

**UNIVERSIDADE DO ALGARVE**

**SISTEMA DE VISÃO PARA A ORIENTAÇÃO E  
MOBILIDADE EM EDIFÍCIOS PÚBLICOS**

**Miguel Pacheco Serrão**

Dissertação para obtenção do grau de  
Mestre em Engenharia Elétrica e Eletrónica  
Área de Especialização: Tecnologias da Informação e Telecomunicações

Trabalho efetuado sob a orientação de:  
Professor Doutor João Miguel Fernandes Rodrigues

**Março, 2013**



# **Sistema de visão para a orientação e mobilidade em edifícios públicos**

*Declaro ser o autor deste trabalho, que é original e inédito. Os autores e trabalhos consultados estão devidamente citados no texto e constam na listagem de referências bibliográficas incluída.*

---

(Miguel Pacheco Serrão)

## **Copyright**

A Universidade do Algarve tem o direito, perpétuo e sem limites geográficos, de arquivar e publicitar este trabalho através de exemplares impressos reproduzidos em papel ou de forma digital, ou por qualquer outro meio conhecido ou que venha a ser inventado, de o divulgar através de repositórios científicos e de admitir a sua cópia e distribuição com objetivos educacionais ou de investigação, não comerciais, desde que seja dado crédito ao autor e editor.

Faro, 28 de Março de 2013



# Agradecimentos

Devo estas breves linhas a todas as pessoas que me motivaram durante a elaboração desta dissertação, onde devo incluir amigos que sempre me deram aquele incentivo extra necessário nas alturas que mais precisava e principalmente à Ana Pacheco que me incentivou incessantemente todo o tempo, permitindo-me ganhar foças em cada batalha travada. Devo ainda um agradecimento especial ao Professor João Rodrigues que é o responsável por me ter despertado o interesse pela área de visão computacional e por me ter facultado os conhecimentos, que sem os quais nada disto tinha sido possível.



## Resumo

Os sistemas de navegação, como o GPS, têm proliferado de forma surpreendente na última década, assumindo hoje um papel fundamental no nosso cotidiano. No entanto, tais tecnologias são de uso exclusivo em campo aberto e inúteis na utilização em ambientes interiores. Esta dissertação foca o desenvolvimento de um sistema para orientação e mobilidade dentro de edifícios públicos utilizando visão artificial para o reconhecimento de objetos e textos que sirvam de pontos de referência para a localização do indivíduo, tais como sinalizadores, placares de informação textual, extintores de incêndio, portas, escadas, janelas, cacifos, retângulos entre outros e um Sistema de Informação Geográfica para efetuar o enquadramento espacial da informação extraída por visão. A utilização de o Sistema de Informação Geográfica permite ainda o planejamento de uma rota de navegação através da verificação das acessibilidades entre as divisões do edifício caracterizado em mapa. Torna-se assim possível, conhecendo a posição e rota de navegação, orientar o indivíduo no seu trajeto até ao seu destino final. O interface com o utilizador, visando como utilizador final pessoas com deficiência visual, é efetuado com recurso a um sintetizador de voz que comunica verbalmente as informações tanto de reconhecimento como de localização e orientação.

**Palavras-chave:** Acessibilidades; Visão Computacional; Sistemas de Informação Geográfica; Navegação; Reconhecimento de objetos; Reconhecimento ótico de caracteres; Reconhecimento de Portas; Reconhecimento de escadas.





**TITLE:** Vision system for orientation and mobility in public buildings

## Abstract

Navigation systems like GPS, have proliferated in surprising ways over the last decade, assuming a key role today in our daily lives. However, such technologies are used exclusively in the open, but are useless indoors. This dissertation focuses on the development of a system for orientation and mobility within public buildings using artificial vision for the recognition of objects which serve as landmarks for the location of the individual, such as beacons, scores of textual information, fire extinguishers, doors, staircases, windows, lockers, rectangles among others and a Geographic Information System to make the framework of spatial information extracted by vision. The use of the Geographic Information System also allows the planning of a navigation route through verification of accessibility between the divisions of the building featured on the map. It thus becomes possible, knowing the position and route navigation, to guide the individual throughout the journey to his final destination. The user interface, aiming visually impaired people as final users, is made using a voice synthesizer to communicate information verbally of both recognition as location and orientation.

**KeyWords:** Accessibilities; Computer Vision; Geographic Information Systems; Localization and Navigation; Object recognition; Optical Character Recognition; Door Recognition; Stair Recognition.



# Índice

<b>Agradecimentos</b> .....	<b>I</b>
<b>Resumo</b> .....	<b>III</b>
<b>Abstract</b> .....	<b>V</b>
<b>1 Introdução</b> .....	<b>1</b>
1.1 Enquadramento e Objetivos .....	3
1.2 Principais Contribuições .....	5
1.3 Vista Geral .....	5
<b>2 Estado de Arte e Conceitos Gerais</b> .....	<b>6</b>
2.1 Mapas Cognitivos.....	6
2.2 Navegação .....	8
2.3 Reconhecimento e Seletividade de Objetos Baseado em Visão por Computador	11
2.4 Sistemas de Informação Geográfica.....	20
2.5 Sistemas de Navegação para Invisuais.....	23
<b>3 Sistema de Visão para Orientação e Mobilidade</b> .....	<b>28</b>
3.1 Arquitetura do Sistema.....	28
3.2 Processo de Reconhecimento por Visão .....	30
3.2.1 Contextualização Visual .....	32
3.2.2 Reconhecimento de Portas .....	37
3.2.3 Detecção e Reconhecimento de Escadas.....	44
3.2.4 Reconhecimento Ótico de Caracteres.....	49
3.2.5 Reconhecimento de Retângulos e Quadrados .....	51
3.2.6 Reconhecimento de Objetos .....	55
3.3 Processo de Localização e Navegação .....	60
3.3.1 Ordens de Navegação .....	71
<b>4 Resultados</b> .....	<b>74</b>
4.1 Testes.....	74
4.2 Discussão de resultados.....	85
<b>5 Conclusões e Trabalhos Futuros</b> .....	<b>89</b>
<b>Bibliografia</b> .....	<b>95</b>



# Capítulo I

## 1. Introdução

A introdução de modelos de visão em sistemas computacionais tem vindo a permitir grandes avanços no desenvolvimento de aplicações em áreas como a automação industrial, robótica, biomedicina e sistemas de segurança. Na indústria, têm sido desenvolvidas um grande número de aplicações baseadas em visão artificial, principalmente na automatização de métodos rotineiros de deteção de defeitos e inspeção de qualidade de produtos, ou nas linhas de montagem como ferramenta auxiliar para a correta fixação de peças. Para além da área industrial, a segurança também é uma das áreas em que se notou um elevado acréscimo de aplicações, que têm como objetivo principal, a deteção de intrusões ou a análise e o seguimento de situações suspeitas. Recentemente, na área da robótica móvel, têm sido desenvolvidos sistemas que procuram tornar a sua navegação totalmente autónoma. Baseada em captura e análise de imagem, foram propostos sistemas que permitem o rastreio de incêndios, exploração espacial ou missões de reconhecimento militar, através de veículos não tripulados. A biomedicina, por sua vez, tem utilizado a visão computacional como ferramenta de diagnóstico de pacientes fazendo recurso por exemplo a imagens ultrassons e de ampliação microscópica.

Nos anos mais recentes, têm-se reunido esforços para aplicar a tecnologia emergente da visão artificial ao serviço do quotidiano das pessoas com dificuldades visuais, como utensílio valioso para a locomoção, identificação de objetos e avaliação de situações de risco. A navegação de pessoas com deficiências visuais está estritamente limitada ao uso de bengala para a deteção de obstáculos e ao uso de cães guias para o auxílio nas suas deslocações diárias, utensílios que não dispensam, contudo, o uso de mapas mentais armazenados em memória, constituídos por fragmentos de “pontos de referência” que os informam da sua correta posição e ajudam-nos na sua orientação e tomada de decisões. No entanto, num meio desconhecido, não podendo recorrer à sua memória, as pessoas com deficiência visual ficam completamente dependentes de transeuntes para os orientar e enquadrar no ambiente desconhecido.

Por outro lado, o ser humano desde os tempos mais remotos procurou formas de organizar e guardar as informações do mundo por si conhecido, com vista a descrever os limites das regiões e seus territórios, bem como registrar a localização de recursos naturais, minérios, rotas de comércio, culturas etc. Esta necessidade provocou o aparecimento dos primeiros mapas, como projeção simplificada do mundo real, permitindo registrar e organizar esta mesma informação espacialmente. Devido à impossibilidade de aglomerar todo o tipo de informação num único mapa, sobre o risco de o tornar ilegível ou de complexa interpretação, eram geralmente criados diferentes mapas para uma mesma região, cada um com as suas informações, escala e projeções, o que simplifica a interpretação mas dificulta a organização e armazenamento. Outro dos inconvenientes de um mapa convencional, em formato físico é que o mundo como o vemos e conhecemos está em constante alteração, o que provoca conseqüentemente constantes alterações e reformulações aos mapas.

O aparecimento dos sistemas computadorizados veio revolucionar a quantidade de dados possíveis armazenar e facilitar a sua manipulação, o que veio gerar a forte proliferação dos Sistemas de Informação Geográfica (SIG). Um SIG oferece a capacidade de processar grandes quantidades de informação e armazená-la numa base de dados uniforme e estruturada que permite uma manipulação rápida e fiável, por um ou mais indivíduos, através das ferramentas disponibilizadas. Devido aos inúmeros benefícios da utilização dos SIG, estes tiveram uma enorme aplicação, tal como em centros comerciais, aeroportos, hospitais, escolas entre outros que devido às dimensões que têm, necessitam de uma organização detalhada e constante de informações, para definição de trajetos de evacuação posicionamento de equipamentos de extinção de incêndio, localização das diferentes áreas de serviço para definição das áreas comuns e áreas de acesso limitado bem como a descrição dos grupos que têm acesso a cada uma dessas áreas limitadas.

Numa sociedade onde se encontram disponíveis tecnologias como por exemplo o GPS (*Geospatial Positioning System*), o SIG (ou GIS - *Geospatial Information System*) e o RFID (*Radio Frequency IDentification*), que combinadas permitem a navegação de carros (GPS e SIG) e o acompanhamento de contentores (GPS e RFID), bem como outras tecnologias que permitem a emissão de excertos de filmes via *Bluetooth*, pode ser questionado quanto custa facilitar a vida a um cego usando a tecnologia mais elementar

disponível no mercado. Esta tecnologia não vai substituir a bengala, mas pode complementá-la auxiliando a sua navegação dentro de espaços públicos.

## 1.1 Enquadramento e Objetivos

Esta dissertação foca o desenvolvimento de um sistema que permita orientar pessoas com dificuldades visuais dentro de espaços públicos como são exemplo os centros comerciais, museus, escolas etc. Combinando os sistemas de informação geográfica já existentes nesses edifícios com um sistema de visão artificial portátil (que seja propriedade do utilizador e que seja facilmente transportável, como por exemplo um computador *Ultrabook* ou um *Smartphone*), de forma localizar o utilizador dentro do edifício. O sistema proposto pretende ser de baixo custo, de forma a poder permitir abranger a maior quantidade possível da população, consequentemente terá de ser o menos exigente possível em termos de processamento, para com isso permitir ser integrado em sistemas de mais baixo custo e dimensões.

Como referido, o sistema visa localizar um indivíduo dentro de um edifício através da combinação da informação do SIG, com informações visuais de objetos obtidas por meio de uma câmara (e.g. *webcam*, no caso de um *Ultrabook*). Pretende-se usar apenas objetos que estejam inseridos, ou que possam ser inseridas no sistema de SIG, sendo a característica principal a imutabilidade dos objetos (que não alterem a sua forma, cor, etc.) ou mudem de posição ao longo do tempo e caso isso aconteça, sejam objetos que obrigatoriamente tenham/possam ser recolocados no SIG. Exemplos são: portas, janelas, extintores, indentificadores de texto e sinalética diversa tal como saídas de emergência, indicação de escadas, sanitários, indicação de zona, etc.

Os objetivos da dissertação são:

- 1) Preparar “um mapa” e testar o sistema num ambiente público real, como por exemplo o edifício do ISE/UALG. Nesse mapa deve ser criado partindo de um Sistemas de Informação Geográfica existente, onde deve ser marcada toda a sinalética vertical, caso não esteja inserida, bem como a localização das portas e outros objetos relevantes;
- 2) Preparar uma base de dados de imagens de objetos existentes em todo o ambiente do referido mapa;

- 3) Criar um interface de interligação com os Sistemas de Informação Geográfica existentes na UALG, usando para tal a GDAL - *Geospatial Data Abstraction Library*;
- 4) Reconhecer os caracteres de placares de informação mais relevantes (e.g. “Sanitários,” “Sala xyz,” etc.);
- 5) Reconhecer portas, janelas, escadas ou outros objetos relevantes no mapa;
- 6) Determinar a posição do utilizador (câmara no ambiente) pela combinação das características atrás referidas;
- 7) Criar uma rota de navegação de um ponto A no edifício até um ponto B.

A aquisição, tratamento de imagem e os processos de identificação são desenvolvidos fazendo recurso à biblioteca OpenCV (*Open Source Computer Vision*) desenvolvida pela *Intel* e atualmente pela *Willow Garage* e *Itseez* (OpenCV, 2012), conjugada com a biblioteca OpenSURF (Evans, 2009) para o reconhecimento de objetos, e a aplicação externa Tesseract para o reconhecimento de caracteres (Tesseract-OCR, 2007).

A informação obtida através de visão é enquadrada e comparada com as informações armazenadas num Sistema de Informação Geográfica, com o objetivo de determinar o posicionamento espacial. A utilização de um SIG permite ainda efetuar o cálculo da rota, com menor custo de distância associado, até ao destino.

As informações geográficas dos objetos reconhecidos e as relações de vizinhança destes com as diferentes regiões/divisões, são obtidas de um Sistema de Informação Geográfica através da livreria GDAL (*Geospatial Data Abstraction Library*) e o motor GEOS (*Geometry Engine Open Source*) para extração dos modelos geométricos e as propriedades de contiguidade entre regiões/divisões, ambos desenvolvidos e mantidos pela *Open Source Geospatial Foundation* (GDAL, 2012).

Os trabalhos e simulação de funcionamento tiveram lugar no ISE/UALG no âmbito do *Vision Laboratory* - UALG e estão integrados no projeto financiado pela Fundação Ciência e Tecnologia: RIPD/ADA/109690/2009 - *Blavigator: a cheap and reliable navigation aid for the blind*.



## 1.2 Principais Contribuições

Esta dissertação pretende contribuir para o desenvolvimento de uma solução que visa melhorar a mobilidade de pessoas com deficiências visuais. As principais contribuições atingidas são:

- a) Permitir a localização através de um sistema de informação geográfica construído em qualquer aplicação para o efeito, muitas das quais livres de encargos, ou mesmo utilizar os desenhos de arquitetura, desenvolvidos por aplicações como o AutoCad (que é a ferramenta mundialmente aceite como “*standard*” para o desenho técnico de plantas em edifícios e ordenamento paisagístico);
- b) Permitir a localização simplesmente através da combinação do SIG com referências visuais, obtidas com recurso a uma câmara de baixo custo, não sendo necessários outros equipamentos auxiliares;
- c) Estabelecer rotas de navegação e determinação de coordenadas de movimento para o auxílio à navegação.

## 1.3 Vista Geral

No presente capítulo foram introduzidos os tópicos ao tema da dissertação bem como os objetivos pretendidos e principais contribuições. No capítulo 2, referente ao estado de arte, irão ser apresentados os principais conceitos que envolvem as técnicas mais recentes utilizadas nos sistemas de navegação, predominantemente na área da robótica, e os modelos para reconhecimento de objetos e seletividade utilizando visão computacional, serão introduzidos ainda os principais conceitos que caracterizam um Sistema de Informação Geográfica. Finalizando o capítulo serão enunciados alguns dos sistemas que serviram como base inspiradora para esta dissertação. No capítulo 3 irá ser descrita a arquitetura do sistema e efetuada a análise detalhada de cada um dos processos constituintes. No capítulo 4, serão apresentados os resultados e discussão dos mesmos, tendo em conta ensaios de funcionamento realizados no Instituto Superior de Engenharia. Para finalizar, o capítulo 5, contempla as considerações finais e propostas a melhorias futuras ao sistema desenvolvido.

# Capítulo II

## 2. Estado de Arte e Conceitos Gerais

Grande parte das teorias que fundamentam os algoritmos de reconhecimento de objetos e navegação têm uma forte componente apoiada em modelos retirados da forma como o cérebro humano processa a sua percepção da realidade e com ele toma decisões lógicas complexas. A utilização de modelos, apoiados nas recentes descobertas do funcionamento do nosso cérebro não é inocente de todo, pois tais modelos já sofreram milhares de anos de evolução, estando otimizados para com poucos recursos tratar o maior número de informação possível em curto espaço de tempo.

### 2.1 Mapas Cognitivos

O ser humano tem a capacidade de adquirir, armazenar, manipular e utilizar a informação obtida do meio ambiente envolvente, tendo desenvolvido um conjunto de mecanismos evolutivos que o tornam capaz de distinguir objetos e desenvolver eventos e processos complexos tendo em conta a percepção da realidade adquirida pelos seus sensores naturais. O processo cognitivo humano inclui sensação, percepção, pensamento, imaginação, capacidade de raciocínio e resolução de problemas, memória, aprendizagem, e a capacidade de comunicação escrita e falada como forma de transmitir informação e conhecimentos para outros indivíduos, elemento este, que nos difere de qualquer outra espécie (Montello e Sas, 2006).

A navegação pode ser caracterizada como sendo um processo cognitivo, que resulta na procura do melhor percurso até um destino, precedido de um ato físico de deslocação (Darken *et al.*, 1998; Montello, 2005), visando o caminho que apresente o menor custo em tempo e energia despendidos. Segundo Montello e Sas, 2006, a navegação consiste em duas componentes, locomoção e definição do trajeto. A locomoção refere-se à capacidade de um individuo se mover e orientar-se em tempo real, através de um espaço ou área tendo em conta as informações obtidas do meio circundante pelo sistema sensorial (localização), permitindo a deslocação entre caminhos definidos por estradas, passeios e corredores, evitando obstáculos e barreiras que se encontrem na sua

passagem. Deste modo o processo de locomoção exige uma forte coordenação entre o sistema sensorial e motor. A definição do trajeto é o planeamento de uma rota, tendo em conta experiências pré-adquiridas pessoalmente ou através de mapas, até um determinado destino. Muitas vezes uma rota pode ter a necessidade de ser novamente planeada devido a situações inesperadas, por exemplo, o aparecimento de um obstáculo intransponível ou congestionamento do percurso, sendo sempre calculado o trajeto que levará a um menor custo de viagem.

