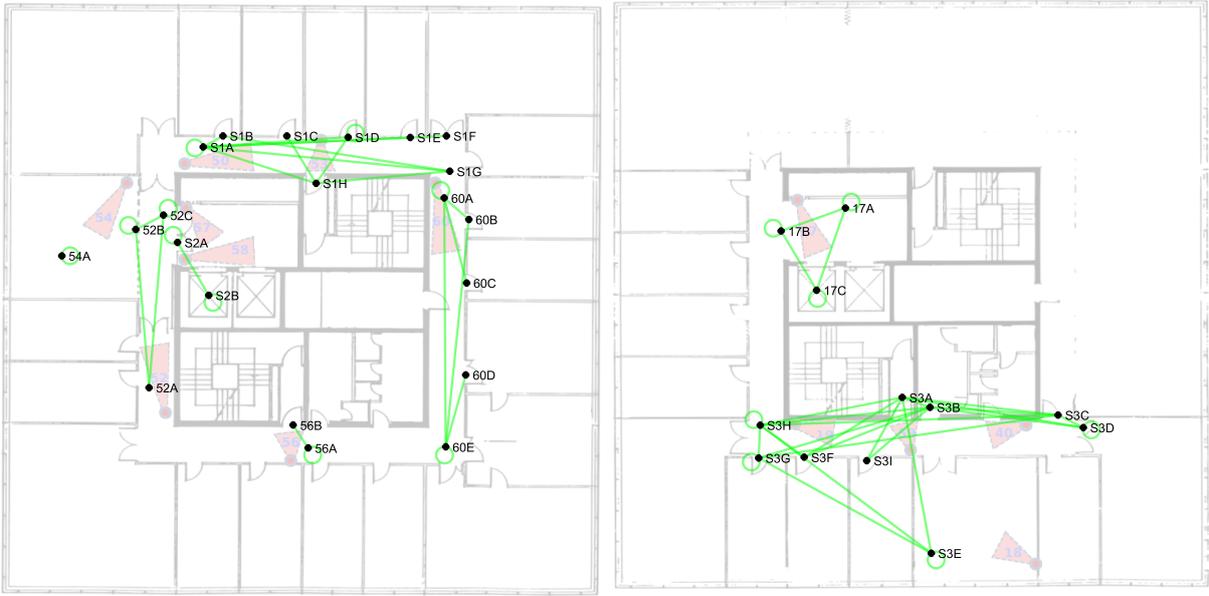


Figura 6.14: Grafo dos percursos visíveis resultante dos dados das simulações com  $P_{FD} = 0.3$ ,  $P_{FP} = 0.025$  e limiar  $L = 4.5$ . A verde estão representadas as ligações correspondentes a positivos verdadeiros.



Figura 6.15: Grafo dos percursos visíveis resultante dos dados das simulações com  $P_{FD} = 0.8$ ,  $P_{FP} = 0.08$  e limiar  $L = 4.5$ . A verde estão representadas as ligações correspondentes a positivos verdadeiros, a vermelho os falsos positivos e a azul os falsos negativos.

Apresentamos na figura 6.16 o grafo resultante dos dados das simulações com valores de probabilidade  $P_{FD} = 0.3$  e  $P_{FP} = 0.025$ , considerando o limiar  $L = 2$ . Na figura 6.17 apresentamos o grafo resultante dos dados das simulações com valores de probabilidade  $P_{FD} = 0.8$  e  $P_{FP} = 0.08$ , considerando o limiar  $L = 3$ .



para todos os casos apresentados. No entanto, não é necessário conhecermos o valor exato do erro para escolhermos o limiar: se tivermos um conhecimento geral do erro do detetor conseguimos escolher um limiar apropriado. De acordo com os testes realizados, para valores de erro mais baixos o limiar que devemos escolher é cerca de 2, enquanto que para erros mais elevados o limiar a escolher será entre 2.5 e 3.

O comportamento das curvas PR e F serão iguais para todos os sistemas, ou seja, à medida que se aumenta o limiar iremos sempre diminuir o número de ligações. Se as ligações que estamos a ignorar corresponderem a ligações erradas, o valor de *F-measure* vai aumentar e a curva F vai crescer. Se por outro lado as ligações ignoradas corresponderem a percursos corretos, o valor de *F-measure* diminui e a curva F vai decrescer.

Apesar de esperarmos um comportamento semelhante das curvas PR e F para qualquer sistema considerado, os limiares para os quais a *F-measure* aumenta ou diminui serão distintos. Mesmo considerando um mesmo sistema, os limiares vão variar consoante o tempo de aquisição de dados considerado. Uma aquisição mais curta corresponderá a menos ocorrências de cada percurso e consequentemente pesos mais baixos de cada ligação. Assim, a curva terá o mesmo comportamento, mas os valores de limiar a considerar serão outros.

Para outros sistemas, mesmo que consideremos o mesmo tempo de aquisição de dados, o número de ocorrências dos vários percursos pode ter um comportamento muito distinto, o que vai gerar outros valores dos pesos das ligações. Isto faz com que não se consiga prever para que limiares a *F-measure* terá valores máximos.

Os resultados obtidos com o método proposto dependem da taxa de erro de falsos positivos e de falhas de deteção do algoritmo de deteção usado. Quanto maior o erro presente, pior serão os resultados obtidos.



## Capítulo 7

# Conclusões e trabalho futuro

Neste trabalho testámos um algoritmo de estimação automática da topologia de uma rede de câmaras. Obtivemos os resultados para dados gerados com um simulador. Realizámos testes na presença de um ruído fixo nos vetores de características e para vários valores de percentagem de erros de falhas de deteção e de falsos positivos.

Avaliámos os resultados obtidos comparando-os com os grafos ideais, que foram construídos usando os dados do *ground truth*. A avaliação foi feita calculando-se, para cada caso testado, as curvas PR e F. Para cada curva F, encontramos o limiar (ou seja, o valor dos pesos a partir do qual devemos escolher as ligações) que nos permite obter o maior valor possível de *F-measure*.

Concluimos, com base nos testes realizados, que para o sistema testado e para o tempo de aquisição de dados considerado, o valor de limiar a escolher para os percursos visíveis será aproximadamente constante para todas as combinações de erros testados.

Para os percursos invisíveis, o limiar a escolher dependerá dos erros presentes no detetor. Se não se souber exatamente o valor das percentagens de erros com que se está a trabalhar, não conseguimos escolher o limiar que maximiza a curva F. Podemos no entanto escolher um limiar aproximado, consoante estejamos a trabalhar com detetores com maior ou menor erro.

O algoritmo mais básico de estimação de topologia é considerar todas as ligações verificadas como sendo verdadeiras, o que seria equivalente a considerar o limiar igual a 0. Portanto, com a nossa solução, mesmo não se conhecendo especificamente a taxa de erro do detetor, ao escolhermos um valor de limiar específico, estaremos sempre a obter melhores resultados que obteríamos com o algoritmo mais básico.

Considerando o mesmo sistema para o qual realizámos os testes, mas um valor distinto para o tempo de aquisição de dados, espera-se que os valores dos pesos se alterem pois o número médio de ocorrências de cada percurso vai alterar-se. Nestes casos, os limiares para os quais a curva F tem valor máximo serão distintos e não conseguimos, com base nos testes realizámos, tirar conclusões acerca destes valores.

Como trabalho futuro poderíamos tentar resolver esta questão variando o valor do parâmetro  $\beta$  do algoritmo de obtenção dos pesos das ligações. Ao variarmos o tempo de aquisição de dados, poderíamos escolher um novo valor de  $\beta$  de modo a que o valor máximo dos pesos obtidos seja aproximadamente igual ao valor máximo dos pesos obtidos para o tempo de aquisição testado. Desta forma, é possível que

os limiares que maximizem as curvas  $F$  tomem valores aproximadamente constantes, para o mesmo valor de probabilidade de erros escolhido e para o mesmo sistema considerado, independentemente do tempo de aquisição selecionado.

Para diferentes sistemas, apesar do comportamento esperado das curvas ser o mesmo, não conseguimos tirar conclusões sobre os valores de limiar a escolher.

Poderíamos ainda, como trabalho futuro, testar a performance do algoritmo para erros de trocas de identidade, algo que não foi feito neste trabalho. Poderíamos realizar testes para dados de um detetor real, visto que neste trabalho só se fizeram testes para dados gerados por um simulador. Ao utilizarmos os dados de um detetor real poderíamos testar o algoritmo para diferentes tipos de vetores de características.

Neste trabalho obtivemos as probabilidades de transição e as distribuições temporais das ligações utilizando os dados do *ground truth*. No futuro, poderia também ser pertinente obter estes valores usando os dados das ligações escolhidas com o algoritmo de obtenção de topologia, visto que esta informação é útil na resolução do problema de re-identificação.

# Bibliografia

- [1] W. Hu, T. Tan, L. Wang, and S. Maybank, “A Survey on Visual Surveillance of Object Motion and Behaviors,” *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 34, no. 3, pp. 334–352, 2004.
- [2] D. Kamenetsky, “Camera Network Topology Discovery Literature Review,” *Intelligence, Surveillance and Reconnaissance Division, Defence Science and Technology Organisation, DSTO-GD-0667*, 2011.
- [3] A. Hampapur, L. Brown, J. Connell, S. Pankanti, A. Senior, and Y. Tian, “Smart Surveillance: applications, technologies and implications,” in *Fourth International Conference on Information, Communications and Signal Processing, 2003 and the Fourth Pacific Rim Conference on Multimedia. Proceedings of the 2003 Joint*, pp. 1133–1138, 2003.
- [4] A. Bedagkar-Gala and S. K. Shah, “A survey of approaches and trends in person re-identification,” *Image and Vision Computing*, vol. 32, no. 4, pp. 270–286, 2014.
- [5] M. Song, D. Tao, and S. J. Maybank, “Sparse camera network for visual surveillance – A comprehensive survey,” *CoRR*, vol. abs/1302.0446, 2013.
- [6] “Hda person dataset.” <http://vislab.isr.ist.utl.pt/hda-dataset/>. Accessed: 2017-09-07.
- [7] G. Jones, J. Renno, and P. Remagnino, “Auto-calibration in multiple-camera surveillance environments,” in *Process of Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, pp. 583–588, 2002.
- [8] Javed, Rasheed, Shafique, and Shah, “Tracking across multiple cameras with disjoint views,” in *Proceedings Ninth IEEE International Conference on Computer Vision*, pp. 952–957 vol.2, IEEE, 2003.
- [9] D. Makris, Tim Ellis, and J. Black, “Bridging the gaps between cameras,” *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 205–210, 2004.
- [10] X. Li, W. Dong, F. Chang, and P. Qu, “Topology Learning of Non-overlapping Multi-camera Network,” *International Journal of Signal Processing, Image Processing and Pattern Recognition*, vol. 8, pp. 243–254, nov 2015.

- [11] B. M. Lake and J. B. Tenenbaum, “Discovering structure by learning sparse graph,” in *Proceedings of the 33rd Annual Cognitive Science Conference*, 2010.
- [12] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*. Pearson Education, Inc., 2nd ed., 2010.
- [13] A. Nambiar, M. Taiana, D. Figueira, J. C. Nascimento, and A. Bernardino, “A Multi-camera video data set for research on High-Definition surveillance,” in *Int. J. of Machine Intelligence and Sensory Signal Processing*, 2014.
- [14] D. Figueira, M. Taiana, A. Nambiar, J. C. Nascimento, and A. Bernardino, “The HDA+ data set for research on fully automated re-identification systems,” in *Workshop, European Conference on Computer Vision (ECCV)*, 2014.
- [15] J. A. Bilmes, “A Gentle Tutorial of the EM Algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models,” 1998.
- [16] B. Ripley, “Pattern Recognition and Neural Networks,” ch. A.2 The EM algorithm, Cambridge University Press, 1996.
- [17] G. Vegas-Sánchez-Ferrero, M. Martin-Fernandez, and J. M. Sanches, “A gamma mixture model for IVUS imaging,” in *Multi-Modality Atherosclerosis Imaging and Diagnosis*, vol. 9781461474, pp. 155–171, New York, NY: Springer Verlag, 2014.
- [18] “Gamma distribution.” [https://en.wikipedia.org/wiki/Gamma\\_distribution](https://en.wikipedia.org/wiki/Gamma_distribution). Accessed: 2017-09-04.
- [19] “Digamma function.” <http://mathworld.wolfram.com/DigammaFunction.html>. Accessed: 2017-09-18.
- [20] “Newton’s method.” <http://mathworld.wolfram.com/NewtonsMethod.html>. Accessed: 2017-09-18.
- [21] “Rayleigh distribution.” [https://en.wikipedia.org/wiki/Rayleigh\\_distribution](https://en.wikipedia.org/wiki/Rayleigh_distribution). Accessed: 2017-09-04.
- [22] R. Szeliski, *Computer Vision: Algorithms and Applications*. Springer-Verlag New York, Inc., 1st ed., 2010.
- [23] R. Hartley and A. Zisserman, “Multiple view geometry in computer vision,” ch. 8.7 Affine 3D measurements and reconstruction, Cambridge University Press, 2nd ed., 2003.

# Apêndice A

## Conhecimentos teóricos

### A.1 O algoritmo Esperança-Maximização

Vamos apresentar o algoritmo de *Esperança-Maximização* (EM) e mostrar a sua aplicação ao caso de estimativa de uma mistura de distribuições Gamma e Rayleigh. Esta secção baseia-se em [15] e [16].