No ato de locomoção existe constantemente a necessidade de nos localizarmos, de forma a determinar qual a posição em que nos encontramos, sendo condição fundamental para as tomadas de decisão posteriores na definição ou redefinição de um trajeto, isto é, será impossível saber que rumo tomar sem primeiro averiguar onde nos encontramos. A localização é efetuada através de características únicas e inconfundíveis que distinguem um determinado local, e que são absorvidas do meio ambiente através dos nossos sensores naturais. Essas características ou pontos de referência são depois comparados com um mapa de natureza cognitiva ou gráfica, dependendo do nível de conhecimento que se tem do meio de navegação, avaliando qual a sua posição atual.

Um mapa cognitivo, ou mental, é criado através de experiências passadas através das sucessivas passagens no local, através de dados pré-adquiridos de um mapa gráfico, ou através de informações passadas verbalmente por outro indivíduo quando não conhecemos a nossa posição exata e pedimos auxílio. Os mapas cognitivos são representações internas, armazenadas em memória, do mundo conhecido e das suas propriedades espaciais. Segundo Golledge, 2002, este mapa mental não consiste num mapa cartográfico, nem em representações fiáveis e à escala, mas sim em pontos discretos e formas geométricas que em conjunto e dispostos hierarquicamente criam uma visão abrangente mas não muito detalhada da realidade. Exemplo disso é se pedirmos a alguém que desenhe o mapa de Portugal de memória, o resultado será algo semelhante mas com muitas falhas em nível de detalhe.

Um mapa cognitivo pressupõe uma fase de aprendizagem para a construção de um modelo mental da realidade. A aprendizagem consiste em alterações do conhecimento ou comportamento que resultam do treino ou experiências adquiridas. A perceção espacial é aprendida através de um ou vários meios de aquisição de informação, como diretamente através da realidade absorvida pelos órgãos senso-motores, ou através de esquemas ilustrativos da realidade, como são exemplo mapas, fotos, desenhos, vídeos

etc. As teorias que envolvem os mapas cognitivos, aqui enunciados, serviram de alicerces para os modelos mais comuns dos algoritmos de navegação, que de seguida iremos analisar.

## 2.2 Navegação

A navegação é um dos problemas fundamentais ao qual se tem procurado resposta de forma a criar sistemas de mobilidade completamente autónomos, principalmente na robótica móvel. Os modelos de navegação podem ser classificados consoante a sua área de referência em modelos locais ou globais. Os modelos locais fazem depender a sua navegação unicamente do meio imediatamente ao seu redor, estando completamente cegos do ambiente para lá do que lhes é possível observar através do seu sistema de sensores. Por sua vez a navegação global lida com o problema da navegação no ambiente no seu todo, muito além do campo de “visão” dos sensores de um robot, permitindo auto-localização e determinação automática de percursos de navegação.

Um dos modelos de localização local mais conhecidos é apresentado por Ulrish e Borenstein, 2000, que é um modelo apoiado num mapa que ilustra a vizinhança imediata, adquirida pelo seus sensores, permitindo a sua orientação e direção de navegação para atingir o destino. No entanto como se trata de um mapa local este é incapaz de efetuar um mapeamento de todo o ambiente e determinar sozinho a rota até ao destino.

Os modelos de navegação global têm sido extremamente estudados e têm sido desenvolvidos significativos progressos no sentido de criar mecanismos que sejam capazes de terem total autonomia na determinação do percurso e capacidade de adaptação a situações imprevistas. As técnicas mais usadas, são através da utilização de mapas pré-construídos ou mapas construídos em tempo real durante a locomoção do robot, segundo modelos de aprendizagem (SLAM – *Simultaneous Localization And Mapping*).

Dentro dos modelos para localização e mapeamento simultâneo (SLAM), Morioka *et al.*, 2011, propõe um modelo que permite a navegação mesmo em ambientes lotados de pessoas como é o caso de centros comerciais ou estações de serviços de transportes. A sua proposta passa pela extração de pontos em 3D da sequência de imagens e cálculo de odometria desses mesmos pontos, de forma a efetuar conclusões quanto ao dinamismo dos objetos, verificando se estes são ou não objetos estáticos. No seu caso de estudo

Morioka *et al.*, 2011, propõe a utilização de algoritmos de extração de pontos de referência baseados no PIRF (*Position Invariant Robust Feature*) proposto por Kawewong *et al.*, 2010. Um algoritmo de SLAM completamente autónomo é apresentado por Saleiro *et al.*, 2012.

Dentro dos modelos baseados em mapas pré-construídos temos várias contribuições de vários autores assentes em métodos de relações topológicas ou métodos estatísticos para a determinação da localização. Kortenkamp e Weymouth, 1994, propõem uma navegação baseada em pontos de referência relacionados topologicamente (mapas topológicos), que visam uma representação compacta do ambiente de navegação, de rápida execução e fácil implementação. A desvantagem deste método é que é fortemente influenciável pelo ruído nos sensores, impedindo-os de distinguir entre dois pontos de referência que sejam muito semelhantes. De forma a criar métodos mais robustos e precisos Arras *et al.*, 2001, propõe um método baseado em filtros de *Kalman* através de métodos estatísticos gaussianos que efetuam a estimação de posição baseada na média e variância das informações calculadas na posição anterior e os dados medidos pelos sensores de odometria e observação geométrica do meio (*scanner* laser e câmara de cores). O método apresentado por Arras tem a desvantagem de não permitir recuperar automaticamente de uma situação de perda de localização.

Dentro dos métodos de localização baseados em processos estocásticos, Fox *et al.*, 1999, desenvolveu um método inspirado nas cadeias de *Markov*, que além de rigoroso permite ainda a recuperação automática em situações em que esteja perdido. Este método utiliza representações probabilísticas em que é estimada a probabilidade de encontrar uma determinada informação a partir da localização atual, isto é, baseia-se nos dados atuais para calcular a probabilidade de detetar no futuro uma determinada referência. Este modelo tem uma componente contínua relacionada com a odometria (dados obtidos através dos sensores de movimento, como o acelerómetro) e outra discreta dependente dos pontos referência detetados durante a deslocação. Este método é um pouco exigente computacionalmente quando tratamos com mapas de grande extensão. Tal modelo veio inspirar modelos mais ambiciosos como o de Seitz *et al.*, 2010, que utiliza as cadeias de *Markov* e as informações de força de sinal e identificação de pontos de acesso *WiFi* para navegação pedestre de um indivíduo num ambiente urbano ou num ambiente *indoor*.

Outro modelo foi apresentado por Simmons e Koenig, 1995, neste caso os autores utilizam as cadeias de *Markov* conjugadas com uma máquina de estados, organizadas topologicamente em função da distância e direção de movimento, este tipo de modelo, designado por POMDP (*Partially Observable Markov Decision Process*), permite uma convergência entre divisões de um espaço muito mais rápida e fiável, no entanto pode apresentar falhas irrecuperáveis quando existem muitos estados possíveis segundo a mesma direção e distância deslocada.

Visando a navegação em grupo Vaughan *et al.*, 2002, construiu um modelo inspirado no sistema biológico das formigas e do comportamento destas na orientação e mobilidade para a execução das suas tarefas. As formigas desenvolveram um dos sistemas de orientação mais eficaz e fiável que existe na natureza, permitindo-lhes sempre escolher o melhor caminho até ao seu destino e recuperar rapidamente no caso de obstrução do caminho. Os algoritmos baseados em colónias de formigas têm atualmente sido amplamente estudados, resultando em progressos bastante interessantes tanto para a robótica como para a implementação de redes de dados robustas a congestionamento de tráfego de pacotes.

Para o caso de estudo desta dissertação são de pouca utilidade, visto ser pretendido desenvolver um sistema uni-agente, mas em termos de perspetivas futuras deverá ser equacionada a hipótese da contribuição de vários utilizadores do sistema para um sistema centralizado, em que todos possam partilhar as informações de tráfego, obstáculos, novos pontos de referência etc. Deste ponto de vista será necessário um sistema multiagente e fortemente cooperativo, como o descrito por Vaughan *et al.*, 2002 e apoiado na filosofia de *AntNet*.

Como podemos verificar, existe uma diversidade de métodos para estimação de posição. Os modelos mais robustos e que apresentaram melhores resultados e mais promissores para navegação, são os métodos que têm por base os filtros de *Kalman* e as cadeias de *Markov*. No entanto ambos fazem depender a sua eficiência fortemente de componentes contínuas, relacionadas com medições de odometria constantes levadas a cabo pelos seus sensores de movimento.

No caso desta dissertação é pretendido criar um modelo simples apoiado unicamente nas informações captadas por uma câmara *Web*, pelo que a estimação da posição será feita unicamente pelas informações detetadas pela câmara. O modelo proposto para a estimação da localização passa então por um mapa com relações topológicas entre os

pontos de referência e pesos relacionados com uma componente discreta de odometria através da estimação da direção do movimento, tendo em conta as referências encontradas.

Concluída a abordagem aos algoritmos computacionais para navegação mais populares, passemos de seguida à análise dos métodos para reconhecimento e seletividade de objetos baseados por visão computacional possíveis de serem utilizados na execução do sistema proposto como tema de dissertação.

## **2.3 Reconhecimento e Seletividade de Objetos Baseado em Visão por Computador**

A função principal da visão humana é o reconhecimento de cenas ou objetos, através da procura de elementos familiares que chamem a nossa atenção. Para que o elemento seja reconhecível é necessário que exista um modelo mental expresso em palavras, imagens, memórias, e características particulares, permitindo a sua fácil organização e classificação por categorias, na nossa memória. A um modelo mental está sempre associado um marcador que permite a identificação individual de cada um dos elementos, no nosso sistema cognitivo. As características que definem esse modelo mental diferem de pessoa para pessoa, devido a ser constituído por experiências individuais, tendo por base experiências vividas, podendo ser complementadas pela troca de impressões com outros indivíduos. As imagens mentais não são completamente detalhadas sendo só armazenadas as características como cor, arestas e forma geométrica que o tornam distinto das demais imagens mentais armazenadas (Russ, 2011a).

Quando os modelos são apoiados unicamente em descrições passadas por experiência de outros indivíduos, são geralmente efetuadas associações com outras experiências já vivenciadas de forma a criar uma imagem mental que possa definir o objeto, vejamos o exemplo de quando alguém nos pede algum artigo de supermercado, por exemplo um pacote de leite, e imaginemos que nunca o vimos na realidade. Para o descrever a pessoa diz que o seu formato é de um paralelepípedo, sendo verificado na nossa “base de dados” qual o formato de um paralelepípedo, de seguida essa mesma pessoa diz que a cor da embalagem é verde, e nós procuramos na nossa “base de dados” qual o aspeto da cor mencionada. Desta forma, é nos possível criar uma imagem mental

do objeto unicamente através dos atributos mencionados, sendo assim possível identificar facilmente o objeto quando formos ao supermercado. No entanto imaginemos agora, que na nossa “base de dados” interna não tínhamos qualquer referência à palavra paralelepípedo, que nunca tínhamos visto a sua forma, de igual modo que não tínhamos armazenado qualquer característica sobre a cor verde. Seria-nos completamente impossível identificar o produto a nós requerido, devido a não termos armazenados nenhuns dados para referência.

A classificação de objetos por categorias (categorização) permite-nos assim de uma forma fácil e rápida identificar um objeto familiar, devido a um conjunto de pontos, linhas, formas geométricas e cores que o categorizam e identificam como pertencente a uma dada família. Por exemplo quando vimos uma mesa facilmente a identificamos como tal, mesmo que nunca tenhamos visto aquela mesa em particular. Esse conjunto de informações, através de pontos, linhas e formas geométricas, são informações discretas que permitem uma abstração muito maior à posição, luminosidade e mesmo obstrução parcial, permitindo o reconhecimento independentemente das condições de luz ou se este se encontra numa posição diferente.

Segundo Russ, 2011a, a técnica que nos permite identificar e reconhecer um objeto, na natureza é através de comparações com os modelos armazenados em memória, que muitas vezes não se encontram segundo as orientações e tamanhos que temos do objeto original em memória, sendo consequentemente necessário um conjunto de operações de rotação e redimensionamento de forma a permitir a comparação e identificação possível.

O processo de reconhecimento (podendo ser considerado a última fase da categorização) foi durante muito tempo interpretado como sendo um processo hierárquico compreendido por diversas fases (detecção, segregação, categorização e reconhecimento). No entanto tal modelo não deverá ser visto inteiramente como sendo uma sequência de processos, existem estudos que indicam que alguns dos processos podem efetivamente não ocorrer sequencialmente (Rodrigues e du Buf, 2009).

O reconhecimento fiável e robusto de objetos é uma das tarefas mais difíceis, devido à complexidade envolvida. Contudo existem alguns modelos com resultados comprovados no reconhecimento de objetos, independentemente do ângulo de visualização e rotação deste em relação ao original. Alguns desses modelos foram popularizados, entre os quais o SIFT - *Scale Invariant Feature Transform* (Lowe, 2004) e o SURF - *Speeded Up Robust Feature* (Bay *et al.*, 2008), e melhorados por outros



autores levando mesmo à criação de uma grande diversidade de variantes. Outros ainda, apresentados recentemente, oferecem grandes perspectivas de futuro principalmente devido à sua rapidez de execução ou base científica. Entre os quais o ORB - *Oriented Fast and Rotated BRIEF* (Rublee *et al.*, 2011) e CHoG - *Compressed Histogram of Gradients* (Chandrasekhar *et al.*, 2012).

O SIFT introduzido por Lowe, 2004, tornou-se um dos algoritmos mais populares para descrever objetos, essencialmente devido à sua eficiência e robustez na utilização em aplicações de reconhecimento de objetos, existindo neste momento várias variantes com foco em módulos do algoritmo original, com o objetivo de o otimizar, entre esses podemos referenciar o SURF. O SIFT é composto por 4 etapas que consistem em: (a) identificar os pontos-chave através dos extremos máximos e mínimos obtidos com recurso a filtros de diferenças gaussianas em diferentes escalas; (b) localizar os pontos-chave e eliminar os candidatos que apresentam baixo contraste; (c) determinar a orientação dos pontos-chave; (d) construir um descritor composto pelas várias informações de orientação e respetivos gradientes de cada um dos pontos-chave identificados.

Na primeira etapa do algoritmo é utilizada uma técnica de reamostragem, em que o tamanho da imagem é sucessivamente redimensionado para metade (oitava), sendo cada uma dessas imagens redimensionadas, ou oitavas, sujeitas a um processo de filtragem utilizando diferenças de Gauss (*DoG – Difference of Gaussian*), para a determinação dos pontos-chave. O processo de diferenças gaussianas consiste na subtração de várias réplicas de uma mesma oitava em que são aplicados filtros de convulsão de Gauss, com valores de variância múltiplos entre si. Para a determinação dos candidatos a pontos de interesse são analisadas e comparadas as imagens, resultantes da diferença gaussiana, com valores de variância adjacentes, através de uma matriz de convulsão entre os 8 pixéis vizinhos a cada um dos pixéis da imagem e os 9 pixéis das imagens de variância imediatamente superior e inferior. Através desse método são identificados os extremos da imagem (Lowe, 2004).

A segunda etapa consiste na localização dos máximos e mínimos, resultantes da diferença gaussiana através de técnicas de interpolação, e na eliminação dos extremos que estejam abaixo de um dado limiar, de forma a retirar os candidatos que apresentem baixo contraste.

Na terceira etapa são calculados a magnitude e orientação do gradiente dos vizinhos em redor do ponto de interesse, considerando uma janela de  $4 \times 4$  centrada no ponto de interesse, sendo construído um histograma de orientações de 36 posições, para representar os ângulos definidos entre o intervalo de  $0^\circ$  a  $360^\circ$ . Cada um dos vizinhos contribui para a orientação correspondente, com o seu valor de magnitude pesado por um fator dependente de uma janela circular gaussiana que é aplicado ao espaço amostra, e que é função da variância da imagem *DoG* que deu origem ao candidato.

Na quarta e última etapa, é criado o descritor dos pontos de interesse que é formado por um vetor contendo os valores das 8 orientações mais relevantes de cada um dos vizinhos (segundo uma janela de  $4 \times 4$  pixéis), resultando num vetor de 128 elementos ( $4 \times 4 \times 8$ ) por cada ponto de interesse detetado. Nesta etapa são aplicadas algumas técnicas para tornar o descritor invariante à rotação e à diferença de iluminação, através da rotação dos pontos relativamente à orientação do ponto de interesse e normalização dos valores do vetor. A correspondência (*matching*) é executada comparando as distâncias euclidianas entre os descritores computados da imagem e os descritores guardados na base de dados. Para aumentar a performance, são utilizadas técnicas de procura do vizinho mais próximo seguido da aplicação de transformadas de *Hough* (*Hough*, 1962) para identificação de agrupamentos pertencentes ao mesmo objeto (*Lowe*, 2004).

O SURF (*Bay et al.*, 2008) tem em vista a criação de um descritor de execução mais rápida não descuidando, contudo, da robustez essencial para o reconhecimento de objetos. Como já referido, muitos dos aspetos foram baseados no SIFT, diferindo deste unicamente na utilização de técnicas que despendessem menor tempo de execução tais como o uso de integrais de imagem e matrizes de *Fast-Hessian*. O SURF pode ser definido por 3 etapas: (a) Criação do integral de Imagem; (b) Determinação dos candidatos a pontos de interesse através da *Fast-Hessian*; (c) Criação do Descritor.

O SURF determina os pontos de interesse através de matrizes de *Hessian* a partir de uma imagem auxiliar que é resultado de uma operação de integração da imagem original. A integração de imagem é uma técnica em que cada ponto armazena a soma dos pixéis de uma área retangular entre o ponto de origem e o ponto em análise, criando uma imagem com regiões em mancha (*blob*).

A *Fast-Hessian* utiliza um sistema de filtros com várias orientações, para detetar estruturas em mancha (*blob detector*), a utilização deste tipo de filtros além de ser mais

rápida, preserva ainda as altas frequências de imagem que podem ser perdidas quando se efetua escalonamento de uma imagem (como no SIFT). Estes filtros além de verificarem a resposta individual em cada uma das orientações verificam também a resposta à mudança de escala, através do aumento gradual do tamanho do filtro (Bay *et al.*, 2008). Este método permite uma diminuição significativa no tempo de processamento, dado que as respostas às diferenças de escala e rotação são analisadas unicamente com a transformação dos filtros tanto na sua orientação como no seu tamanho, ao contrário do seu homólogo SIFT que necessita de exigentes transformações de imagem, criando um conjunto de imagens para efetuar as operações necessárias, resultando num processo muito mais exigente no processamento.

Detetados os pontos de interesse é efetuada a sua localização através de técnicas de interpolação nas diferentes escalas de análise, sendo ainda retiradas as informações de orientação do ponto de interesse em torno de uma área circular de raio predeterminado (Bay *et al.*, 2008). Finalmente para descrever os pontos de interesse, é utilizada uma janela quadrática centrada no ponto de interesse e de orientação idêntica ao ponto-chave a descrever. Essa janela é depois segmentada em conjuntos de 4×4 pixéis e retiradas as informações de orientação desse conjunto através da soma das direções segundo a componente de  $x$  e de  $y$ , bem como os seus valores absolutos. Estas 4 componentes irão formar o descritor para esse conjunto, resultando num descritor formado por 64 valores (16×4). O processo de correspondência (*matching*) entre os descritores obtidos de uma imagem capturada e os descritores de objetos guardados na base de dados é realizada através da medição da distância euclidiana entre os descritores, no entanto para que o processo seja mais rápido são só comparados os descritores que tenham o mesmo contraste, através de uma pré-análise do sinal do Laplaciano. Dois descritores só são comparáveis caso tenham o sinal do Laplaciano idêntico, ou seja tenham o mesmo contraste.

Ao contrário do SIFT, em que são criadas várias imagens, cada uma com uma escala diferente, o SURF tem por base sempre a mesma imagem mudando unicamente o tamanho do filtro Hessiano a aplicar, resultando num processo mais rápido. O SURF segundo o autor (Bay *et al.*, 2008) apresentou resultados bastante mais rápidos para a descrição de objetos, e apresentou melhores resultados que o SIFT em imagens muito influenciadas por ruído.

No entanto a procura de métodos cada vez mais robustos e de maior eficiência computacional tem sido incessável, principalmente devido à necessidade crescente da sua introdução em aplicações para dispositivos móveis que apresentam processadores mais limitados, resultado das exigências de autonomia e tamanho. Na procura de solução para os anseios da comunidade científica Rublee *et al.*, 2011, propôs o ORB (*Oriented Fast and Rotated BRIEF*). O método apresentado pelo autor conjuga um detetor FAST (*Features from Accelerated Segment Test*), apresentado por Rosten e Drummond, 2006, para a deteção de pontos com um descritor extremamente rápido baseado no BRIEF proposto por Calonder *et al.*, 2010. O ORB é constituído basicamente por duas etapas: (a) Determinação dos pontos-chave, através da aplicação do algoritmo FAST, num conjunto de imagens em diferentes escalas. (b) Criação de um descritor para cada um dos pontos-chave detetados, através da aplicação do algoritmo de BRIEF.

O primeiro passo do algoritmo é criar um conjunto em pirâmide de imagens em diferentes escalas, e aplicar o algoritmo de FAST para a deteção dos pontos-chave em cada uma das imagens. Este método possibilita a deteção dos candidatos em diferentes escalas, introduzindo ao FAST a componente necessária da invariância à escala. O algoritmo FAST deteta os pontos de interesse através da diferença de intensidade entre um pixel central e os seus vizinhos dispostos numa janela circular de raio de 16 pixéis, sendo muitas vezes descrito, por vários autores, como um algoritmo de deteção de cantos. Devido à grande sensibilidade que apresenta na deteção de pontos pertencentes a contornos de objetos, o autor sugere a medição das repostas através do detetor de *Harris* (Harris e Stephens, 1988), para efetuar a distinção entre contornos e cantos, e eliminar todos os candidatos que tenham repostas coincidentes com contornos.

O segundo passo do algoritmo é criar um descritor com recurso ao BRIEF (Calonder *et al.*, 2010), este modelo de descritor foi escolhido pelo autor pela sua extrema rapidez na construção dos descritores e nas operações de correspondência (*matching*). Devido ao BRIEF ser desprovido de mecanismos que o tornem invariante à rotação, o autor propõe um modelo em que as operações do BRIEF são rodadas em relação à orientação do ponto de interesse. A orientação do ponto por sua vez é calculada tendo em consideração o ângulo do vetor formado pelo ponto de interesse e o seu centróide, que no caso dos cantos de um objeto nunca são coincidentes. O descritor BRIEF é baseado num vetor de *bits*, que é construído através de testes binários num conjunto de pixéis em

torno de um ponto central da imagem. Cada *bit* do vetor de descrição representa uma comparação simples entre dois pontos dentro de uma secção da imagem, em que toma o valor de '1' caso o primeiro ponto tenha uma maior intensidade que o segundo e '0' caso contrário. A correspondência entre os pontos de dois descritores, para verificar a similaridade entre dois objetos, é efetuada utilizando o algoritmo de *Hamming* para as distâncias (Hamming, 1950). Como os descritores são puramente constituídos por valores booleanos, pode ser utilizado um operador XOR diretamente, o que torna a operação de correspondência extremamente rápida.

No mesmo sentido, da integração de modelos de reconhecimento em aplicações móveis, Chandrasekhar *et al.*, 2012, propôs o CHoG (*Compressed Histogram of Gradients*), que utiliza um sistema de quantificação e compressão de histogramas construídos com base na análise do gradiente nas componentes *x* e *y* na área em redor dos pontos de interesse detetados, sejam eles cantos, arestas, ou regiões segmentadas (*blob*). A correspondência (*matching*) é efetuada através da comparação entre as distâncias dos histogramas dos descritores do objeto capturado e o(s) objeto(s) armazenados em memória.

Segundo os autores o ORB apresenta-se como uma boa alternativa aos descritores SIFT e SURF, apresentando taxas de reconhecimento mais altas em tempos de execução consideravelmente menores. Relativamente ao CHoG, os autores referem que é mais eficiente e robusto que o SURF ou o SIFT, sendo especialmente desenhado com o intuito de ser incorporado em aplicações desenvolvidas para dispositivos de fraco processamento. Devido ao seu sistema de compressão os descritores para os objetos ocupam pouco espaço de armazenamento em comparação aos outros métodos citados.

Dos modelos apresentados para reconhecimento o ORB é o que apresenta melhores performances (em relação ao SURF e ao SIFT) em tempos de processamento mais reduzidos. No entanto devido a ser um algoritmo recente, a sua aplicação em sistemas que envolvam reconhecimento de objetos ainda é pouco utilizada. Por outro lado o SURF é um dos algoritmos mais utilizados em aplicações e uma referência para o reconhecimento de objetos, oferecendo uma maior maturidade e resultados comprovados. O SURF apresenta-se atualmente como a melhor escolha entre os algoritmos mencionados para o reconhecimento de objetos.

Do ponto de vista da seletividade, devido ao excedente de informação que chega ao nosso cérebro proveniente da visão, o nosso sistema biológico criou uma estratégia que

permite aumentar a rapidez de processamento para a detecção e interpretação de situações críticas e informações úteis. Esta estratégia baseia-se no processamento detalhado de apenas algumas partes do campo visual, correspondentes ao Foco-de-Atenção normalmente representado por mapas de saliência (Itti e Koch, 1998), muitas vezes designados apenas por regiões de interesse – RoI (*Region of Interest*). A seleção destas regiões é realizada por um mecanismo evolutivo que permite a detecção imediata de situações de risco em detrimento de outras informações menos prioritárias e irrelevantes que não podem ser processadas simultaneamente (Desimone e Duncan, 1995), atuando deste modo como uma estratégia natural que assegura que a informação selecionada é a mais relevante, para executar ações prioritárias ou ir de encontro a um conjunto de objetivos imediatos.

Segundo Treisman e Gelade, 1980, o sistema cognitivo visual, é composto por duas etapas sequenciais. A primeira etapa é realizada através da retina, sendo depois transmitida para o córtex visual, onde informações relacionadas com deslocamento, disparidade, cor, orientação e forma são retiradas segundo processos paralelos. Após a primeira etapa ter sectorizado as informações, existe uma segunda etapa de nível mais alto que é responsável por reconhecer e interpretar as informações extraídas, através de comparações com informações guardadas em memória e que foram apreendidas durante as experiências vivenciadas do individuo (Treisman e Gelade, 1980).

Este modelo biológico serviu de inspiração a modelos computacionais, com o intuito de poder acelerar o processo de processamento de imagem digital, para a detecção de objetos, utilizando técnicas de seletividade de informação, tendo ao longo dos últimos anos sido propostos vários modelos entre os quais, o modelo apresentado por Koch e Ullman, 1985, e aperfeiçoado por Itti e Koch, 1998.

O modelo apresentado por Itti e Koch é talvez o modelo com mais difusão, sendo baseado no facto de existirem pontos de maior contraste, em relação às regiões vizinhas, que têm forte probabilidade de serem pontos de interesse, tais como diferenças de cor, intensidade, orientação ou movimento. Numa primeira etapa a imagem é segmentada em vários mapas sensíveis a diferentes características como a cor, a orientação, intensidade etc. depois são criados mapas de saliência, onde são registadas as regiões que apresentam diferenças significativas em relação às regiões vizinhas de acordo com a característica em análise. Após terem sido criados os diferentes mapas de saliências, é aplicado um filtro que pretende simular o nosso campo visual dando maior relevância às

áreas centrais da imagem e menor relevância às áreas que se encontram mais afastadas do ponto central da imagem. De seguida é efetuada a integração desses mapas num único mapa onde são marcadas as regiões com maior relevância. Finalmente são selecionadas as regiões por grau de maior realce, tendo sido implementado um mecanismo de inibição de retorno IoR (*inhibition of return*) que previne a escolha de uma região que já tenha sido detetada.

Outro modelo apresentado foi de Le Meur *et al.*, 2004, que tem em consideração a sensibilidade visual, isto é tem em consideração que a visibilidade humana é restringida por determinados parâmetros que diminuem ou aumentam consideravelmente a percepção e rigor da imagem captada, entre os quais a luminescência e diferenças cromáticas. A sensibilidade visual tem em conta que quanto maior o contraste entre regiões, maior será o interesse associado, existindo um valor de *threshold* para o qual pequenas diferenças tendem para não ser perceptíveis pelo sistema visual.

Outros modelos foram apresentados tendo em conta mapas de disparidade, utilizando câmaras *stereo*, para criar uma imagem com as diferenças de profundidade realçando os objectos do ambiente de visão (Meger *et al.*, 2008). Mais recentemente, Martins *et al.*, 2009, propõe um modelo simples para a criação de mapas de saliência baseados na conjugação das informações de cor e textura extraídos da imagem. O autor utiliza técnicas para segregar as zonas onde existam diferenças acentuadas de cor ou textura e efetua a soma das duas componentes extraídas para criar um mapa de saliências.

A seletividade do sistema de visão é uma das importantes características evolutivas que nos facultou o processamento mais rápido de informação pelo cérebro, focando a atenção nos dados que apresentam maior importância e evitando o gasto de recursos no tratamento de informação supérflua. O aproveitamento desta técnica para os sistemas de visão assistida por computador é de extrema importância, visto permitir obter resultados de processamento mais rápidos.

Através das técnicas aqui referidas torna-se possível efetuar o reconhecimento de objetos de uma imagem. Contudo tendo em vista a localização, será ainda necessário enquadrar os objetos reconhecidos espacialmente de modo a prever qual a localização correta. Essa tarefa será auxiliada, no caso concreto desta dissertação, por um Sistema de Informação Geográfica e como tal, será necessário rever alguns conceitos no que diz respeito à sua filosofia e organização de dados, os quais serão de seguida abordados.

## 2.4 Sistemas de Informação Geográfica

Um SIG oferece a capacidade de processar grandes quantidades de informação e armazená-la numa base de dados uniforme e estruturada que permite uma manipulação rápida e fiável, por um ou mais indivíduos, através das ferramentas disponibilizadas. De forma a permitir uma filosofia multiagente e de fácil interpretação os SIG utilizam uma organização disposta em camadas (*layers*), para sectorizar a informação em temáticas, permitindo a intervenção de vários projetistas de forma independente, e a integração simples e interrelacionada dos dados (de By, 2001a). Um SIG permite além do referenciamento geográfico de informação, estabelecer relações métricas e topológicas entre regiões e objetos, quanto à sua contiguidade e distância, facultando de forma fácil e rápida a extração de informações de vizinhança.

Um objeto ou região de um SIG, é definido por coordenadas geográficas e por um conjunto de campos, de identificação e atributos, dispostos sobre a forma de tabela de dados, que conferem uma modelo relacional comum entre todos os objetos e regiões. Através desse modelo relacional é possível efetuar operações de comparação, procura ou seleção por intervalo de pesquisa, típicos de qualquer base de dados moderna, através de pedidos externos efetuados por “*Queries*”, tornando o SIG um sistema aberto a consulta por parte de aplicações de terceiros (de By, 2001b).

As duas principais formas de representação gráfica de um SIG são, o modelo matricial, ou *raster*, e o modelo vectorial. Ambos surgiram como solução para a estruturação de dados gráficos, contudo tanto um como o outro possuem vantagens e desvantagens consoante a utilização a que se destinam (Goodchild, 2012). A grande maioria dos SIG suporta transformações entre os dois modelos consoante os objetivos desejados. O modelo matricial (*raster*) é um método muito utilizado em mapas digitalizados, que divide o espaço do mapa numa matriz retangular, constituída por pixéis. As coordenadas que definem cada uma das localizações no mapa matricial não necessitam de ser armazenadas, visto ser possível saber a posição da célula relativamente ao seu ponto de origem na matriz, e saber o enquadramento da matriz relativamente ao seu posicionamento geográfico global. Por outro lado o método vectorial representa a informação de um mapa como uma coleção de pontos, linhas, e polígonos. Os pontos são representados por coordenadas e são a estrutura mais simples, que organizados sequencialmente e em conjunto formam as restantes estruturas como linhas e polígonos (Goodchild, 2012).



Devido a serem uma representação de pontos, os polígonos e linhas são estruturas que não perdem detalhe, independentemente da escala de trabalho, possibilitando a alteração de escala sem que com isso haja perda de resolução, sendo por isso a representação gráfica optada para os sistemas CAD (*Computer-Aided Design*), em que se trabalha a diversas escalas, para atingir vários níveis de detalhe. O modelo matricial é muito mais rápido de trabalhar, contudo o modelo vetorial é muito mais detalhado e rigoroso (Goodchild, 2012).

Um dos pontos sensíveis num sistema de informação geográfica é o sistema de referência espacial. Esta tarefa tem sido especialmente dificultada devido à forma não uniforme da terra e às irregularidades da superfície terrestre, que têm levado à criação de modelos geométricos e matemáticos que conferissem a maior precisão possível para a determinação de posição através de um sistema de coordenadas. Um dos primeiros modelos existentes descrevia a terra como sendo puramente esférica, no entanto veio a verificar-se que na realidade esta é achatada nos polos, devido às forças gravíticas e ao movimento de rotação da terra, sendo melhor descrita por um elipsoide de revolução, que é caracterizado por um sub-eixo-maior, um sub-eixo-menor e um “grau” de achatamento. Contudo, o elipsoide não resolve só por si o problema das irregularidades topográficas da superfície terrestre, devido às diferentes elevações do terreno, o que deu à origem ao conceito de geoide, que é definido como sendo um modelo físico para a superfície terrestre equipotencial (superfície de potencial gravitacional constante) e que em média coincide com o valor médio do nível do mar (Bolstad, 2012).

Atualmente, o modelo global para o elipsoide mais utilizado é o modelo WGS84 (*World Geodetic System 1984*) que é o modelo de referência para o sistema global de posicionamento GPS, bem como o modelo utilizado para caracterizar o sistema de coordenadas dos mapas utilizados para esta dissertação.

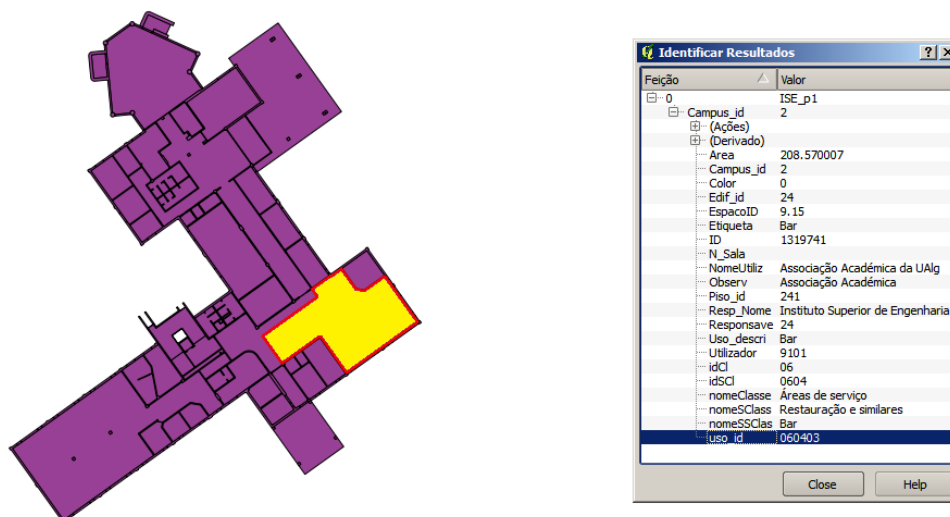
Estes modelos por sua vez permitem descrever um ponto na superfície terrestre por coordenadas geográficas de latitude, longitude e elevação acima do nível médio do mar. No entanto, para trabalhar em unidades que facilmente sejam utilizadas para o cálculo de distâncias e áreas é necessário utilizar um sistema cartesiano simples com um sistema de coordenadas  $x$ ,  $y$ , e  $z$ , em vez das coordenadas geográficas de latitude e longitude. A conversão de coordenadas geográficas em coordenadas cartesianas é efetuada através da transformação do elipsoide terrestre numa superfície plana equivalente. Contudo este processo de transformação implica distorção de áreas,

distâncias, ângulos e direções, devendo ser tida em consideração as características a preservar antes de escolher a técnica de projeção, pois cada uma salvaguarda características distintas. As principais técnicas de projeção são efetuadas através de métodos matemáticos que têm por base a projeção do globo terrestre sobre uma superfície plana, ou sólidos geométricos de revolução como o cilindro e o cone (Bolstad, 2012).