O EM é um algoritmo iterativo que permite calcular a estimativa de máxima verosimilhança dos parâmetros de uma distribuição de variável aleatória, sobre a qual temos um conjunto de observações  $\mathcal{X}$ , em situações em que existe uma outra variável aleatória  $\mathcal{Y}$ , que é conjuntamente distribuída com a primeira mas que não é observada.

Admitindo que temos um conjunto de dados escalares  $\mathcal{X} = \{x^1, \dots, x^N\}$ , de tamanho  $N$ , formado por amostras independentes e identicamente distribuídas de uma distribuição de probabilidade desconhecida que queremos estimar. Sabemos que os dados observados  $\mathcal{X}$  podem pertencer a várias classes  $C_i$  mas não sabemos a que classe correspondem. Sabemos que a distribuição de probabilidade desconhecida é uma mistura das distribuições das várias classes e que o número distribuições é igual à quantidade de classes existentes. Definimos uma variável escalar aleatória  $\mathcal{Y}$  tal que  $\mathcal{Y} = i$  se  $\mathcal{X}$  foi gerado pela classe  $C_i$ . Nesta situação podemos aplicar o EM para obtermos a estimativa de máxima verosimilhança do vetor de parâmetros  $\Theta$ , que é formado pelos parâmetros de todas as distribuições da mistura considerada. Cada classe gera dados com uma densidade  $P(\mathcal{X}|C_i) = p_i(\mathcal{X})$ , sendo o seu *prior*  $P(C_i) = w_i$ .

Considerando que existem  $M$  classes, a densidade de probabilidade de  $\mathcal{X}$  é dada por

$$p(x) = \sum_i^M w_i p_i(x). \quad (\text{A.1})$$

O EM consiste na iteração de uma sequência de dois passos: *E step* e *Q step*. Vamos passar a explicar estes passos para a situação apresentada acima, no caso específico em que existem duas classes: uma que gera dados com uma distribuição de probabilidade Gamma e outra com distribuição Rayleigh. Consideramos que temos os valores obtidos para a iteração  $n$  e que vamos agora proceder à iteração  $n+1$ :

- **E step** Este passo consiste no cálculo de uma função auxiliar  $Q(\Theta, \Theta^n)$ , definida como

$$Q(\Theta, \Theta^n) = E'[\log p(\mathcal{X}, \mathcal{Y}|\Theta)]. \quad (\text{A.2})$$

Em que  $E'(\cdot)$  é o valor esperado em relação a  $\mathcal{Y}$ , calculado usando a distribuição  $p(\mathcal{Y}|\mathcal{X}, \Theta^n)$ . E em que  $\Theta^n$  é a estimativa atual dos parâmetros, ou seja o valor obtido na iteração  $n$  do algoritmo. A função  $Q(\Theta, \Theta^n)$  pode ser obtida através da expressão

$$Q(\Theta, \Theta^n) = \sum_{k=1}^N \sum_i^2 \pi_i^k [\log p_i(x^k) + \log w_i], \quad (\text{A.3})$$

em que  $\pi_i^k$  é normalmente interpretado como a probabilidade da classe  $i$  ter gerado a observação  $x^k$  e pode ser obtido com a expressão

$$\pi_i^k = \frac{w_i^n p_i^n(x^k)}{\sum_j^2 w_j^n p_j^n(x^k)}. \quad (\text{A.4})$$

- **M step** Neste passo calculamos a próxima estimativa dos parâmetros  $\Theta^{n+1}$ , através da maximização da função  $Q(\Theta, \Theta^n)$  e que é dada por

$$\Theta^{n+1} = \arg \max_{\Theta} Q(\Theta, \Theta^n). \quad (\text{A.5})$$

Esta função consiste em maximizar  $Q(\Theta, \Theta^n)$  dado em A.3 em relação a  $\Theta$ . Vamos considerar separadamente as maximizações em relação aos pesos  $w_i$  e em relação aos parâmetros das distribuições consideradas.

- **Atualização dos pesos** Neste passo é maximizada a equação A.3 em relação a  $w_i$ , com a restrição de  $\sum_i w_i = 1$ , obtendo-se

$$w_i^{n+1} = \frac{1}{N} \sum_{k=1}^N \pi_i^k. \quad (\text{A.6})$$

- **Maximizar  $Q(\Theta, \Theta^n)$**  Neste passo é maximizada a equação A.3 em relação aos parâmetros de cada distribuição considerada. A equação A.3 é equivalente a

$$Q(\Theta, \Theta^n) = \sum_{k=1}^N \pi_1^k [\log p_1(x^k) + \log w_1] + \sum_{k=1}^N \pi_2^k [\log p_2(x^k) + \log w_2]. \quad (\text{A.7})$$

Em que  $p_1$  é a função de densidade de probabilidade da distribuição Gamma e  $p_2$  a da distribuição Rayleigh. Vamos agora apresentar a maximização para cada distribuição, com o objetivo de estimar os seus parâmetros.

- **Distribuição Gamma** A distribuição Gamma tem parâmetros  $\alpha$  e  $\beta$  e é dada pela expressão

$$f(x; \alpha, \beta) = \frac{x^{\alpha-1} e^{-\frac{x}{\beta}}}{\beta^\alpha \Gamma(\alpha)} \quad \text{para } x > 0 \text{ e } \alpha, \beta > 0. \quad (\text{A.8})$$

Para obtermos a estimativa  $\hat{\alpha}$  e  $\hat{\beta}$  maximizamos a expressão A.7 [17] [18]. Ou seja, vamos calcular a derivada parcial de A.7 em relação a  $\alpha$  e a  $\beta$ , respectivamente, e igualar a zero. Por questões de simplicidade de notação vamos deixar de apresentar o índice  $i$ , que indica a distribuição a que se referem as variáveis e função, deixando claro que correspondem todas à distribuição Gamma,  $i = 1$ . Derivando A.7 e resolvendo em relação a  $\beta$ , obtemos  $\hat{\beta}$ :

$$\hat{\beta} = \frac{\sum_{k=1}^N \pi^k x^k}{\alpha \sum_{k=1}^N \pi^k} \quad (\text{A.9})$$

Derivando A.7 em relação a  $\alpha$  obtemos

$$\log \alpha - \psi(\alpha) - \log \left( \frac{\sum_{k=1}^N \pi^k \log x^k}{\sum_{k=1}^N \pi^k} \right) + \frac{\sum_{k=1}^N \pi^k \log x^k}{\sum_{k=1}^N \pi^k} = 0, \quad (\text{A.10})$$

em que  $\psi(\alpha)$  é a função Digamma [19]. Esta expressão não tem solução em forma fechada para  $\alpha$  mas pode ser resolvida iterativamente usando o método de Newton [20]. Considerando a expressão A.10 equivalente a  $f(\alpha) = 0$  e sendo  $\hat{\alpha}^m$  a estimativa obtida para  $\alpha$  na iteração  $m$  do método de Newton, a estimativa seguinte pode ser obtida calculando

$$\hat{\alpha}^{m+1} = \hat{\alpha}^m - \frac{f(\hat{\alpha}^m)}{f'(\hat{\alpha}^m)}. \quad (\text{A.11})$$

O critério de paragem será atingido quando a diferença entre sucessivas estimativas for inferior a um valor  $T_\alpha$  escolhido pelo utilizador,  $|\hat{\alpha}^{m+1} - \hat{\alpha}^m| < T_\alpha$ .

- **Distribuição Rayleigh** A distribuição Rayleigh tem como parâmetro  $\sigma$  e é dada pela distribuição

$$f(x; \sigma) = \frac{x}{\sigma^2} e^{-\frac{x^2}{2\sigma^2}} \quad \text{para } x \geq 0. \quad (\text{A.12})$$

Obtemos a estimativa  $\hat{\sigma}$  calculando a derivada parcial de A.7 em relação a esta variável [21], igualando a zero, e resolvendo em ordem a  $\sigma$ . O resultado é apresentado na expressão A.13. Por questões de simplicidade de notação não apresentamos o índice  $i$ , correspondendo a expressão seguinte à distribuição Rayleigh, com  $i = 2$ .

$$\hat{\sigma} = \sqrt{\frac{\sum_{k=1}^N \pi^k (x^k)^2}{2 \sum_{k=1}^N \pi^k}} \quad (\text{A.13})$$

O critério de paragem do algoritmo EM será atingido quando a diferença entre estimativas sucessivas do vetor de parâmetros for inferior a um valor  $T_\Theta$  escolhido pelo utilizador,  $|\hat{\Theta}^{n+1} - \hat{\Theta}^n| < T_\Theta$ .

## A.2 Homografia

No nosso trabalho vamos usar homografia para obtermos a correspondência entre pontos da imagem e pontos de planos do mundo real [22]. Uma homografia é uma transformação projetiva de pontos em coordenadas homogêneas. Esta pode tratar-se de uma transformação 2D ou 3D. Vamos concentrar-nos no caso a 2D, em que a homografia nos permite obter a transformação entre dois planos, sendo  $\tilde{\mathbf{x}}' = (x', y', 1)$  e  $\tilde{\mathbf{x}} = (x, y, 1)$  pontos em coordenadas homogêneas de dois planos distintos.

$$\lambda \tilde{\mathbf{x}}' = \tilde{\mathbf{H}} \tilde{\mathbf{x}}. \quad (\text{A.14})$$

Como estamos a falar de uma transformação em 2D temos que  $\tilde{\mathbf{H}}$  é uma matriz de dimensão  $3 \times 3$ . Pela definição de coordenadas homogêneas, sabemos que  $\tilde{\mathbf{H}}$  é apenas definida até uma escala. Isto significa que duas matrizes  $\tilde{\mathbf{H}}$  que sejam diferentes apenas de um fator de escala são consideradas equivalentes. A equação A.14 é equivalente a

$$\lambda \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}. \quad (\text{A.15})$$

O ponto  $\mathbf{x}' = (x', y')$  em coordenadas cartesianas pode ser obtido resolvendo a equação A.15. A solução é dada por

$$x' = \frac{h_{11}x + h_{12}y + h_{13}}{h_{31}x + h_{32}y + h_{33}} \text{ e } y' = \frac{h_{21}x + h_{22}y + h_{23}}{h_{31}x + h_{32}y + h_{33}}. \quad (\text{A.16})$$

Para estimar a matriz de homografia entre dois planos é necessário usar correspondências conhecidas de pontos entre os dois planos em questão. Vamos considerar que existe correspondência entre os pontos  $\mathbf{x}_i = (x_i, y_i)$  e  $\mathbf{x}'_i = (x'_i, y'_i)$  e que temos  $n$  correspondências conhecidas. Para estimar uma homografia vamos primeiro reescrever a equação A.15 de forma a ficar no formato  $\mathbf{A}\mathbf{h} = 0$ , sendo  $\mathbf{h}$  a matriz coluna

dos parâmetros da matriz de homografia. Obtemos o resultado

$$\mathbf{A}\mathbf{h} = \begin{bmatrix} x_1 & y_1 & 1 & 0 & 0 & 0 & -x'_1x_1 & -x'_1y_1 & -x'_1 \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -y'_1x_1 & -y'_1y_1 & -y'_1 \\ & & & & & \vdots & & & \\ x_n & y_n & 1 & 0 & 0 & 0 & -x'_nx_n & -x'_ny_n & -x'_n \\ 0 & 0 & 0 & x_n & y_n & 1 & -y'_nx_n & -y'_ny_n & -y'_n \end{bmatrix} \begin{bmatrix} h_{11} \\ h_{12} \\ h_{13} \\ h_{21} \\ h_{22} \\ h_{23} \\ h_{31} \\ h_{32} \\ h_{33} \end{bmatrix} = 0. \quad (\text{A.17})$$

Em que, se tivermos correspondências entre  $n$  pontos, a matriz  $\mathbf{A}$  terá dimensão  $2n \times 9$ .

Para encontrarmos a matriz homografia podemos resolver a equação  $\mathbf{A}\mathbf{h} = 0$  para  $\mathbf{h} \neq 0$ . Esta pode ser resolvida usando decomposição em valores singulares (SVD, do inglês *singular value decomposition*) da matriz  $\mathbf{A}$ :

$$\mathbf{A} = U\Sigma V^T = \sum_{i=1}^9 \sigma_i \mathbf{u}_i \mathbf{v}_i^T. \quad (\text{A.18})$$

A solução de  $\mathbf{h}$  é vetor coluna  $\mathbf{v}_i$  que corresponda ao menor valor singular  $\sigma_i$ , ou seja obtemos assim os coeficientes da matriz homografia  $\mathbf{H}$  que melhor se adapta aos pontos usados.

### A.3 Método de medição da altura de uma pessoa através de uma imagem

Nesta secção vamos apresentar um método para calcular a altura de uma pessoa que esteja presente numa imagem. Este permite-nos obter a relação entre alturas de dois objetos paralelos entre si. Deste modo, se conhecermos a altura de um objeto presente na imagem, como por exemplo de uma porta ou de um armário, conseguimos calcular a altura de uma pessoa. O algoritmo pode ser encontrado em [23].