Existem diversos modelos de projeção desenvolvidos, no entanto os mais comuns atualmente são; a projeção Transversa de Mercator UTM (*Universal Transverse Mercator*) baseada num cilindro transverso; projeção de Robinson baseada num pseudo-cilindro; projeção de Albers baseada num cone (Bolstad, 2012). Atualmente, o UTM é o modelo recomendado pelo comité das Nações Unidas, preservando as informações relacionadas com a forma, e possibilitando a medição precisa de distâncias. Existem também, diversas aplicações para manipulação de sistemas de informação geográfica, que disponibilizam ferramentas gráficas que tornam possível a edição, criação e visualização de objetos num espaço cartesiano através de um interface totalmente “*user friendly*”. No caso concreto desta dissertação foi utilizada a aplicação *Open Source Quantum GIS V1.7.1* (QGIS, 2011) da OSGeo (*Open Source GeoSpatial Foundation*), para a introdução de objetos de referência complementares como extintores, placas de sinalização, cacifos etc. A escolha por esta aplicação deveu-se ao fato de ser atualmente uma das ferramentas gratuitas para manipulação de Sistemas de Informação Geográfica mais populares e de fácil utilização.

Para além das aplicações de interface gráfico para manuseio da informação, existem bibliotecas para integração em aplicações que permitem extrair de um mapa SIG, todas essas informações. Uma dessas bibliotecas é o GDAL (GDAL, 2012), *Geospatial Data Abstraction Library*, para uso gratuito licenciado pela OSGeo (*Open Source GeoSpatial Foundation*), e que serve no caso em estudo para adquirir as informações de posição e relacionamento, de um mapa vetorial do Instituto Superior de Engenharia (ISE) da Universidade do Algarve (UALG).

A Fig. 2.4.1 mostra à esquerda o mapa de um Sistema de Informação Geográfica do R/C de uma parte do ISE e à direita a respetiva tabela de dados com os atributos da região selecionada a amarelo.



**Figura 2.4.1** – Mapa de um Sistema de Informação Geográfica (à esquerda) e respetiva tabela de dados com os atributos da região selecionada a amarelo (à direita).

Até aqui foram abordados os conceitos chave para o desenvolvimento de um sistema de localização e navegação tendo por base reconhecimento de objetos e um Sistema de Informação Geográfica. De seguida irão ser expostos algumas dos trabalhos mais recentes, que serviram de inspiração a esta dissertação.

## 2.5 Sistemas de Navegação para Invisuais

A locomoção de pessoas com dificuldades visuais é difícil, estando dependentes do uso de uma bengala, muitas vezes complementada pelo apoio de cães guia, para o desempenho das suas tarefas diárias. A navegação é efetuada através de mapas mentais construídos através das várias passagens que são efetuadas num local. No entanto quando um cego se encontra num local desconhecido ou pretende se dirigir para tal é necessário pedir indicações a pessoas para o orientar.

Esta limitação estimulou o uso de outros sentidos, tais como o tato e audição, que serviram como base para o desenvolvimento de modelos de apoio como são o caso de painéis tácteis ou instruções/avisos sonoros como é exemplo a tecnologia RIAS - *Remote Infrared Audible Signs* (Crandall *et al.*, 2001), que utiliza um sistema de mensagens audíveis colocadas em pontos estratégicos, como elevadores, paragens de transportes públicos, indicadores de direção, aproximação de passadeiras, semáforos entre muitos outros. As mensagens são gravadas num emissor e transmitidas através de infravermelhos para os recetores dos utilizadores nas proximidades, que por sua vez transformam a mensagem numa informação audível. A tecnologia RIAS proliferou nos

últimos anos nos Estados Unidos e Japão, fundamentalmente devido a ser de extrema facilidade de utilização e ser apoiada pelas próprias agências governamentais. No entanto é uma tecnologia cara, devido à necessidade de instalação e manutenção dos equipamentos emissores.

Dhristi (Ran *et al.*, 2004) é um sistema RTLS - *Real Time Locating System*, de navegação para ambientes mistos *indoor/outdoor*, constituído por um pequeno computador com um módulo GPS, módulo *WiFi*, auscultadores e um recetor de ultrassons. A navegação exterior é efetuada através do GPS, enquanto a navegação *indoor* é auxiliada através de identificadores de ultrassom, instalados em objetos ou em posições estratégicas que fornecem informações de posicionamento e identificação, lidas através do leitor de ultrassons transportado pelo indivíduo, e conjugadas com um SIG da área *indoor* de navegação. A determinação da rota é otimizada em tempo real tendo em conta as condições de tráfego, e de obstrução temporárias das vias de navegação que se verificam no momento. Todo o trabalho de localização e gerenciamento de pedidos é desempenhado através de um servidor remoto acessível através das redes *wireless* instaladas num *campus*. Este modelo servidor/cliente permite que o *hardware* do cliente seja o mais básico possível, ficando só encarregue das funções mais simples como obtenção de sinais dos sensores e transmissão das informações de navegação. A utilização de um modelo servidor/cliente permite ainda centralizar as alterações de percursos, devido a obstruções de via, ou tráfego acentuado num dado ponto. A interação entre o utilizador e o sistema é efetuada através de um módulo de reconhecimento de voz, permitindo que este forneça comandos de voz diretamente ao sistema. Do mesmo modo, as dicas de navegação são fornecidas através de um sintetizador de voz.

Outro dos sistemas mistos implementados de grande potencialidade é o NAVIO - *Pedestrian Navigation Systems in Combined Indoor/Outdoor Environments*, apresentado por Retscher e Thienelt, 2004. Este sistema apresenta modelos combinados para efetuar a localização, incluindo módulos GPS e GSM essencialmente para posicionamento *outdoor*, e um módulo *WiFi* para determinar a posição num ambiente *indoor*, incorpora ainda uma bússola digital e um acelerómetro, para obter os dados de velocidade de deslocamento e orientação.

Tjan *et al.*, 2005, apresenta por sua vez, um sistema para navegação de baixo custo, baseado em *tags* que são caracterizadas por padrões especiais de fácil identificação

através de uma câmara de imagem. As *tags* são construídas tendo por base material refletor, cuja função é a de refletir a luz emitida por um anel de LEDs infravermelhos presente à volta do obturador da câmara. A luz refletida, por sua vez, é detetada através de um outro anel de fototransístores de infravermelhos, identificando a presença de uma *tag*. Quando é detetada a presença de uma *tag*, uma câmara a preto e branco de sensibilidade próxima dos infravermelhos, captura a imagem e procede à descodificação.

Trabalho semelhante foi desenvolvido por Cougland e Manduchi, 2009, tendo em vista a aplicabilidade em telemóveis. Segundo os autores o método permite uma rápida deteção e descodificação bem como apresenta elevada robustez ao ruído. Este utiliza um método auxiliar, em que coloca um marcador colorido com as três cores RGB, uniformemente distribuídas num círculo, próximo do código de barras ou texto a descodificar. Este marcador é de fácil e rápida deteção mesmo a alguns metros e permite ainda calibrar o sistema quanto às cores RGB influenciadas pela iluminação ou sombras.

Ainda dentro dos sistemas que utilizam indentificadores gráficos ou textuais como base para localização podemos referenciar o trabalho desenvolvido por Foong e Razali, 2011, que propõe um método simples de navegação apoiado unicamente em reconhecimento de caracteres, utilizando segmentação de caracteres através do método de Otsu, 1975, para a leitura de sinalizadores que fornecem informações quanto à localização e avisos de perigo, através de um sintetizador de voz.

Outros autores deram preferência à utilização de dispositivos eletrónicos de transmissão e receção sem fios como complemento à localização. Exemplo disso é o trabalho desenvolvido por Wilis e Helal, 2005, que apresenta uma solução baseada em identificadores RFID, para o fornecimento das informações necessárias para a localização e navegação. Por sua vez Kim e Song, 2007, utiliza uma tecnologia mista em que incorporou numa bengala, uma pequena câmara, um leitor de RFID e um módulo sem fios de rede *ZigBee*. No mesmo sentido, procurando utilizar toda a potencialidade da cada vez mais comum infraestrutura de redes *wireless*, presente em maior parte dos empreendimentos públicos, Feng e Liu 2012, criaram um sistema de localização baseado na triangulação das células wireless (*Access Points*) através das medições das potências do sinal recebido, combinado com um SIG para o enquadramento geográfico e um sintetizador de voz para uma comunicação

bidirecional. Balakrishnan *et al.*, 2005, desenvolveu uma solução de navegação, fazendo recurso à visão *stereo* e à disparidade medida, notificando através de tons musicais os objetos quanto à sua posição, distância e forma. A solução é composta por um capacete equipado com um pequeno computador, uma câmara *stereo* e auscultadores.

Tendo em vista a inequívoca necessidade e aplicabilidade dos sistemas de localização em tempo real para o quotidiano das populações, alguns projetos tem como objetivo alargar a sua utilização para meios urbanos e citadinos. Uma dessas iniciativas é o projeto e-Adept (*Electronic Assistance for Disabled Elderly Pedestrians and Travelers*) (e-Adept, 2009). Este projeto combina, numa arquitetura Cliente-Servidor, diversas tecnologias e técnicas avançadas de posicionamento, auxiliadas por um conjunto de mapas digitais para auxílio da navegação. O servidor incorpora 3 módulos, um módulo de mapas digitais, um módulo para determinação de rotas de navegação e um módulo para interação direta com o utilizador. É ainda no servidor que são armazenadas todas as informações e mapas para a navegação. Por sua vez o utilizador transporta um *SmartPhone* com um módulo GPS e um módulo de inércia, com um acelerómetro, para estimar a distância percorrida e uma bússola digital para obtenção dos dados de direção, para as zonas de sombra do GPS, muito frequentes em zonas de grande densidade populacional e com prédio altos. Para a orientação *indoor*, não sendo possível utilizar o GPS, é utilizada uma técnica mista de *Bluetooth* e *tags* RFID. As informações são relatadas através de um módulo de voz.

Por último, podemos ainda referir o projeto *SmartVision* (du Buf *et al.*, 2011). Este projeto concebido em Portugal pretende desenvolver um sistema portátil de baixo custo para assistir pessoas com deficiência visual na sua navegação diária tanto em ambientes interiores como exteriores. Este sistema prevê a utilização de tecnologias mistas tais como o GPS, as redes *WiFi*, identificadores RFID e visão artificial auxiliadas por um Sistema de Informação Geográfico para o planeamento da rota de navegação e localização através de pontos de referência detetados pelos sensores. Além do auxílio na navegação o sistema identifica e localiza obstáculos comunicando a sua existência e posição ao utilizador permitindo a este evitar situações de potencial risco. O interface do utilizador com o sistema é realizado através de 4 botões instalados numa bengala, sendo atribuído funções específicas a cada um. No sentido inverso, do sistema para o utilizador, é utilizado um módulo de vibração instalado na bengala para instruções simples tais como, vire à esquerda ou à direita e um sintetizador de voz para fornecer

informações mais detalhadas. Este projeto serviu de base ao projeto *Blavigator* (José *et al.*, 2012), no qual esta dissertação se enquadra.

Neste capítulo foram apresentados alguns dos conceitos base que envolvem os processos de visão para reconhecimento de objetos e seletividade de informação mais recentes e analisados sucintamente alguns dos algoritmos mais populares para navegação, aplicados principalmente ao ramo da robótica móvel. Foram ainda apresentados alguns dos princípios intrínsecos a um Sistema de Informação Geográfica, de forma a compreender a sua filosofia e analisados os mais recentes esforços desenvolvidos na área dos sistemas de navegação para invisuais, alguns dos quais inspiradores desta dissertação. No capítulo seguinte é proposto analisar-se detalhadamente a arquitetura construída no desenvolvimento da aplicação para orientação e mobilidade por visão computacional.

# Capítulo III

## 3. Sistema de Visão para Orientação e Mobilidade

O sistema proposto pretende auxiliar pessoas com dificuldades visuais a movimentarem-se em ambientes fechados como centros comerciais, escolas, hospitais e demais infraestruturas públicas, baseado em referências visuais extraídas por ferramentas e técnicas de visão computacional, a partir de imagens capturadas por uma câmara. Estas mesmas referências permitem posteriormente efetuar a localização do indivíduo, assistida por um Sistema de Informação Geográfica que contém os dados das coordenadas das referências visuais e as relações de vizinhança entre regiões (divisões) e os objetos tidos como pontos para referência. Tratando-se o público-alvo, a população com dificuldades visuais, é usado um sistema de interface apoiado em fala artificial, que utiliza informações áudio para advertir sobre os pontos de referência detetados e prestar auxílio na navegação.

### 3.1 Arquitetura do Sistema

O sistema é desenvolvido com o propósito de ser executado a partir de um computador portátil de pequenas dimensões e baixos recursos, utilizando unicamente como equipamentos periféricos uma câmara *Web* e umas colunas de áudio. A aplicação responsável por todos os processos foi implementada em linguagem de programação C, e foi compilada para correr na plataforma *Windows*. No entanto é completamente compatível a sua integração em plataformas de uso livre e gratuito como o *Linux*, tornando o sistema significativamente mais barato.

Na Fig. 3.1.1 está representada a arquitetura do sistema e os seus argumentos de entrada e saída. O sistema é composto por: (a) **Módulo Principal** – sistema de localização em tempo real, responsável por todo o processo de localização e por um (b) **Módulo de Treino**, desenhado para a criação de bibliotecas de objetos, constituídas pelos pontos-chave extraídos de um conjunto de imagens de várias posições de um objeto. Posteriormente essas bibliotecas irão auxiliar o reconhecimento dos objetos, através da utilização do *SURF* (ver Secção 3.2), nas imagens capturadas pela câmara. O



módulo principal é composto por um conjunto de processos sequenciais (sub-módulos), os quais segue abaixo uma descrição sucinta.

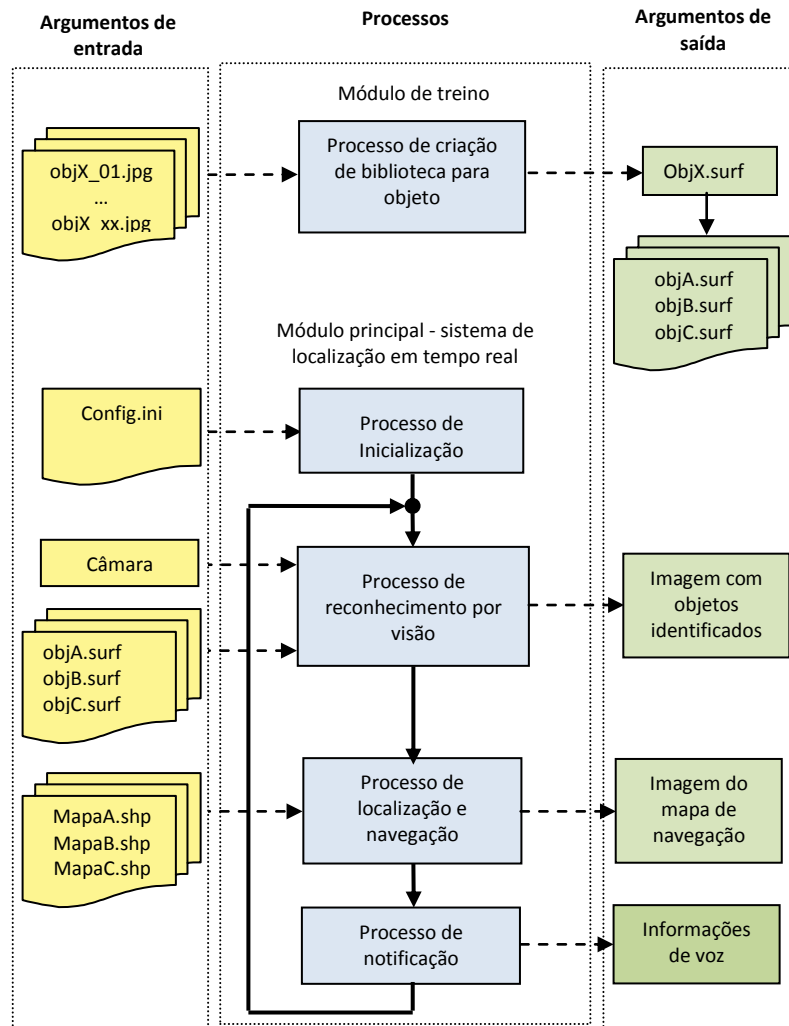


Figura 3.1.1 – Arquitetura geral do sistema de mobilidade e orientação.

No **processo de inicialização** são carregadas as configurações referentes à parametrização do sistema, à definição das livrarias que armazenam os modelos de cada um dos objetos para reconhecimento, através da biblioteca *OpenSurf* (Evans, 2009) e a descrição dos ficheiros dos mapas que contêm as informações geográficas a serem extraídas e manipuladas através da biblioteca *GDAL* (GDAL, 2012). São ainda definidas quais as aplicações auxiliares usadas para o reconhecimento de caracteres, *Tesseract-OCR* (Tesseract-OCR, 2007), e para o módulo de *text-to-speech* responsável por transformar as informações de texto em informações audíveis ao utilizador, através de um *script* (Bennett, 2010) que utiliza o reconhecimento de voz nativo do *Windows*.

O **processo de reconhecimento por visão** é responsável por efetuar o reconhecimento e extração das referências visuais adquiridas através da câmara, bem como efetuar a sua contextualização no meio, consoante as similaridades entre *frames* capturados, atribuindo identificadores únicos que distinguem se as referências visuais pertencem ou não a imagens similares, invalidando que o mesmo objeto (na mesma posição), seja detetado várias vezes em *frames* consecutivos de captura.

O **processo de localização e navegação** tem a seu cargo a tarefa de estimar a posição do indivíduo através das referências adquiridas e determinar o caminho mais curto entre a posição atual e o destino pretendido pelo utilizador.

O **processo de notificação** realiza a interação entre o sistema e o utilizador através de comandos de voz, relatando quaisquer objetos identificados e as informações de localização e coordenadas de movimento necessárias para o utilizador atingir o seu destino.