Vamos considerar uma situação visível numa imagem. A sua geometria 3D está representada na figura A.1a e a geometria da imagem apresentada na figura A.1b. Consideremos o caso de estarem visíveis dois segmentos de reta paralelos entre si e perpendiculares ao plano do chão. Estes poderiam, por exemplo, corresponder à altura de uma pessoa e à de um objeto. Os segmentos estão representados na figura A.1a como  $\mathbf{L}_1$  e  $\mathbf{L}_2$ . Estes segmentos têm um comprimento real de  $d_1$  e  $d_2$ , pontos de topo  $\mathbf{T}_1$  e  $\mathbf{T}_2$  e pontos de base  $\mathbf{B}_1$  e  $\mathbf{B}_2$ , respetivamente. Na geometria 3D é ainda representado o ponto  $\tilde{\mathbf{T}}_1$ , que resulta da interseção de  $\mathbf{L}_2$  com uma linha paralela ao plano do chão e que intersesta  $\mathbf{L}_1$  no ponto  $\mathbf{T}_1$ . Todos estes pontos e segmentos de reta estão também representados na geometria da imagem, na figura A.1b, com a mesma terminologia mas com letras minúsculas.

Consideramos que temos a linha de fuga do plano de chão  $\mathbf{l}$ , o ponto de fuga vertical  $\mathbf{v}$  e os pontos  $(\mathbf{t}_1, \mathbf{t}_2)$  e  $(\mathbf{b}_1, \mathbf{b}_2)$ .

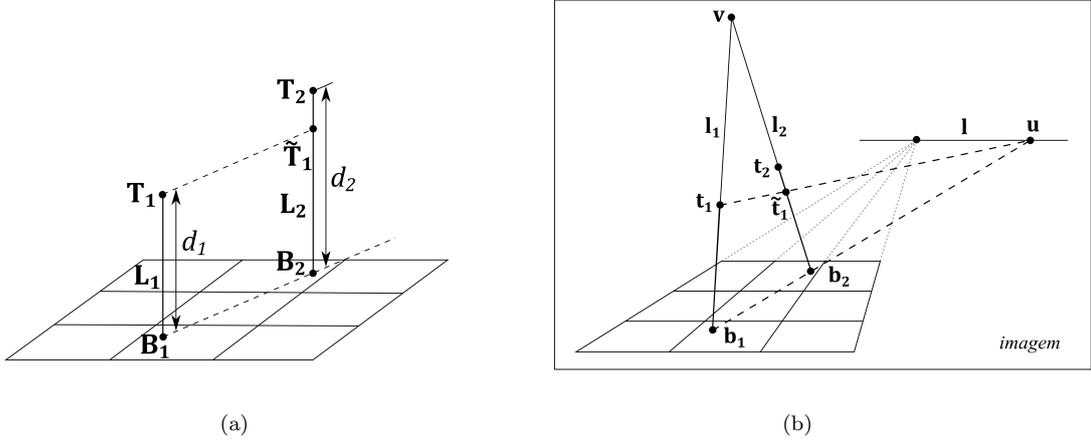


Figura A.1: Geometria do cálculo de rácios de comprimentos entre linhas de cena paralelas. **(a)** Geometria 3D: Os segmentos de reta  $L_1$  e  $L_2$  são paralelos e têm comprimentos  $d_1$  e  $d_2$ , respetivamente. Os pontos  $T_1$  e  $B_1$  pertencem a  $L_1$  e os pontos  $T_2$  e  $B_2$  pertencem a  $L_2$ .  $B_1$  e  $B_2$  pertencem ao plano do chão.  $T_1$  é o ponto de interseção de  $L_2$  com um segmento de reta paralelo ao plano do chão e que intersecta  $L_1$  no ponto  $T_1$ . **(b)** Geometria da imagem:  $l$  é a linha de fuga do plano do chão e  $v$  é o ponto de fuga vertical.  $l_1$  e  $l_2$  são as representações na imagem das linhas em 3D  $L_1$  e  $L_2$ , respetivamente.  $t_1$ ,  $t_2$ ,  $b_1$ ,  $b_2$  e  $\tilde{t}_1$  são representações na imagem dos pontos em 3D  $T_1$ ,  $T_2$ ,  $B_1$ ,  $B_2$  e  $T_1$ , respetivamente. As imagens são adaptadas de [23].

O primeiro passo do algoritmo é obtermos o ponto de fuga  $u$ , que pertence à linha de fuga horizontal  $l$ . Este pode ser obtido calculando a interseção entre  $l$  e o segmento que passa nos pontos  $b_1$  e  $b_2$ . Assim,  $u$  é obtido calculando-se

$$\mathbf{u} = (\mathbf{b}_1 \times \mathbf{b}_2) \times \mathbf{l}. \quad (\text{A.19})$$

De seguida vamos obter  $\tilde{t}_1$ , que resulta da interseção de  $l_2$  com o segmento que passa nos pontos  $t_1$  e  $u$ . Computamos

$$\tilde{t}_1 = (\mathbf{t}_1 \times \mathbf{u}) \times \mathbf{l}_2. \quad (\text{A.20})$$

em que  $l_2$  é encontrado através de

$$\mathbf{l}_2 = \mathbf{v} \times \mathbf{b}_2. \quad (\text{A.21})$$

Considerando que as distâncias dos pontos  $b_2$ ,  $\tilde{t}_1$ ,  $t_2$  e  $v$ , pertencentes a  $l_1$ , são 0,  $\tilde{t}_1$ ,  $t_2$  e  $v$ , respetivamente, o rácio entre os comprimentos das retas  $L_1$  e  $L_2$  do mundo real é dado por

$$\frac{d_1}{d_2} = \frac{\tilde{t}_1(v - t_2)}{t_2(v - \tilde{t}_1)}. \quad (\text{A.22})$$

Deste modo, sabendo  $d_1$  ou  $d_2$ , conseguimos calcular o altura desconhecida.

## A.4 Método de estimação da topologia

Nesta secção descrevemos o método de estimação de topologia apresentado em [11]. Este método permite-nos aprender grafos esparsos com base na comparação de um conjunto de características dos objetos considerados.

Considerando um conjunto de objetos, que serão os nós do grafo, e um conjunto de características que nos permitem comparar os diversos objetos, conseguimos calcular os pesos de todas as ligações. Os pesos de um grafo são representados por uma matriz simétrica  $W$  de dimensão  $n \times n$ , sendo  $n$  o número de objetos considerado. Os dados de características são representados numa matriz  $D$  de dimensão  $n \times m$ , em que  $n$  é o número de objetos considerados e  $m$  o número de características. As colunas de  $D$  são denotadas de características  $\{f^{(1)}, \dots, f^{(m)}\}$ . Um vetor de características,  $f^{(k)} = (f_1^{(k)}, \dots, f_n^{(k)})$ , observado para  $n$  objetos, é modelado por

$$p(f^{(k)} | W) \propto \exp \left( -\frac{1}{4} \sum_{i,j} w_{ij} \left( f_i^{(k)} - f_j^{(k)} \right)^2 - \frac{1}{2\sigma^2} f^{(k)T} f^{(k)} \right). \quad (\text{A.23})$$

Em que  $w_{ij}$  pertence à matriz  $W$ . Este modelo assume que a média das características é zero logo é necessário que os dados sejam escalados de modo a que o valor da média de  $D$  seja zero e que o valor máximo da matriz covariância  $\frac{1}{m}DD^T$  seja um. O parâmetro  $\sigma^2$  pode ser pensado como o valor *à priori* da variância das características. Este modelo é equivalente a uma distribuição Gaussiana  $n$ -dimensional, de média zero e covariância  $\tilde{\Delta}^{-1}$ .  $\tilde{\Delta} = Q - W + I/\sigma^2$  é a matriz de precisão em que  $Q = \text{diag}(q_i)$  é a matriz diagonal de entradas  $q_i = \sum_j w_{ij}$  e  $I$  é a matriz identidade. Restringimos  $w_{ij} \geq 0$  de modo a que o modelo represente apenas correlações positivas.

Para completar o modelo é necessário um prior  $p(W)$ , que é definido como

$$p(W) = \prod_{1 \leq i < j \leq n} \beta \exp^{-\beta w_{ij}} \quad (\text{A.24})$$

O objetivo é obter  $W$ , calculando-se  $\arg \max_W \log p(W|D)$ , que é equivalente a resolver o seguinte problema de otimização convexo

$$\max_{\tilde{\Delta} \succ 0, W, \sigma^2} \log |\tilde{\Delta}| - \text{trace} \left( \tilde{\Delta} \frac{1}{m} DD^T \right) - \frac{\beta}{m} \|W\|_1$$

s.t.

$$\tilde{\Delta} = \text{diag} \left( \sum_j w_{ij} \right) - W + I/\sigma^2 \quad (\text{A.25})$$

$$w_{ii} = 0, i = 1, \dots, n$$

$$w_{ij} \geq 0, i = 1, \dots, n; j = 1, \dots, n$$

$$\sigma^2 > 0.$$

Em que  $\tilde{\Delta} \succ 0$  denota uma matriz simétrica positiva definida. O único parâmetro livre é  $\beta$  - quanto

maior o valor de  $\beta$ , menor serão os valores dos pesos obtidos e conseqüentemente menor o número de ligações do grafo obtido. Quanto maior o número de características observadas, ou seja quanto maior o valor de  $m$ , maior o valor dos pesos. A solução deste problema pode ser encontrada usando o CVX, um pacote do MATLAB® usado para resolver problemas convexos.

## Apêndice B

# Representação da posição de peões no referencial do mundo real

É útil representar a posição dos peões no referencial do chão do mundo real. Isto pode ser feito através do ponto, no referencial da imagem, correspondente à posição dos pés e que é obtido através da *bounding box* (ver figura 4.1). Conhecendo-se este ponto e a matriz homografia que relaciona os dois referenciais, a posição no mundo real é obtida de forma trivial.

Há no entanto casos em que não é possível observar os pés das pessoas na imagem. Primeiro tentámos resolver este problema ao calcularmos a homografia que relaciona os pontos da cabeça de um peão com o referencial do mundo real. Os resultados não foram satisfatórios e este método não foi utilizado. A explicação do método encontra-se explicado no apêndice C.

A solução encontrada foi utilizar um método que nos permite estimar a posição dos pés de um peão no referencial da imagem, mesmo quando estes não se encontram visíveis. Este utiliza o ponto da cabeça do peão e ainda a *bounding box* referente a um outro frame em que a pessoa tenha estado totalmente visível. Encontrada a posição dos pés na imagem, a posição no referencial pode ser obtida com a homografia. Explicamos o método usado na secção seguinte.

### B.1 Método de estimação da posição dos pés de um peão

Este método é baseado no algoritmo apresentado na secção A.3 e tem como objetivo encontrar a posição dos pés de uma pessoa, no referencial da imagem respetiva, quando estes não estão visíveis.

Vamos considerar uma situação em que temos duas imagens adquiridas pela mesma câmara. Em ambas é possível visualizar o mesmo peão, em instantes de tempo distintos. Numa das imagens a pessoa está totalmente visível enquanto que na outra tem os pés fora do limite da imagem. Esta situação é representada na figura B.1. Em B.1a representamos a geometria 3D, em que  $L_1$  e  $L_2$  representam a pessoa nos dois instantes de tempo considerados. Em B.1b representamos a geometria da imagem, estando a pessoa representada por  $l_1$  e  $l_2$ . Os comprimentos dos segmentos de reta  $\mathbf{L}_1$  e  $\mathbf{L}_2$  são dados por  $d_1$  e  $d_2$  e correspondem à altura pessoa. Assim, sabemos que  $d_1 = d_2$ . Vamos utilizar relações geométricas

para obter o ponto  $b_1$  no referencial da imagem, que representa a posição dos pés que queremos obter.

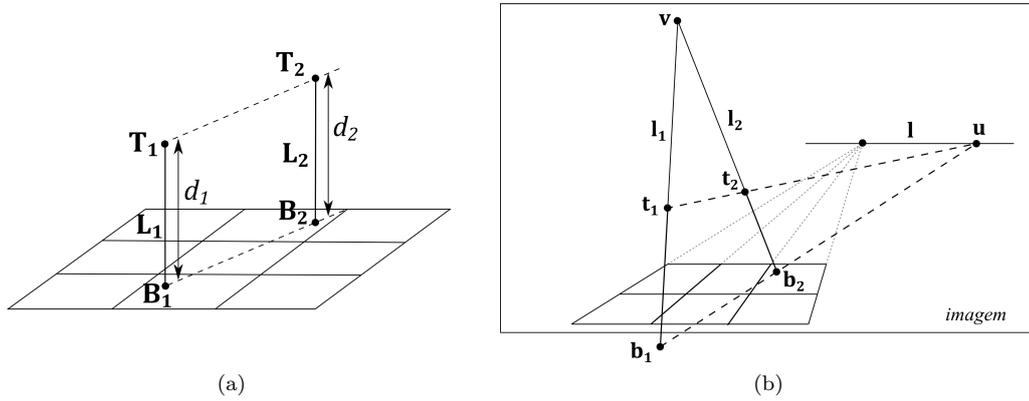


Figura B.1: Geometria para estimar posição dos pés através de imagem. **(a)** Geometria 3D: Os segmentos de reta  $\mathbf{L}_1$  e  $\mathbf{L}_2$  são paralelos e correspondem à mesma pessoa em instantes de tempo distintos. Têm comprimentos  $d_1$  e  $d_2$ , sendo  $d_1 = d_2$ .  $\mathbf{T}_1$  e  $\mathbf{B}_1$  correspondem aos pontos da cabeça e dos pés da pessoa na posição representada por  $\mathbf{L}_1$ . O mesmo para  $\mathbf{T}_2$  e  $\mathbf{B}_2$  em relação a  $\mathbf{L}_2$ . **(b)** Geometria da imagem:  $\mathbf{l}$  é a linha de fuga do plano do chão e  $\mathbf{v}$  é o ponto de fuga vertical.  $\mathbf{l}_1$  e  $\mathbf{l}_2$  são as representações na imagem das linhas 3D  $\mathbf{L}_1$  e  $\mathbf{L}_2$ , respetivamente.  $\mathbf{t}_1$ ,  $\mathbf{t}_2$ ,  $\mathbf{b}_1$  e  $\mathbf{b}_2$  são representações na imagem dos pontos 3D  $\mathbf{T}_1$ ,  $\mathbf{T}_2$ ,  $\mathbf{B}_1$  e  $\mathbf{B}_2$ , respetivamente. É de notar que  $\mathbf{b}_1$  não se encontra representado na imagem e é o ponto que queremos obter. As imagens são adaptadas de [23].