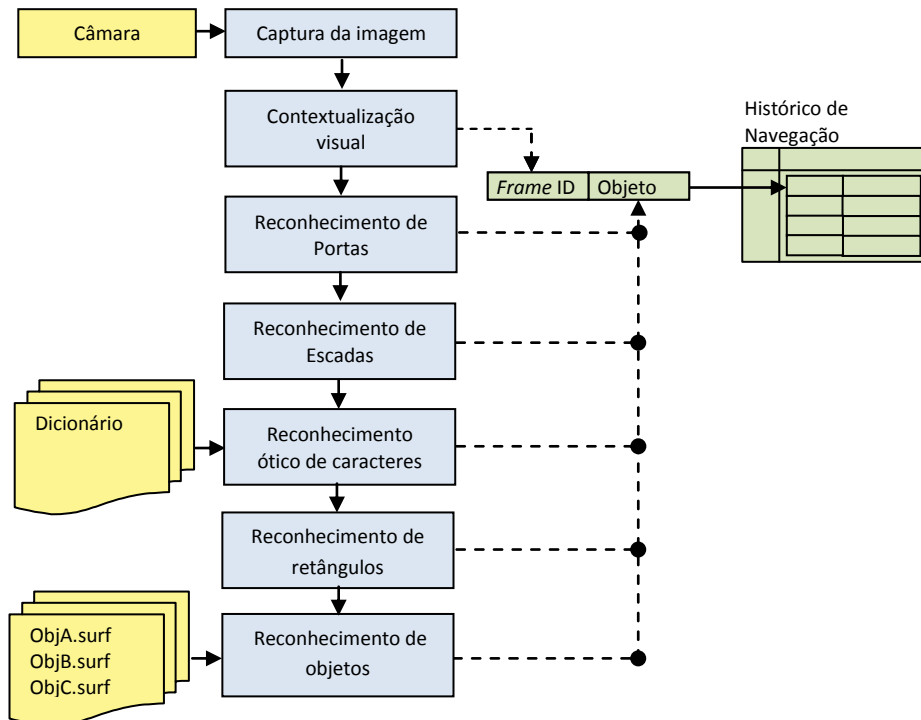
De seguida vamos analisar mais detalhadamente os dois processos principais, o de reconhecimento por visão e o de localização e navegação. O processo de notificação por voz encontra-se fora dos objetivos desta dissertação, apenas para prova de conceito foi utilizado o sintetizador nativo de voz do *Windows*.

### 3.2 Processo de Reconhecimento por Visão

Como referido anteriormente, o modelo de aquisição de dados é baseado na visão, fazendo recurso a uma câmara *Web*, com o objetivo de extrair objetos de referência, tais como portas, janelas, indicadores textuais, cacifos, sinalização, extintores de incêndio, entre outros com o intuito de auxiliar a navegação no meio *indoor*. Os objetos são contextualizados através de um identificador de *frame* e são armazenadas as informações referentes ao tipo de objeto e coordenadas de posição (relativamente à imagem), num *buffer* do tipo FIFO (*First In First Out*), para posteriormente ser efetuada a estimação da localização com base no histórico de objetos reconhecidos.

Na Fig. 3.2.1, apresenta-se o diagrama de blocos das diversas etapas constituintes do processo de reconhecimento por visão. Numa primeira fase é realizada a (a) captura da imagem através da câmara, seguido de (b) um processo que visa contextualizar a imagem capturada com a anterior verificando se as imagens são idênticas. Após a verificação estar concluída são executadas as etapas responsáveis pelo reconhecimento

de (c) portas, (d) escadas, (e) caracteres (OCR), (f) retângulos e (g) objetos utilizando o SURF e as bibliotecas construídas.



**Figura 3.2.1** – Diagrama de blocos do processo de visão.

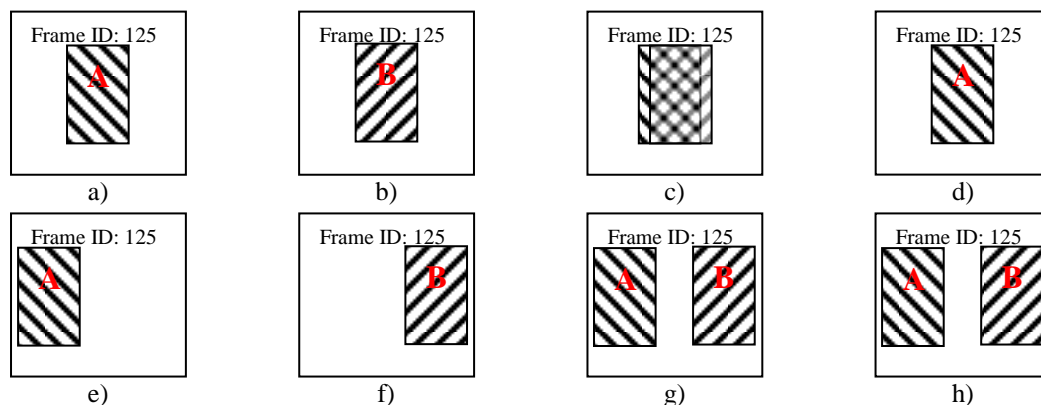
O passo referente à contextualização visual é fundamental, devido ao ruído produzido pela iluminação artificial, reflexos, e próprias limitações provocadas por alguns dos algoritmos, é impossível capturar toda a informação disponível no cenário visual numa única captura (*frame*) sendo necessário usualmente várias capturas para detetar os vários objetos ou mesmo esperar um ligeiro movimento do utilizador (câmara) para detetar um objeto. As informações adquiridas, apesar de capturadas em instantes diferentes podem ser relacionáveis, visto pertencerem a um mesmo cenário visual. Do mesmo modo, um mesmo objeto, pode ser sucessivamente detetado numa sequência de *frames*, podendo resultar em interpretações incorretas se erradamente for assumido que os objetos capturados são objetos diferentes, quando na realidade são o mesmo, criando a necessidade de contextualizar a informação obtida com o cenário visual a que estes pertencem. Ou seja, é necessário verificar se a informação retirada do *frame* de imagem atual pertence a uma imagem semelhante ou distinta, da imagem cuja as últimas informações foram retiradas. A técnica de contextualização permite ainda criar um modelo mais completo do cenário, possibilitando interligar várias informações detetadas em sequências de *frames* sucessivas, como se tratasse de uma deteção num

único *frame* e eliminar a repetição de objetos já detetados. Antes de adicionados ao histórico de navegação os objetos reconhecidos são comparados, com os presentes no *buffer*, quanto ao identificador da imagem do qual foram extraídos, tipo de objeto e área em que se localizam dentro da imagem, de forma a poder descartar objetos já anteriormente reconhecidos na mesma imagem e posição.

### 3.2.1 Contextualização Visual

A contextualização visual, é um processo que pretende verificar se uma informação obtida do meio está relacionada com informação obtida anteriormente, ou seja se esta pertence à mesma imagem, ou cenário visual.

Numa primeira fase é necessário testar a semelhança entre as imagens dos *frames* e aferir a probabilidade de serem parte do mesmo cenário visual. Consoante o resultado será atribuído ao *frame* analisado e respetivos objetos detetados um identificador igual ou diferente do *frame* referência (*frame* anteriormente capturado). Através desta técnica tornasse possível comparar objetos identificados em instantes diferentes de captura mas pertencentes a *frames* semelhantes, permitindo descartá-los ou relacioná-los como objetos contíguos, avaliando a área que ocupam na imagem. Essa avaliação é efetuada comparando se o centro de algum dos novos objetos é coincidente com um objeto posteriormente detetado, que possua idêntico identificador, caso haja sobreposição e seja idêntico a um objeto já reconhecido é descartado (Fig. 3.2.1.1, 1ª linha), caso contrário é adicionado ao histórico como sendo um novo objeto (Fig. 3.2.1.1 2ª linha). A Fig. 3.2.1.1 mostra em a) no primeiro *frame*, a identificação de um objeto (A) e em b) no segundo *frame*, a identificação de um objeto (B) idêntico a (A). Através da c) sobreposição das áreas que compreendem os dois objetos é d) verificada a existência de uma coincidência e o objeto detetado no segundo *frame* (B) é descartado. Na segunda linha, no e) primeiro *frame* foi identificado um objeto (A) e no f) segundo *frame* foi identificado um segundo objeto (B) idêntico ao já identificado (A). Analisando as áreas g) dos dois objetos em h) verifica-se que não existe coincidência e ambos os objetos são aceites como novas deteções.



**Figura 3.2.1.1** – Processo de avaliação para aceitação da existência de um novo objeto, em dois *frames* similares, com sobreposição em a), b), c) e d) e sem sobreposição em e), f), g) e h).

Para efetuar a contextualização e enquadramento dos objetos é necessário criar mecanismos que permitam testar a semelhança de uma sequência de *frames*, de forma a verificar se existe ou não transição para um novo cenário visual e consequentemente atribuir identificadores e decidir a validade da nova detecção.

Existem diversas técnicas que pretendem testar a correlação entre imagens para se verificar a sua semelhança, entre as quais a subtração de imagens, correlação de histograma de cores (Gargi *et al.*, 2002), transformadas de *Hilbert* (Priya e Dominc, 2012), análise de fluxo de movimento (Kruklikovská e Polec, 2012), correlação de pontos-chave através de algoritmos como o *SIFT* ou o *SURF* (Zheng *et al.*, 2005; Huang *et al.*, 2008), entre outras técnicas que têm sido desenvolvidas nos últimos anos.

No caso em estudo é tido em consideração que a técnica para análise de similaridade entre *frames* terá que ser relativamente imune às oscilações da câmara, provocadas pelo movimento do utilizador e ao ruído ambiente provocado pelo comportamento oscilatório da iluminação. Devido ao seu baixo custo de processamento, facilidade de implementação e sua relativa robustez às oscilações da câmara e iluminação, a técnica de correlação de histograma de cores é a mais indicada para ser utilizada no sistema, apesar de possuir a desvantagem de poder haver situações em que dois *frames* de imagem, aparentemente distintos, possam ter histogramas de cores muito semelhantes e consequentemente resultar em falhas de detecção. Contudo a sua ocorrência tem uma probabilidade muito reduzida.

O primeiro passo do método proposto é criar o histograma de cor do *frame* de imagem em níveis de cinzento e efetuar a subtração desse histograma, para cada um desses níveis, com o histograma da imagem anterior, criando assim um novo histograma

que armazena as disparidades de cor entre os *frames*. Através da relação da soma das diferenças de cor entre os histogramas e o número total de pixéis da imagem é determinada a razão de correlação ( $rc$ ) entre as imagens, sendo que os seus valores tomarão valores entre 0 e 1, em que 0 significa que as imagens são estritamente idênticas e 1 que as imagens são totalmente diferentes. Considerando,  $N_c$  o número de cores,  $A$  a altura e  $L$  a largura do *frame* e  $H(k)$  o histograma de cada um dos *frames* ( $f$ ) consecutivos, temos que a razão de correlação  $rc$  é dada por:

$$rc = \frac{1}{A \times L} \times \sum_{k=0}^{N_c} \sum_{k=0}^{N_c} |H_f(k) - H_{f-1}(k)|. \quad (3.2.1.1)$$

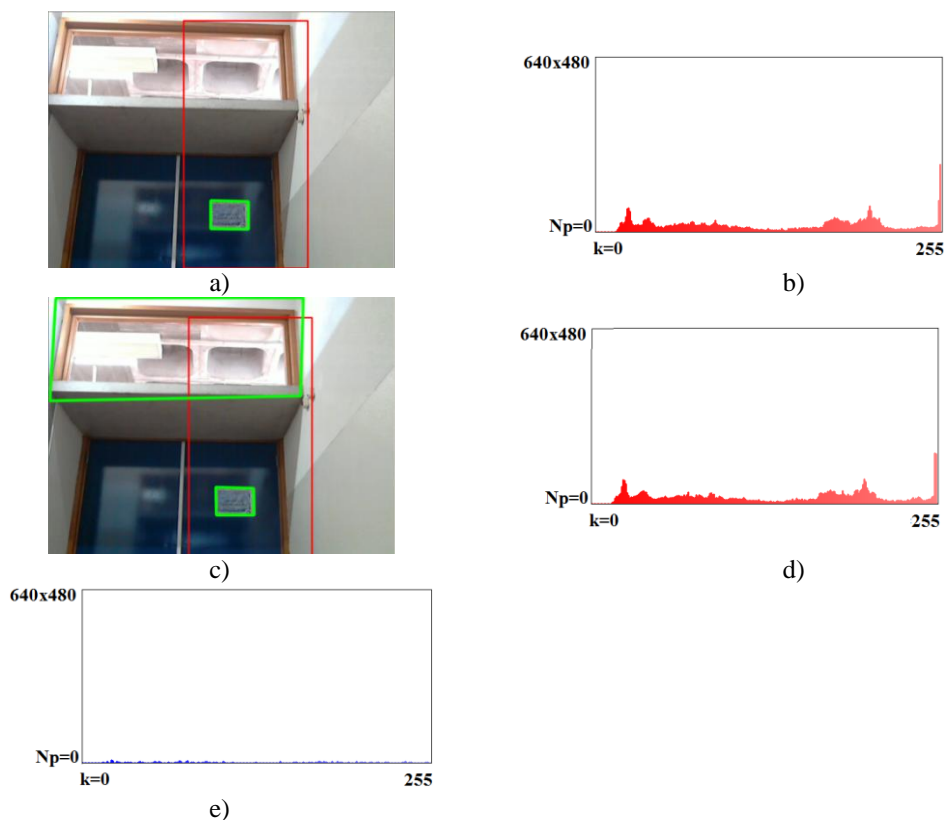
Através da razão de correlação, é possível efetuar a indexação do *frame* de imagem, consoante esse valor esteja acima ou abaixo de um valor de *threshold* ( $T_{rc}$ ) predeterminado (valor apresentado nos testes é de  $T_{rc} = 0.6$ ). Só é atribuído ao *frame* de imagem e objetos reconhecidos no mesmo, um identificador diferente do anterior caso o seu valor ultrapasse ( $rc > T_{rc}$ ) o limite estabelecido para *threshold*.

Devido ao ruído ambiente provocado pela iluminação, os pixéis da imagem sofrem pequenas oscilações na sua cor, resultando em diferenças que podem ser significativas quando somadas, o que pode resultar numa indexação errónea, atribuindo identificadores distintos a imagens semelhantes. De forma a minimizar o efeito do ruído ambiente é proposto que as imagens sejam redimensionadas para metade do seu tamanho de forma a diluir essas pequenas diferenças, antes de construir o histograma de cor. Podem ainda ser utilizados outros métodos alternativos, como a diminuição do número de cores da imagem, ou mesmo equacionar a utilização de filtros de ruído apesar deste último método ter um custo de processamento ligeiramente mais elevado, que as mudanças de escala ou de número de cores.

Os objetos detetados serão identificados com o identificador do *frame* no qual foram reconhecidos e com as coordenadas da área que ocupam na imagem. Sendo assim possível em deteções posteriores do mesmo objeto determinar se este se trata de um novo objeto ou do mesmo.

No exemplo ilustrado na Fig. 3.2.1.2, foram capturadas duas imagens em sequência. Na primeira imagem são detetados uma porta e um pequeno retângulo no centro da mesma (Fig. 3.2.1.2a)), enquanto na segunda imagem, além de uma porta e um pequeno retângulo é identificado um segundo retângulo na parte superior da imagem (Fig. 3.2.1.2c)). Pela análise dos histogramas de cor de ambas as imagens (Fig. 3.2.1.2e))

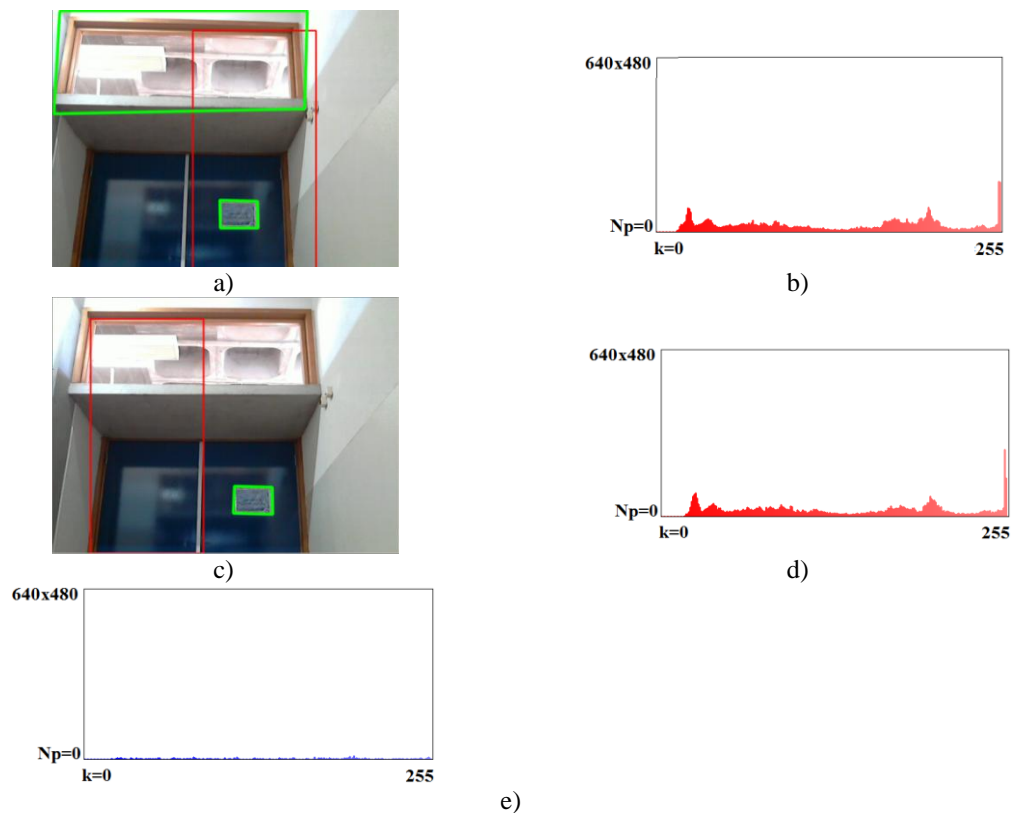
resultante da subtração dos histogramas do primeiro (Fig. 3.2.1.2b)) e segundo (Fig. 3.2.1.2d)) *frames*, verifica-se que são muito idênticos, pelo que o seu identificador deverá ser o mesmo indicando que os objetos são todos eles parte de um mesmo cenário visual. A porta e o pequeno retângulo detetado na Fig. 3.1.1.2c), não são validados como novos objetos, visto que ambos ocupam as mesmas posições relativamente ao *frame* anterior, Fig. 3.1.1.2a). Este fator é indicador de uma duplicação da informação anteriormente obtida. Deste modo, após a sequência das duas imagens, só será identificado uma porta, o pequeno retângulo na zona central e o retângulo superior.



**Figura 3.2.1.2** – Processo de verificação de similaridade entre dois *frames*. a) Primeiro *frame* e b) respetivo histograma de cores. c) Segunda imagem da captura e o d) histograma de cores respeitante. Em e) está o resultado da diferença entre os dois histogramas, do qual é possível concluir que se trata de imagens similares.

No exemplo da Fig. 3.2.1.3, também se pode verificar a semelhança das imagens através da análise do histograma de cor. No entanto, neste caso de análise foi detetado uma segunda porta na Fig. 3.2.1.3c) que tem uma localização não coincidente com a porta detetada na Fig. 3.2.1.3a), fator que indica que apesar de pertencerem ao mesmo cenário visual, são portas diferentes. Esta é validada como uma nova deteção. Por sua

vez o pequeno retângulo reconhecido na Fig. 3.2.1.3c) é descartado, pois já foi detetado previamente na Fig. 3.2.1.3a).



**Figura 3.2.1.3** – Processo de verificação de similaridade entre dois *frames*. a) Primeira imagem capturada e b) respetivo histograma de cores. c) Segunda imagem da captura e o d) histograma de cores respeitante. Em e) está o resultado da diferença entre os dois histogramas, do qual é possível concluir que se trata de imagens similares.

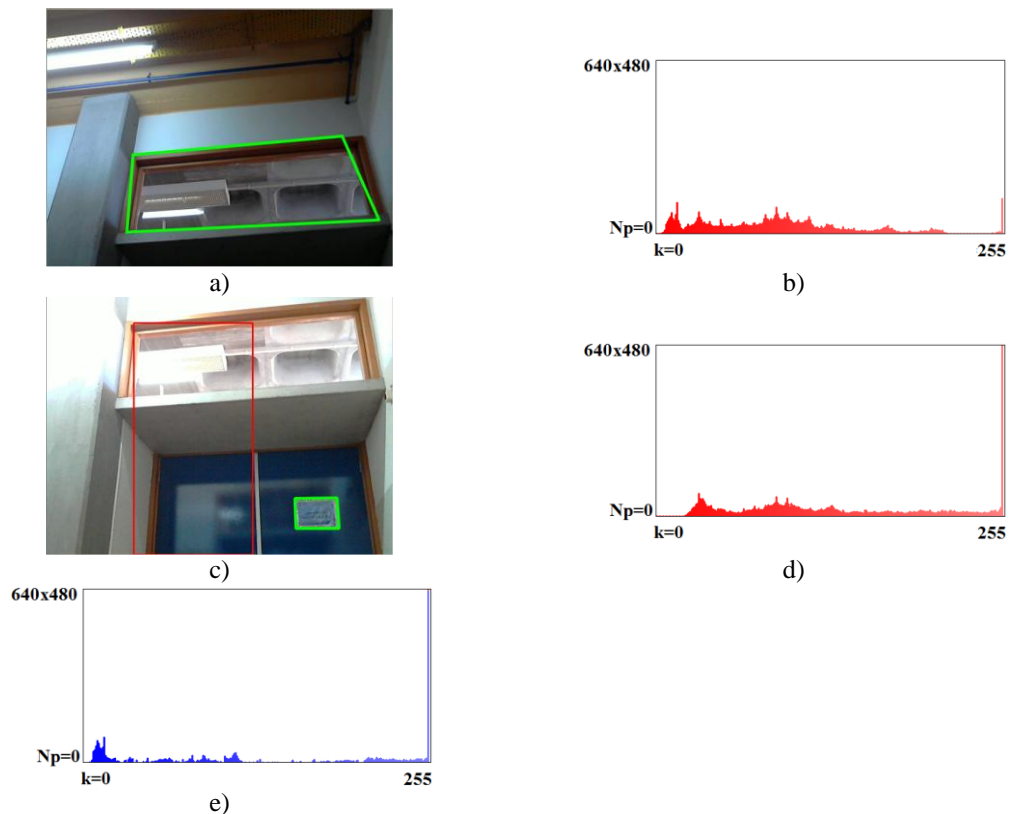
No exemplo, apresentado na Fig. 3.2.1.4, é verificado que as imagens pertencem a ambientes totalmente diferentes, pelo que os objetos detetados são indexados como fazendo parte de deteções em separado e conseqüentemente nenhuma operação de análise de coincidência de localização é executada e todos os objetos são validados como novas deteções.

Através da técnica de contextualização utilizada, é possível efetuar a indexação das imagens e dos objetos detetados, bem como aferir a sua validade através da verificação de sobreposição com objetos do mesmo tipo e com o mesmo identificador.

Na secção seguinte serão descritos os métodos utilizados para o reconhecimento de portas nas imagens capturadas. As portas são um dos objetos mais comuns de encontrar em ambientes interiores (*indoor*), o que as torna pontos fracos para referência, contudo



são imprescindíveis para determinar os pontos de passagem para outras divisões essenciais na navegação.



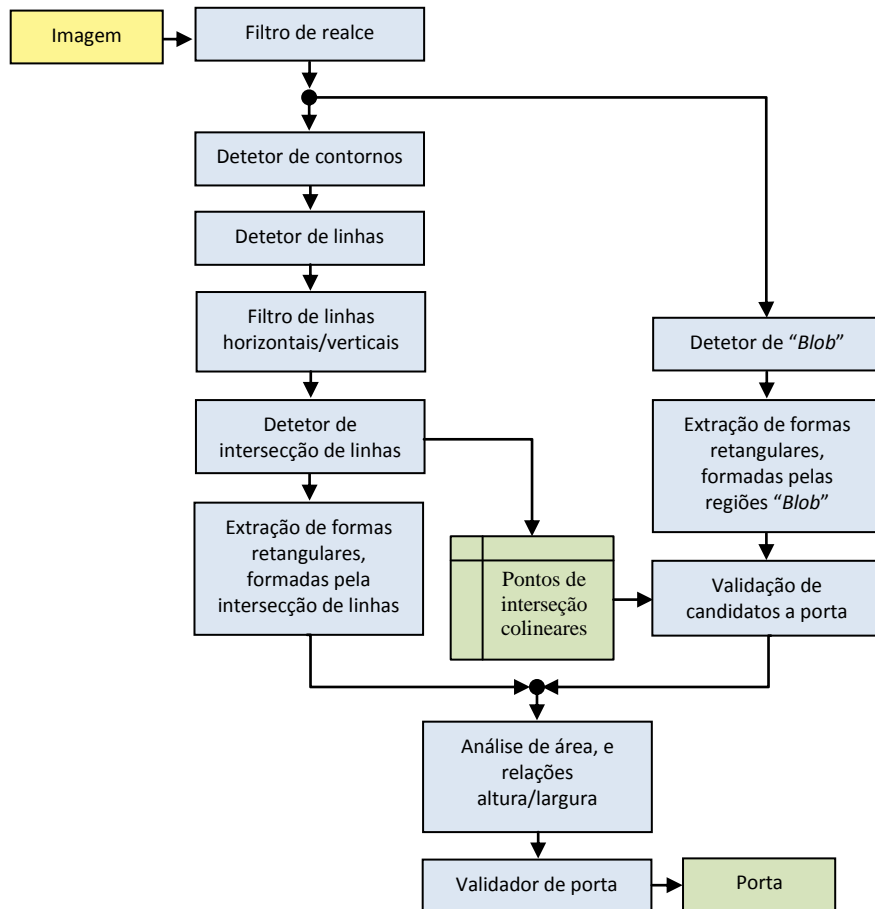
**Figura 3.2.1.4** – Processo de verificação de similaridade entre dois *frames*. a) Primeira imagem capturada e b) e respetivo histograma de cores. c) Segunda imagem da captura e o d) histograma de cores respeitante. Em e) é possível verificar que os *frames* são parte de cenários visuais diferentes, através da diferença entre os dois histogramas.

### 3.2.2 Reconhecimento de Portas

Do ponto de vista da visão as portas quando vistas de frente (ou quase de frente) são genericamente definidas por formas retangulares numa imagem, pelo que o foco do algoritmo de deteção será procurar possíveis retângulos na imagem que possam pertencer a portas.

A Fig. 3.2.2.1 representa o diagrama de blocos para o reconhecimento de portas. O algoritmo é composto por dois métodos, aqui representados em paralelo, para reconhecimento de portas: (1) um baseado em agrupamentos de linhas que pelas suas intersecções descrevam formas retangulares e (2) outro baseado em regiões criadas a partir de uma segmentação binária cujos extremos sejam próximos com intersecções entre linhas. Esta abordagem mista para deteção permite aumentar a eficácia de reconhecimento, possibilitando detetar uma porta através de uma vista parcial da

mesma, ou em condições em que a sua distinção é dificultada pelas suas semelhanças com a parede de fundo.



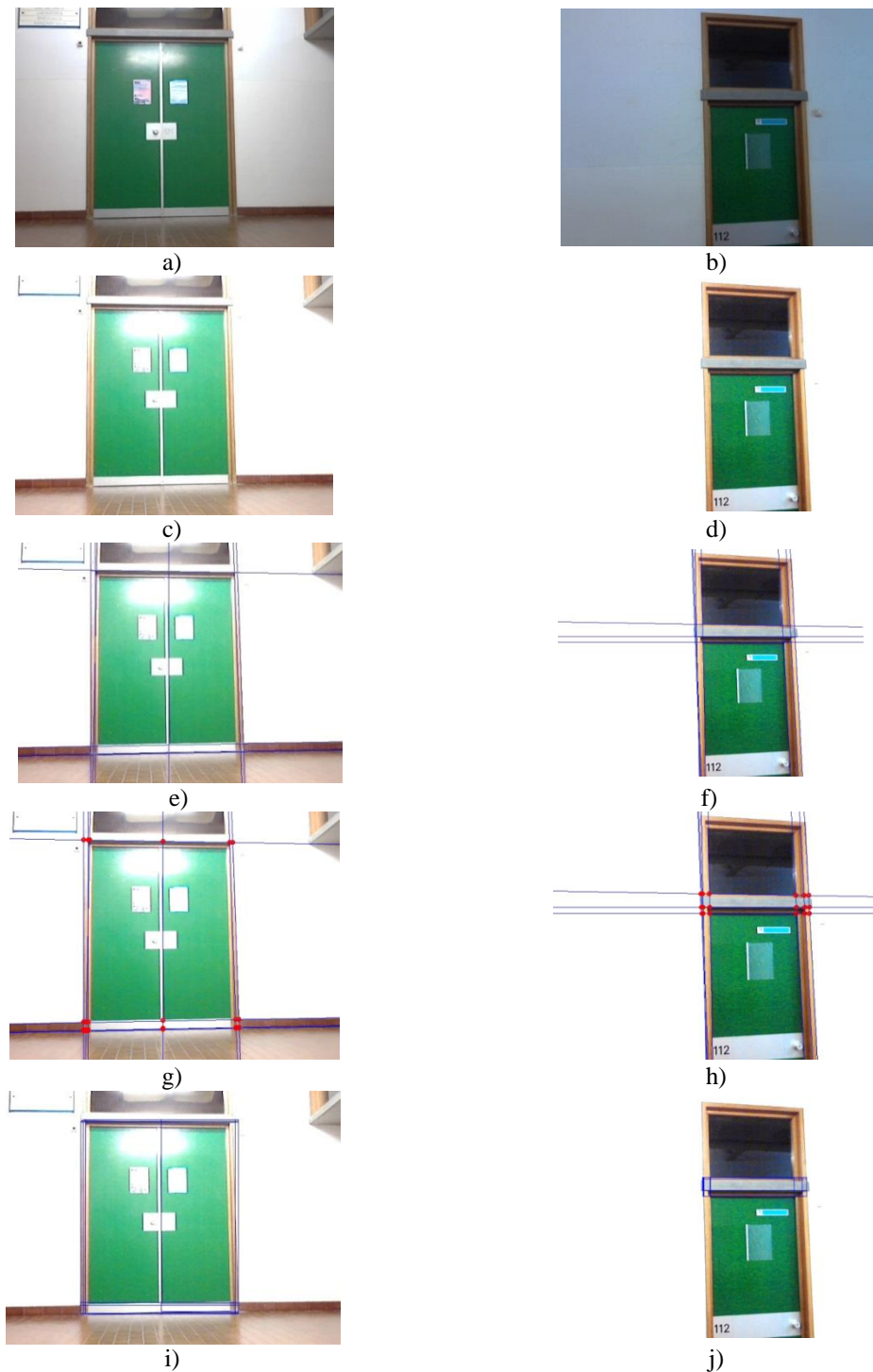
**Figura 3.2.2.1** – Diagrama de blocos para a detecção de porta.

Ambos os métodos utilizam a imagem capturada (a) após pré-tratamento através de um filtro de realce (Russ, 2011b) tendo em vista a obtenção de uma maior definição da imagem. O primeiro método consiste na (b1) aplicação de um detetor de arestas utilizando o Canny (Canny, 1986) e (c1) um detetor de linhas como o Hough (Hough, 1962). Extraídas as linhas é (d1) realizada uma filtragem que tem por objetivo seleccionar unicamente as linhas horizontais e verticais da imagem, sendo de seguida (e1) efetuada uma verificação pelas intersecções entre as linhas. Finalmente, através dos pontos de intersecção é (f1) efetuada uma procura por formas retangulares formadas pela intersecção entre linhas horizontais e verticais. O segundo método (b2) procura regiões em mancha (“Blob”) e executa a procura de (c2) formas que sejam semelhantes a retângulos. Após a extração (d2) são validados como candidatos a portas, aqueles que apresentarem pontos de intersecção colineares (pertencentes a uma mesma reta

horizontal), na proximidade dos seus vértices superiores. Extraídos os candidatos pelos métodos descritos (g) é realizada uma análise quanto à área e às relações altura e comprimento de forma a descartar os candidatos com pouca probabilidade de serem pertencentes a portas. Vejamos mais detalhadamente cada um dos passos referidos em ambos os métodos utilizados.

O primeiro método visa encontrar relações entre linhas que pelas suas interseções possam pertencer a portas na imagem. A informação referente a arestas na imagem torna-se assim condição fundamental sendo o seu nível de detalhe essencial para a eficácia de detecção de linhas sendo aconselhável a aplicação de um filtro de realce, para destacar essa informação e torná-la mais definida. Consideremos então  $I(x, y)$  como sendo a imagem capturada (Fig. 3.2.2.2a) e 3.2.2.2b)) e  $I_{UM}(x, y)$  a imagem realçada (Fig. 3.2.2.2c) e 3.2.2.2d)) por um filtro do tipo *Unsharp Mask* (Russ, 2011). De seguida é utilizado um detetor de arestas do tipo *Canny* (Canny, 1986) com um filtro gaussiano de tamanho  $\sigma = 1.0$  e valores de *threshold* inferior  $T_l = 0.3$  e superior  $T_h = 0.5$ , sendo a imagem resultante  $I_c(x, y)$  sujeita ao detetor de linhas de *Hough* (Hough, 1962) para a extração das linhas da imagem. As linhas são extraídas e armazenadas em  $I_h(\rho, \theta)$  e sujeitas a uma filtragem quanto à sua orientação em linhas horizontais e verticais da imagem. O filtro utiliza a diferença, considerando um sistema em coordenadas polares, entre o ângulo referência ( $90^\circ$  para os segmentos de reta horizontais e  $0^\circ$  para os segmentos de reta verticais) para cada uma das orientações e o ângulo individual das retas. As retas são validadas se estiverem abaixo de um determinado limiar  $T_\theta$  como horizontais se  $|90 - I_h(\rho, \theta)| < T_\theta$ , ou verticais se  $|0 - I_h(\rho, \theta)| < T_\theta$ , resultando num novo histograma de linhas  $I_{fh}(\rho, \theta)$ , unicamente com as linhas filtradas. Em ambos os casos foi utilizado um valor de limiar  $T_\theta = 20^\circ$ , valor o qual comprovado empiricamente como sendo o que conduzia a melhores resultados.

Valores inferiores para  $T_\theta$  baixam consideravelmente o número de retas extraídas e a eficácia de detecção de portas, enquanto valores superiores resultam num aumento superior de retas e como consequência um maior número de falsos positivos. Na Fig. 3.2.2.2 e) e f) encontra-se ilustrado os segmentos de reta verticais e horizontais que delimitam as portas após aplicação do filtro.



**Figura 3.2.2.2** – Imagem de uma a) porta dupla e uma b) porta simples. Imagens filtradas através de um filtro de realce do tipo *Unsharp Mask* c) e d). Detecção das linhas horizontais e verticais em e) porta dupla e f) simples, e os pontos de intersecção entre linhas em g) e h). Rectângulos candidatos a portas, formados pelas intersecções de linhas para uma i) porta dupla e j) uma porta simples.

Sendo o objetivo a deteção de formas retangulares que possam ser pertencentes a portas na imagem e extraídas as retas horizontais e verticais são analisados os conjuntos de linhas filtrados  $I_{fh}(\rho, \theta)$  que pelas suas intersecções descrevam formas retangulares.

Para determinar essas interseções e respectivos pontos de intersecção entre pares de linhas horizontais e verticais (dentro dos limites da imagem), é utilizado um sistema de equações que relaciona as equações de reta em coordenadas polares como o seguinte:

$$\begin{cases} \rho_1 = x\cos(\varphi_1) + y\sin(\varphi_1) \\ \rho_2 = x\cos(\varphi_2) + y\sin(\varphi_2). \end{cases} \quad (3.2.2.1)$$

Este sistema pode ser transformado num sistema matricial o que permite uma resolução rápida, do ponto de vista computacional, fazendo recurso ao teorema de *Cramer* para sistemas de equações lineares. Os pontos de intersecção das retas estão representados na Fig. 3.2.2.2 g) e h). Obtidos os pontos de intersecção entre as linhas, são agrupados os conjuntos que definam formas retangulares (Fig. 3.2.2.2i) e j)).