O primeiro passo é obter os parâmetros da linha do horizonte  $\mathbf{l}$ . Para tal, encontramos dois pontos pertencentes a  $\mathbf{l}$ ,  $\mathbf{v}_1$  e  $\mathbf{v}_2$ . Estes pontos são obtidos através da resolução do sistema de equações B.1 em ordem a  $\lambda$ ,  $x$  e  $y$ . Encontramos  $\mathbf{v}_1$  atribuindo os valores  $X = 1$  e  $Y = 0$  e  $\mathbf{v}_2$  atribuindo os valores  $X = 0$  e  $Y = 1$ . A matriz de homografia a usar no sistema B.1 é a da homografia do plano do chão para a câmara com que estamos a trabalhar. Conhecendo-se os pontos  $\mathbf{v}_1$  e  $\mathbf{v}_2$  conseguimos calcular os parâmetros da reta  $\mathbf{l}$ .

$$\lambda \begin{bmatrix} X \\ Y \\ 0 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (\text{B.1})$$

O segundo passo é obtermos o ponto de fuga vertical  $\mathbf{v}$ . Para tal, selecionamos pontos de duas linhas da imagem que sejam verticais no mundo real, selecionando no mínimo dois pontos de cada uma delas. Com estes pontos é possível obtermos os parâmetros associados a cada uma destas duas linhas e depois obtermos o ponto  $\mathbf{v}$  calculando a intersecção entre elas.

É possível agora obtermos os parâmetros da linha  $\mathbf{l}_1$ , pois temos conhecimento de dois pontos que lhe pertencem: os pontos  $\mathbf{v}$  e  $\mathbf{t}_1$ .

O passo seguinte é obtermos o ponto de fuga horizontal  $\mathbf{u}$ , que pode ser obtido através da intersecção da linha  $\mathbf{l}$  com a linha que passa nos pontos  $\mathbf{t}_1$  e  $\mathbf{t}_2$ , ou seja através da equação

$$\mathbf{u} = (\mathbf{t}_1 \times \mathbf{t}_2) \times \mathbf{l}. \quad (\text{B.2})$$

Por fim é possível estimar o ponto  $\mathbf{b}_1$  calculando-se

$$\mathbf{b}_1 = (\mathbf{b}_2 \times \mathbf{u}) \times \mathbf{l}_1. \quad (\text{B.3})$$

É de notar que só é necessário calcular os parâmetros da linha  $l$  e do ponto  $v$  uma vez por câmara, visto que as câmaras estão numa posição fixa.



## Apêndice C

# Estimação da homografia do plano dos pontos da cabeça

Queremos encontrar a matriz da homografia do plano da cabeça  $H_c$ . A homografia do plano da cabeça para uma pessoa, numa determinada câmara, dá-nos a relação entre o ponto da cabeça de uma pessoa numa imagem e a posição da pessoa no plano do chão, apresentada na equação C.1. O ponto que representa a posição no referencial do mundo real  $(X, Y)$  será igual ao que obteríamos com a posição dos pés na imagem e com a homografia do plano do chão, para a mesma pessoa e para a mesma câmara, no mesmo instante de tempo. Visto que as pessoas têm alturas diferentes cada pessoa terá um plano de pontos da cabeça único, por câmara.

$$X = \frac{h_{11}^c x_c + h_{12}^c y_c + h_{13}^c}{h_{31}^c x_c + h_{32}^c y_c + h_{33}^c} \text{ e } Y = \frac{h_{21}^c x_c + h_{22}^c y_c + h_{23}^c}{h_{31}^c x_c + h_{32}^c y_c + h_{33}^c}. \quad (\text{C.1})$$

Reparámos que a equação C.1 e a equação equivalente que utiliza as coordenadas dos pontos dos pés e a homografia do plano do chão são muito semelhantes, variando apenas nos valores da matriz homografia e o valor da coordenada  $y$ . Consideramos então que a matriz  $H_c$  será igual à matriz  $H_p$ , à exceção dos valores  $h_{12}^c$ ,  $h_{22}^c$  e  $h_{32}^c$ , que são os que multiplicam por  $y_c$  e que teremos de calcular.

Para cada câmara e para cada pessoa é necessário obtermos duas imagens em que a pessoa esteja totalmente visível e destas tiramos os pontos  $(x_c^1, y_c^1)$ ,  $(x_p^1, y_p^1)$ ,  $(x_c^2, y_c^2)$  e  $(x_p^2, y_p^2)$ . Com os dois pontos obtidos para a posição dos pés calculamos a respetiva posição da pessoa no mundo real, obtendo-se assim  $(X^1, Y^1)$  e  $(X^2, Y^2)$ .

Utilizando todos estes pontos e os valores conhecidos para  $H_c$  nas equações apresentadas em C.1 obtemos um sistema de quatro equações com variáveis  $h_{12}^c$ ,  $h_{22}^c$  e  $h_{32}^c$ . Dessas quatro necessitamos apenas de utilizar três e resolver esse sistema em função das variáveis que queremos obter. Após este passo conhecemos a matriz  $H_c$ . É necessário realizarmos este cálculo para cada pessoa, em cada câmara. Após isto conseguimos utilizar a matriz de homografia encontrada e um ponto da posição da cabeça na imagem para obter a posição da pessoa no mundo real. É de notar que só conseguimos utilizar este método em pessoas que tenham estado totalmente visíveis em dois *frames*, para uma câmara considerada.