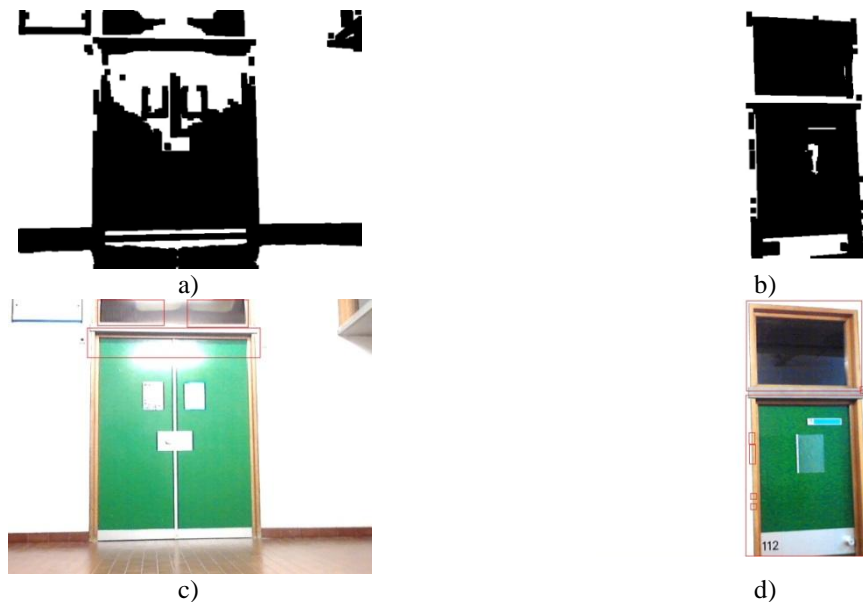
O segundo método verifica a existência de regiões cujos extremos superiores sejam limitados pela intersecção de linhas horizontais e verticais à esquerda e direita, em uma imagem binária  $I_b(x, y)$  resultante de uma operação de binarização baseada num valor de *threshold* ( $T_b$ ) calculado automaticamente a partir do histograma da imagem realçada  $I_{UM}(x, y)$ .

O método de segmentação binária automática utiliza o algoritmo de Ridler e Calvard, 1978, para calcular o centro de massa do histograma de cor  $h_c(n)$  obtido de  $I_{UM}(x, y)$  e com base no seu valor efetuar uma segmentação binária da imagem  $I_{UM}(x, y)$ . Este algoritmo utiliza um processo iterativo, que determina a cada iteração um novo valor ótimo para o valor de *threshold* a partir do valor médio calculado entre duas classes separadas pelo valor de *threshold* da iteração anterior. Considerando  $T_{b,i}$  o valor de *threshold* a ser calculado na iteração  $i$ ,  $T_{b,i-1}$  o valor de *threshold* calculado na iteração imediatamente anterior,  $\mu_{0_{-}tb,i-1}$  e  $\mu_{tb_{-}255,i-1}$  os valores médios do intervalo compreendido entre os níveis cinza 0 a  $T_{b,i-1}$  e de  $T_{b,i-1}$  a 255 do histograma  $h_c(n)$ , temos que:

$$T_{b,i} = \frac{\mu_{0_{-}tb,i-1} + \mu_{tb_{-}255,i-1}}{2}. \quad (3.2.2.2)$$

O processo termina quando a diferença entre os valores de *threshold* calculados na presente iteração ( $T_{b,i}$ ) e na iteração anterior ( $T_{b,i-1}$ ) for desprezável, isto é se  $|T_{b,i} - T_{b,i-1}| = 0$ . Para efeitos de inicialização de o algoritmo é considerado que o centro de massa inicial está no ponto médio da gama de cores cinza, isto é 127 (de uma gama entre 0 e 255).

Calculado o valor de *threshold* ( $T_b$ ) a utilizar é criada a imagem binarizada  $I_b(x, y)$  a partir da imagem  $I_{UM}(x, y)$  revelando o conjunto de regiões segmentadas. De seguida são retirados os contornos de cada uma das regiões de  $I_b(x, y)$  através de uma função nativa do *OpenCV* (OpenCV, 2012) denominada por *cvContour* (v.d. Susuki e Abe, 1985; Bradski e Kaehler, 2008a) que devolve os pontos constituintes do limite das regiões segundo uma estrutura em árvore. Esta função admite como argumento de entrada uma imagem resultante de uma operação de deteção de contornos ou uma imagem binarizada como  $I_b(x, y)$ . Obtidos os contornos de cada região é efetuada a verificação de proximidade de intersecções de linhas verticais e horizontais com os extremos superiores esquerdo e direito da região. São aceites como candidatos as regiões que apresentem, em número de pixéis, distâncias na horizontal e vertical entre os pontos extremos e os pontos de intersecção das retas do histograma  $I_{fh}(\rho, \theta)$ , inferiores a 3% da largura da imagem (valor determinado empiricamente). Na Fig. 3.2.2.3 a) e b) pode ser visualizado o resultado final da binarização da imagem da Fig. 3.2.2.2 c) e d), e respetivos retângulos detetados através do método de segmentação nas imagens da Fig. 3.2.2.3 c) e d).



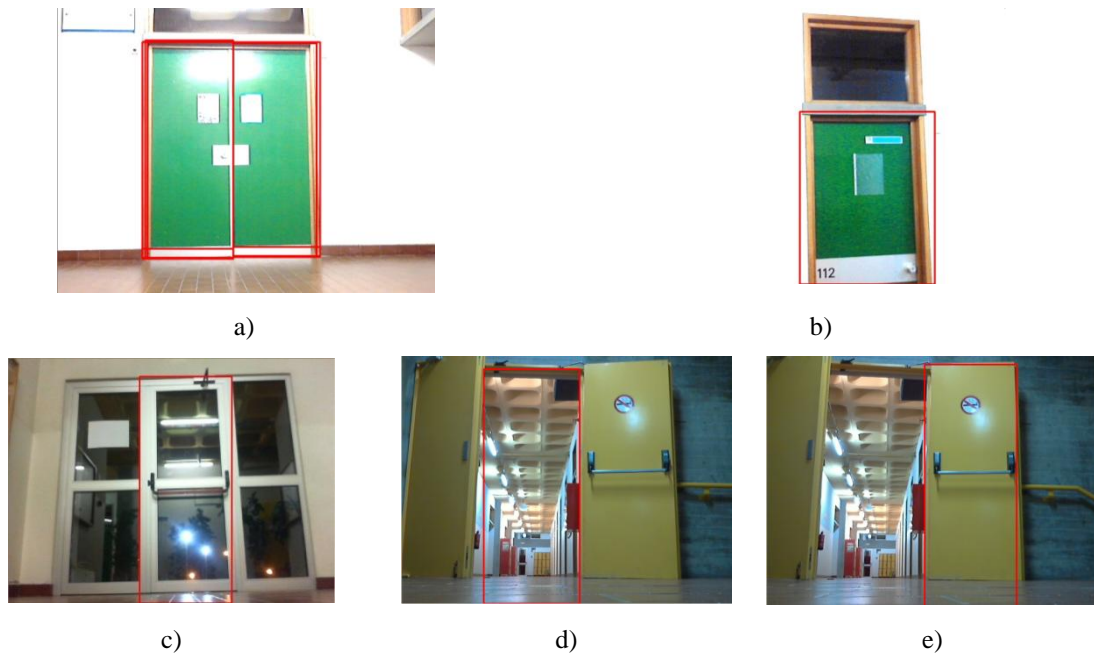
**Figura 3.2.2.3** – Segmentação binária das imagens de a) uma porta dupla e uma b) porta simples e os respetivos rectângulos detectados c) e d).

Encontrados os candidatos a portas pelos dois métodos é efetuada uma triagem, de forma a eliminar os retângulos que não façam parte de uma estrutura idêntica a uma

porta. Tomando como premissa que as portas apresentam uma altura relativamente superior à sua largura e ocupam uma área significativa da imagem, é efetuada uma análise à área e relação Altura/Largura, sendo unicamente aceites os candidatos que tiverem uma área superior a  $T_A$  e altura  $T_R$  vezes superior à largura do mesmo, ou seja, para que os candidatos a porta sejam validados como verdadeiros, têm que respeitar a relação de área  $A_{CP} \geq T_A$ , com  $A_{CP}$  a ser a área do candidato, e a relação entre a altura ( $H_{CP}$ ) e largura ( $W_{CP}$ ) descrita por  $\frac{H_{CP}}{W_{CP}} \geq T_R$ . Nos testes realizados foram aceites os candidatos com áreas superiores a 16% ( $T_A$ ) do número total de pixéis da imagem e relações de altura e largura superior a 1.2 ( $T_R$ ). Estes valores foram obtidos empiricamente após serem testadas várias portas em vários ambientes.

Esta abordagem mista, utilizando dois métodos para o reconhecimento, aumenta consideravelmente a taxa de deteção e a diversidade de portas para reconhecimento. A Fig. 3.2.2.4 apresenta em a) e b) os resultados da combinação obtida pelos 2 métodos propostos. O algoritmo desenvolvido para o reconhecimento de portas permite a sua deteção a partir b) de uma imagem parcial de uma porta e independentemente da sua natureza c) ou grau de abertura d) e e), resultando num algoritmo bastante completo para o reconhecimento de portas em ambientes interiores. Segundo os testes efetuados as portas podem ser reconhecidas para distâncias até 4 metros, com os valores para  $T_A$  e  $T_R$  acima referidos. O aumento da distância de deteção é possível, diminuindo os valores de  $T_A$  e  $T_R$ , contudo é notado um significativo acréscimo de falsos positivos gerados por objetos de forma semelhante e menores dimensões. Além disso, não faz sentido procurar portas a distâncias maiores que 4 metros para o tipo de utilizador previsto para este produto.

Na próxima secção será abordado a deteção e o reconhecimento de escadas. As escadas são estruturas que apresentam um elevado grau de risco para pessoas invisuais, podendo resultar em graves acidentes quando não detetadas. Contudo, as mesmas não podem ser vistas unicamente como um obstáculo de navegação, mas também como uma informação imprescindível numa navegação para a identificação de zonas de acesso a outros pisos.



**Figura 3.2.2.4** – Detecção dos rectangulos de possiveis portas, gerados a partir dos 2 métodos propostos.

### 3.2.3 Detecção e Reconhecimento de Escadas

Do ponto da visão computacional, as escadas podem ser caracterizadas por um conjunto de linhas paralelas com distâncias aproximadamente equidistantes entre si, quando vistas de frente ou quase de frente. No entanto, tais padrões de linhas são bastante comuns em estantes, gradeamentos ou mesmo paredes com diferentes texturas e que ao serem vistas de forma simplista podem ser confundidas com uma escada. De forma a minimizar os falsos positivos é adotado um conjunto de técnicas que visam aumentar o nível de certeza do algoritmo de reconhecimento.

O algoritmo proposto para reconhecimento encontra-se representado no diagrama de blocos da Fig. 3.2.3.1, é composto por uma fase (a) de pré-tratamento de imagem com o objetivo de realçar os contornos dos objetos, uma fase de extração de atributos para determinar (b) os contornos com recurso ao *Canny* (Canny, 1986) e (c) possíveis linhas da imagem através do detetor de linhas de *Hough* (Hough, 1962). Extraídos os conjunto de linhas (d) são excluídas as linhas que não respeitem os critérios de orientação requeridos e (e) distância mínima entre linhas. Para finalizar existe uma última fase onde é efetuado o pós-tratamento dos dados obtidos, através de uma análise (f) estatística através dos valores de média, desvio padrão e (g) histograma de cores do conjunto de linhas, sendo (h) retiradas ilações quanto à validação de escadas na imagem.



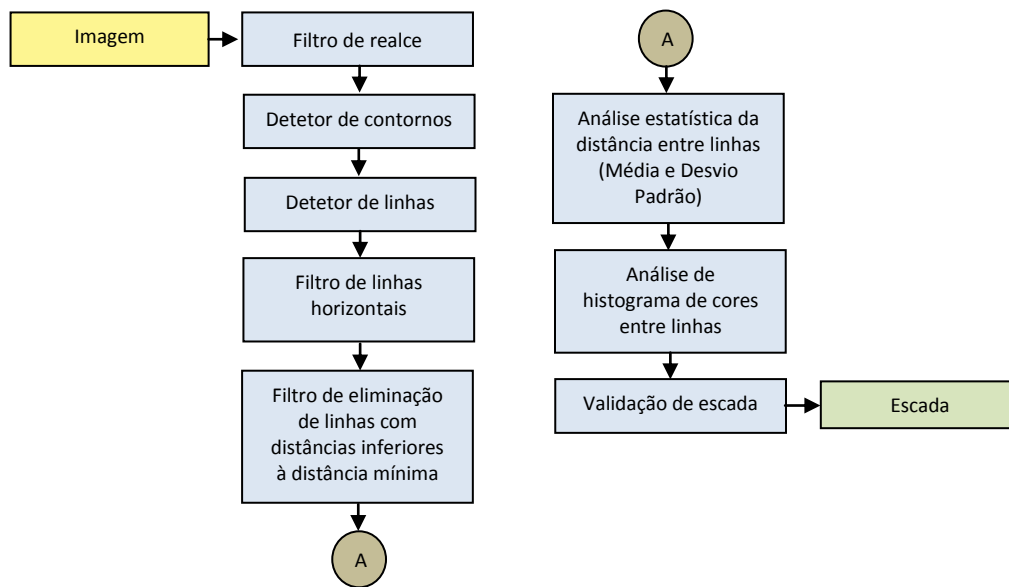


Fig. 3.2.3.1 – Diagrama de blocos do algoritmo para reconhecimento de escadas.

Como já referido, as escadas podem ser vistas como um conjunto de linhas dispostas paralelamente entre si, pelo que a sua deteção torna-se condição essencial. Como tal, e para que a eficiência de deteção de linhas aumente é necessário aplicar um filtro de realce á imagem original  $I(x, y)$ , ilustrada na imagem Fig. 3.2.3.2 a), com um filtro *Unsharp Mask* (Russ, 2011b) idêntico ao utilizado na secção anterior resultando numa imagem com maior nível de detalhe  $I_{UM}(x, y)$  como a apresentada na Fig. 3.2.3.2 b). De seguida é aplicado um o detetor de arestas Canny (Canny, 1986) com filtro de tamanho  $\sigma = 1.0$  e valores de *threshold* inferior  $T_l = 0.3$  e superior  $T_h = 0.5$  à imagem realçada em  $I_{UM}(x, y)$  originando uma nova imagem  $I_c(x, y)$  com os contornos dos objetos. Sendo a finalidade do algoritmo detetar linhas que definam uma possível escada é aplicada a transformada de Hough (Hough, 1962) à imagem  $I_c(x, y)$  sendo extraídas as linhas para um histograma  $I_h(\rho, \theta)$ . Por sua vez ao conjunto de linhas é realizada uma filtragem por linhas horizontais com inclinação limitada por um valor de *threshold*  $T_\theta$ , que segundo os testes efetuados não deverá ser superior a  $10^\circ$ , através da relação  $|90^\circ - I_h(\rho, \theta)| < T_\theta$  sendo criado um novo histograma de linhas  $I_{fh}(\rho, \theta)$  com os resultados da filtragem. De notar que o algoritmo é idêntico, em grande parte, a um dos métodos de deteção de portas, com a exceção do  $T_\theta$  usado, com isso consegue-se poupar tempo de cálculo, embora se perca alguma precisão que poderia ser melhorada,

contudo não significativamente, com o ajuste dos parâmetros do detetor de arestas Canny.

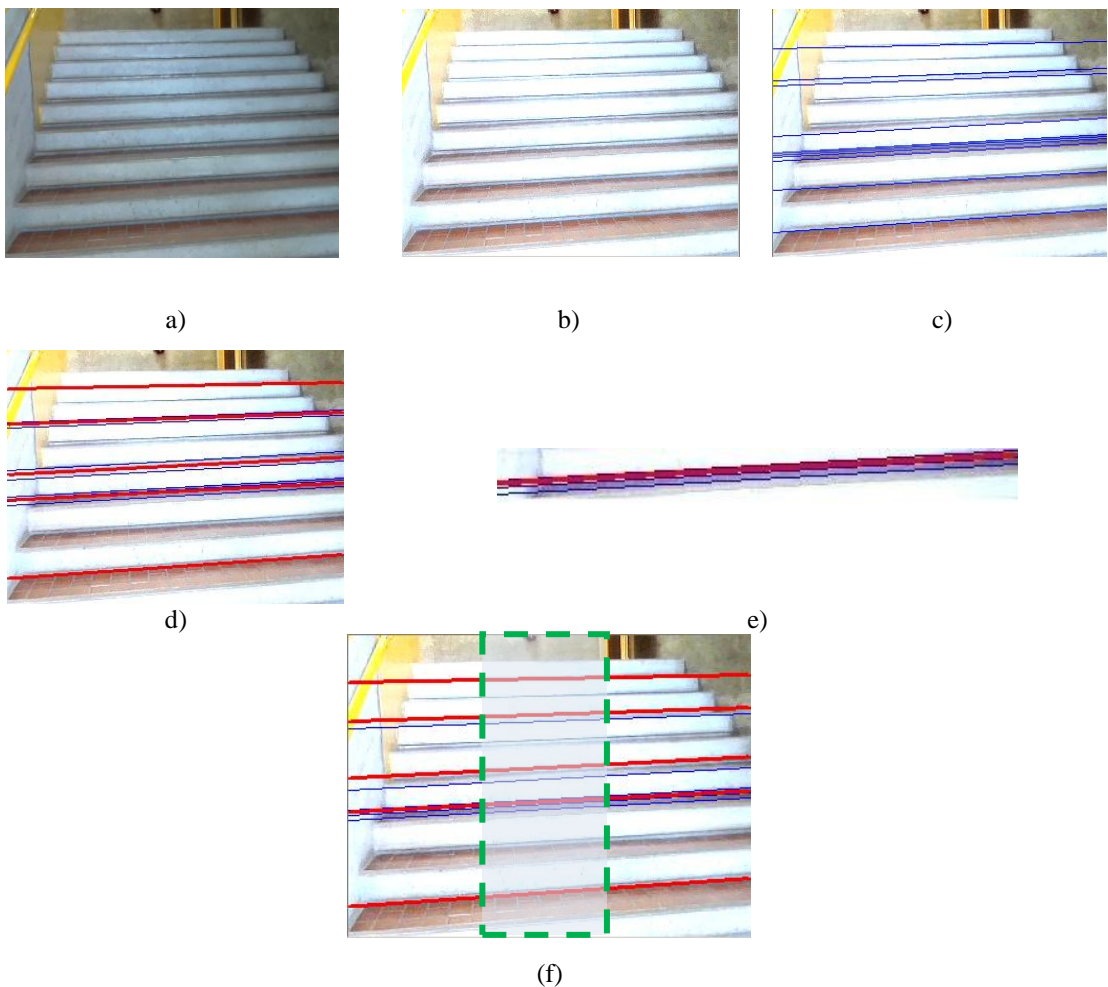
Na Fig. 3.2.3.2 c) encontra-se ilustrada a imagem com o conjunto de linhas horizontais extraídas identificando os degraus de uma escada. De modo a que sejam eliminados possíveis erros de detecção é necessário estabelecer um limite mínimo para a distância entre linhas para que seja considerado como válido a presença de um degrau, não menosprezando contudo a sua informação visto o conjunto de linhas poder fazer parte de um dos limites (superior ou inferior) de um degrau. Para que essa informação não seja ignorada é proposto substituir os conjuntos de linhas do histograma  $I_{fh}(\rho, \theta)$  com distâncias abaixo de um determinado *threshold*  $T_D$  de valor igual a 3% da altura da imagem, por uma única linha que seja a média do conjunto em análise (Fig. 3.2.3.2 d) e em mais detalhe em e)) resultando num novo histograma  $I_{ofh}(\rho, \theta)$ .