## Apêndice D

# Características do sistema de câmaras

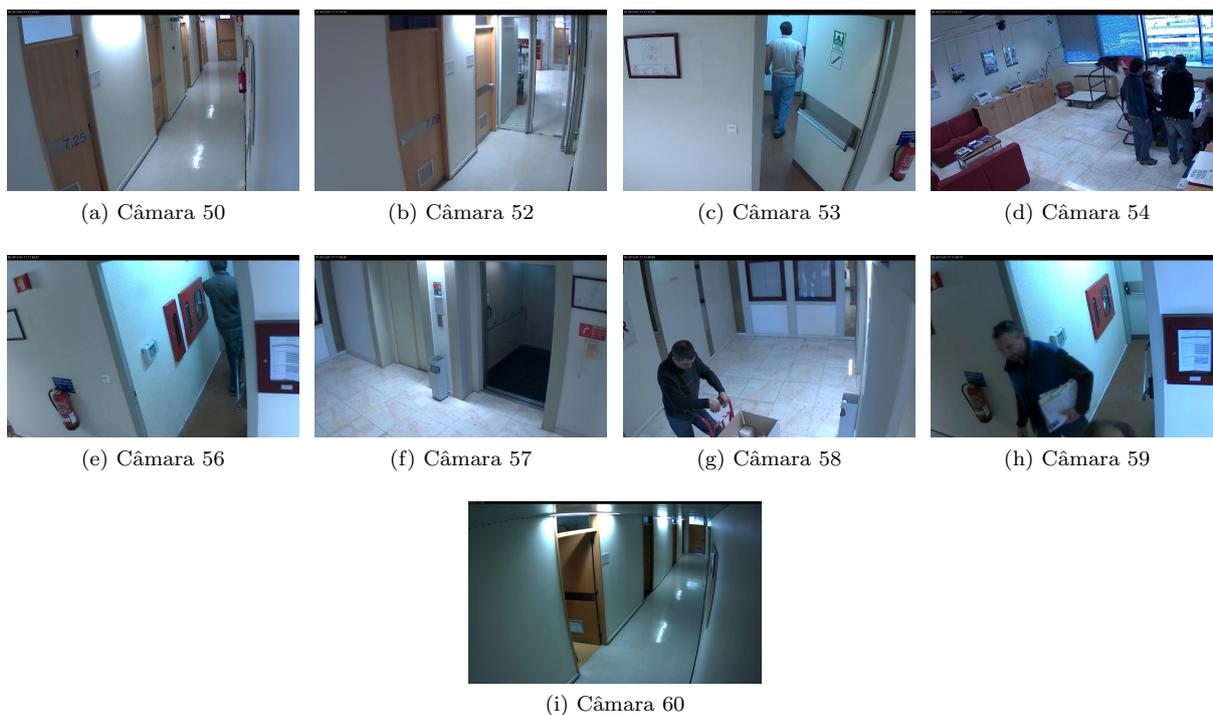


Figura D.1: Exemplos de imagens adquiridas pelas câmaras 50, 52, 53, 54, 56, 57, 58, 59 e 60 do *HDA dataset*.



Figura D.2: Representação das ZES de cada câmara. Estas encontram-se delimitadas a amarelo e estão associadas a um nome identificativo de estrutura  $nnX$ , em que  $nn$  é o número identificativo da câmara correspondente e  $X$  a letra identificativa da zona.

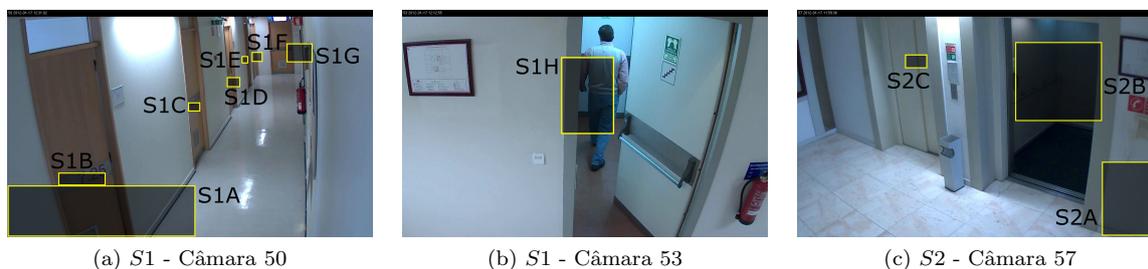


Figura D.3: Representação das ZES das super-câmaras  $S1$  e  $S2$ . Estas encontram-se delimitadas a amarelo e estão associadas a um nome identificativo de estrutura  $SnX$ , em que  $n$  é o número identificativo da super-câmara correspondente e  $X$  a letra identificativa da zona.

Tabela D.1: Dados dos percursos a que vamos aplicar o modelo das distâncias.

<b>Tipo de percurso</b>	<b>Nó 1</b>	<b>Nó 2</b>	<b>Distância (m)</b>
Visível	S3A	S3D	12,1270
	S3A	S3I	4,8720
	S3A	S3B	5,2587
	S3A	S3F	8,0220
	S3D	S3C	2,4129
	S3D	S3B	8,5297
	S3C	S3B	7,4876
	S3C	S3G	15,8709
	S3I	S3B	4,6387
	S3B	S3F	7,7887
	S3B	S3G	10,0447
	S3F	S3H	3,9550
	S3G	S3H	1,7610
	S1A	S1D	8,6376
	S1A	S1E	10,7662
	S1A	S1F	12,5962
	S1A	S1B	1,8151
	S1A	S1H	8,4235
	S1C	S1H	5,2036
	S1D	S1H	4,7734
	S1G	S1B	12,5356
	S1G	S1H	9,8270
	52A	52B	10,5502
	60A	60C	5,1270
	60D	60E	5,0600
	Invisível	S1A	S2A
52A		60E	14,4930

Tabela D.2: Segmentos de percursos escolhidos, respetivas descrições e exemplos.

<b>Tipo de percurso</b>	<b>Segmentos</b>	<b>Características</b>
Visíveis circulares	Zona de espera dos elevadores	É o caso dos percursos dos nós 17A, 17B e S2A.
	Corredores	Casos de percursos circulares em nós de câmaras que filmam corredores. Exemplos: 56A, 60E, S3D
	Elevadores	Percursos de nós correspondentes ao interior de um elevador. Os casos existentes são os percursos circulares dos nós 17C e S2B.
	Salas de convívio	Percursos dos nós localizados no interior das salas de convívio. Os casos existentes são os percursos circulares dos nós S3E e 54A.
Invisíveis circulares	Casa de banho	Caso do percurso circular do nó S3B, que corresponde à entrada para a casa de banho
	Gabinetes	Todos os percursos circulares de nós correspondentes a portas de gabinetes. Exemplos: S1D, 60B, S3G
	Zonas de espera	Caso dos nós 17A e S3E que se encontram, respetivamente, no átrio do elevador e na sala de convívio do 8.º piso.
	Zonas mistas	São os percursos circulares dos nós que se localizam no corredor, significando que as pessoas podem entrar num dos gabinetes ou apenas circular no corredor. Exemplos: 60E, 56A, 52A
Visíveis de espera	Sala de convívio do 8.º piso	Todos os percursos visíveis que passam pela sala de convívio do 8.º piso. São todas as ligações não circulares do nó S3E.
	Zona de espera dos elevadores do 7.º piso	Percursos visíveis entre os nós da zona de espera dos elevadores do 7.º piso. Ou seja, pelos nós S2A, S2B e S2C.
	Zona de espera dos elevadores do 8.º piso	Percursos visíveis entre os nós da zona de espera dos elevadores do 8.º piso. Ou seja, pelos nós 17A, 17B e 17C.