Para determinar a hipótese de escada é efetuada uma análise estatística à distância entre linhas, onde é deduzido a média ( $\mu_s$ ) e desvio padrão ( $\sigma_s$ ) entre linhas, de forma a verificar se existe um padrão bem definido entre elas. Para um número  $N_L$  ( $N_L \geq 4$ ) de linhas dispostas na horizontal, quanto menor for o desvio padrão ( $\sigma_s$ ) das distâncias entre linhas, maior será a probabilidade de pertencerem a uma escada. Para tal é tomado como princípio, para que seja validada a hipótese de escada, que a relação entre a média e o desvio padrão ( $R_s$ ) seja superior a um valor  $T_s$ , sendo que essa relação estabelece o grau de incerteza segundo a relação  $R_s = \frac{\mu_s}{\sigma_s}$ . São aceites como candidatos a escadas os conjuntos de linhas horizontais (superiores a 4 elementos) que satisfaçam o critério  $R_s \geq T_s$ . Nos ensaios realizados foi utilizado um valor de  $T_s=2$ , valor o qual determinado empiricamente. Valores superiores para  $T_s$  resultam em falhas de detecção enquanto valores inferiores por sua vez aumentam consideravelmente o número de candidatos e falsos positivos detetados.

No entanto a análise estatística por si só não indica fielmente a existência de uma escada, mas sim a existência de um padrão de linhas, pelo que é necessário efetuar uma validação que permita confirmar o padrão como pertencente a uma escada. O modelo proposto realiza a validação de escada através da análise do histograma de cores de cada um dos pares de linhas de  $I_{ofh}(\rho, \theta)$  que formam os possíveis degraus de escadas na imagem  $I_{UM}(x, y)$ . Considerando uma janela retangular centrada na imagem com largura de 40% e altura idêntica à da imagem  $I_{UM}(x, y)$ , como se encontra ilustrado na

Fig. 3.2.3.2 f), são criados os histogramas  $he(n)$  dos níveis de cinzento de cada um dos pares de linhas  $l_k$ , com  $k$  a variar entre 1 a  $N_L-1$ , resultando num conjunto de histogramas individuais, em que  $he_{l_k}(n)$  é o histograma em níveis cinza de 0 a 255 do par de linhas  $l_k$ , e  $he_{l_{(k+1)}}(n)$  o histograma do par de linhas seguinte  $l_{(k+1)}$ . Criados os histogramas são analisadas quais os níveis de cinzento que obtiveram maior contribuição, em cada um dos histogramas, e é verificado se a diferença entre as cores, com maior pico, dos histogramas de pares de linhas adjacentes  $l_k$  e  $l_{(k+1)}$  não ultrapassa um dado valor  $T_c$ , i.e., se

$$\left| \max\left(he_{l_k}(n)\right) - \max\left(he_{l_{(k+1)}}(n)\right) \right| < T_c, \text{ com } k = \{1, \dots, N_L-1\}. \quad (3.2.3.1)$$



**Figura 3.2.3.2** – Imagem de a) escadas antes e b) após aplicado o filtro de realce. Linhas detetadas, após a aplicação do filtro de linhas horizontais em c) e em d) a imagem resultante após eliminação dos *outliers* do conjunto de linhas detetadas. Detalhe da substituição e) de um conjunto de linhas, cujas distâncias estão abaixo da distância de *threshold*, por uma única linha que é a média de todo o conjunto. Em baixo, f) a janela central para computação dos histogramas entre linhas.

Caso se verifique que pelo menos 70% dos histogramas dos pares de linhas obtiveram picos de cinza não afastado mais que  $T_c = 40$  (valores determinados empiricamente) então o conjunto de linhas é validado como pertencente a uma escada. A utilização de um valor de 70% entre igualdade de histogramas permite uma maior flexibilidade, devido aos efeitos de perspectiva da imagem de uma escada originar com que na área inferior da imagem seja visível a superfície superior dos degraus e na área superior da mesma seja visível unicamente a superfície inferior dos degraus. Esta fase final de verificação permite inviabilizar regiões definidas por materiais de diferentes cores (cinzas) segundo um padrão como são exemplo algumas paredes.

É importante referir que o método final utilizado para a validação de um conjunto de linhas horizontais como pertencente a uma escada, através da comparação dos valores de pico dos histogramas de cor de cada par de linhas é muito influenciável pelo ruído, podendo resultar mesmo na não deteção de uma escada. Num futuro desenvolvimento, tendo em vista obter melhores resultados é sugerido uma análise não por valor de pico, mas sim pelo centro de massa dos histogramas, que acabam por ser amortecidos por desvios provocados pelo ruído na imagem.

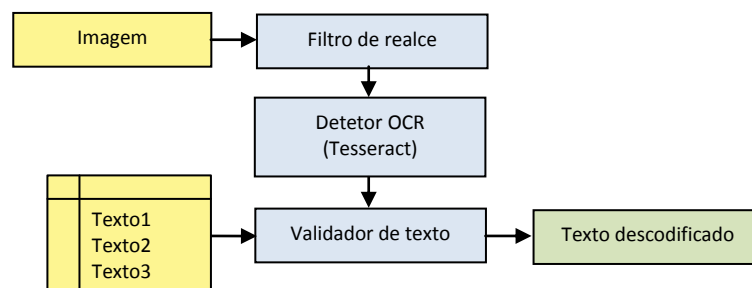
É possível ainda diferenciar escadas de outros objetos similares como estantes com prateleiras, visto teoricamente o tamanho dos degraus diminuir com o aumento da distância a que nos encontramos dele. Desta forma os degraus que se encontram mais distantes tendem a ter uma distância entre linhas inferior, aquelas que se encontram mais próximos de nós, ao contrário de uma vulgar estante em que as distâncias são sensivelmente iguais entre as prateleiras que a constituem. No entanto, este método (de validação de escadas vs estantes) é fortemente dependente da precisão de deteção, podendo em lugares de fraca iluminação inviabilizar o reconhecimento de escadas devido a falhas de deteção de degraus. Segundo os testes efetuados em ambientes interiores, cuja influência dos pontos de luz é significativa foram obtidos resultados bastante fracos sendo a eficiência de deteção de escadas comprometida, segundo as condições enunciadas. Deste modo este método de validação foi eliminado para o caso em demonstração (instalações do ISE, onde muitas vezes existem muitas luzes fundidas em alguns lanços de escadas), contudo, pode ser uma alternativa viável para a deteção de escadas em ambientes exteriores, onde as condições de iluminação não são tão restritas.

A seção seguinte aborda o reconhecimento ótico de caracteres, com o intuito de extrair textos de placares de informação e de sinalética vertical que oferecem geralmente informações de grande precisão para a localização.

### 3.2.4 Reconhecimento Ótico de Caracteres

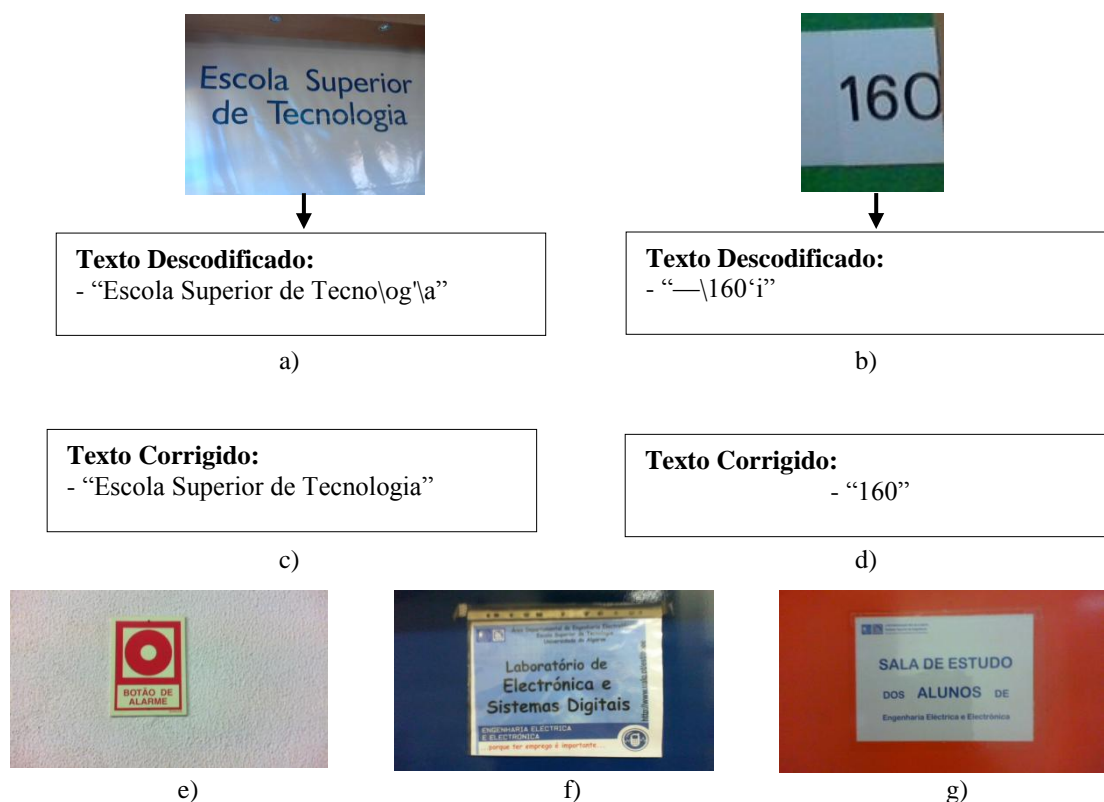
Para o reconhecimento de caracteres foi utilizada uma aplicação auxiliar Tesseract-OCR (Tesseract-OCR, 2007) que foi originalmente concebida pela Hewlett-Packard mas agora é suportada e mantida por uma equipa da Google. Esta aplicação é livre para utilização académica e comercial. O Tesseract é uma aplicação de consola que aceita como argumentos de entrada uma imagem, e retorna para um ficheiro de texto o resultado da descodificação.

Contudo, apesar de a aplicação possuir os seus próprios mecanismos de correção baseado num dicionário interno, existem muitas palavras cuja correção não é a mais apropriada resultando em frações de texto inválidas, havendo assim a necessidade de efetuar um pós-tratamento do texto descodificado de forma a tornar os textos válidos e possíveis de comparar com o SIG. É necessário portanto um conjunto de procedimentos, para além do reconhecimento ótico de caracteres, tendo em vista o reconhecimento de pontos de referência baseados em texto. Tais procedimentos encontram-se representados no diagrama de blocos na Fig. 3.2.4.1. A primeira etapa do algoritmo é (a) realçar a imagem capturada através de um filtro de realce apropriado como o *Unsharp Mask* (Russ, 2011b) sendo depois (b) evocada a aplicação periférica Tesseract (Tesseract-OCR, 2007) para extração de texto e (c) executado um processo de validação do texto segundo uma base de dados de palavras construídas a partir de informações introduzidas no SIG.



**Figura 3.2.4.1** – Diagrama de blocos do reconhecimento de caracteres.

Considerando, como exemplo, os trechos de texto descodificados resultantes das imagens representadas na Fig. 3.2.4.2 a) e b), pode-se observar que existem secções de texto descodificado inválidas, pelo que é necessário criar um modelo de pós-tratamento que permita retirar a informação necessária. Como solução foi desenvolvido um algoritmo que isola cada uma das palavras descodificadas e procede a uma comparação com palavras referência presentes numa base de dados, que é construída automaticamente a partir de referências textuais encontradas no SIG, de forma a efetuar a validação do texto. O método para validação proposto elimina todos os caracteres que não sejam alfanuméricos e efetua uma comparação caracter a caracter entre a palavra descodificada e cada uma das palavras presentes numa base de dados. Caso haja uma correspondência de caracteres superior a uma percentagem  $P$  (90%) entre as palavras que apresentam uma sequência idêntica, é efetuada a substituição pela sua congénere da base de dados. Pelo que o texto descodificado das imagens Fig. 3.2.4.2 a) e b), depois de aplicado o algoritmo de correção será o demonstrado na mesma figura c) e d).



**Figura 3.2.4.2** – Exemplos de trechos de texto descodificado utilizando o Tesseract-OCR, a) em um placar de parede e b) em uma porta de um gabinete e correção efetuada em c) e d). Imagens em que foi obtido descodificações corretas de e) um sinalizador de alarme, de f) um identificador de laboratório e uma g) sala de estudo.

Como referido o texto decodificado é filtrado e comparado com uma lista de palavras retiradas automaticamente do SIG, sendo que um texto do SIG é considerado como detetado caso haja no mínimo uma similaridade de 90% (*P*) entre o texto decodificado e o texto existente no Sistema de Informação Geográfica. Na Fig. 3.2.4.2 e-g), encontram-se ilustrados outros exemplos de decodificações de textos a partir de sinalizadores de alarme (e) e identificadores de área (f) e (g).

Nos testes efetuados verificou-se que a distância de deteção depende fortemente do tamanho dos caracteres e das condições de luminescência locais, sendo que na generalidade o alcance para deteção, considerando um tamanho de letra de 2 *cm* não é superior a 1 metro. No entanto esta é a distância normal de conforto de leitura, é de esperar que um invisual, ou alguém com deficiências visuais só pretenda obter informações de um placar de texto à mesma distância que as outras pessoas os leem, caso contrário o sistema estaria sempre a reportar todos os textos que encontrava ao longo da navegação, com todos os inconvenientes que isso implicaria ao utilizador invisual, pois este baseia-se muito na análise sonora do ambiente que o rodeia. Durante o período de ensaio, foi ainda verificado, que textos cujo espaçamento entre caracteres é reduzido apresentam taxas de decodificação reduzidas, sendo muitas vezes só possível decodificar letras soltas sem qualquer significado.

No seguimento desta seção será apresentado um método para o reconhecimento de retângulos e quadrados em imagens, que são de pouca utilidade para a localização visto existir uma infinidade de objetos cuja forma pode ser caracterizada por um retângulo, no entanto a sua deteção no meio é importante para servir de complemento a outras informações extraídas, podendo ainda ser utilizada como ferramenta para validação das mesmas.

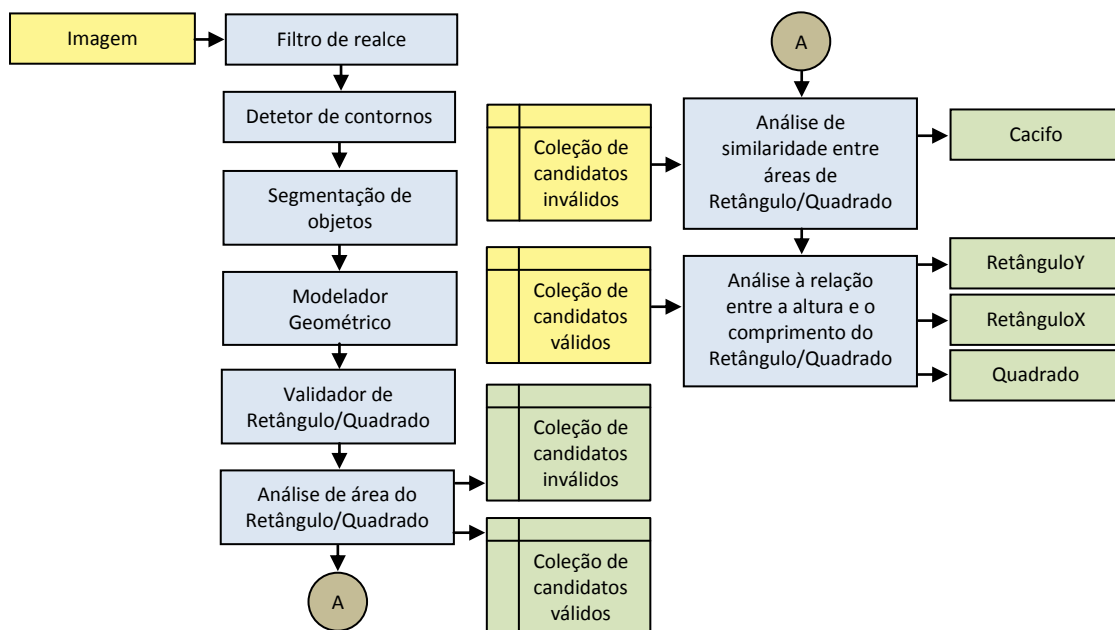
### **3.2.5 Reconhecimento de Retângulos e Quadrados**

Existe um número elevado de objetos que são caracterizados por estas formas tais como janelas, placares, vitrinas, indicadores, cacifos, etc. De forma a diferenciar esses objetos com base na sua forma geométrica, é realizada uma análise ao comprimento e altura dos retângulos efetuando uma distinção segundo essas componentes.

Esses retângulos são ainda classificados distintamente, quanto à área, sendo que os retângulos que não satisfaçam os critérios exigidos (devido às suas dimensões e por apresentarem pouca significância como pontos de referência) são considerados para

uma análise mais cuidada, procurando similaridades entre as áreas que sigam um padrão que possa ser indício da existência de um cacifo.

No diagrama de blocos da Fig. 3.2.5.1, encontra-se esquematizado o diagrama de blocos do algoritmo para reconhecimento. Tal como para o reconhecimento de portas e escadas é necessário que as fronteiras dos objetos sejam bem definidas, pelo que é necessário a aplicação (a) de um filtro de realce a preceder o (b) detetor de contornos Canny (Canny, 1986). Na imagem resultante com os contornos dos objetos é (c) efetuada a segmentação de cada um dos objetos e aplicado um (d) modulador geométrico para extração de formas geométricas que (e) segundo critérios de avaliação sejam idênticas a retângulos ou quadrados. Ao conjunto extraído, por sua vez, são realizadas (f) análises às áreas dos objetos, dividindo-os em dois conjuntos de candidatos, sendo que os que apresentam áreas que não satisfaçam os critérios exigidos sujeitos a uma (g) segunda análise, procurando possíveis similaridades entre as áreas, que indique a existência de um cacifo. Por outro lado, os candidatos com áreas dominantes são (h) comparados e classificados quanto às suas relações de altura e comprimento.

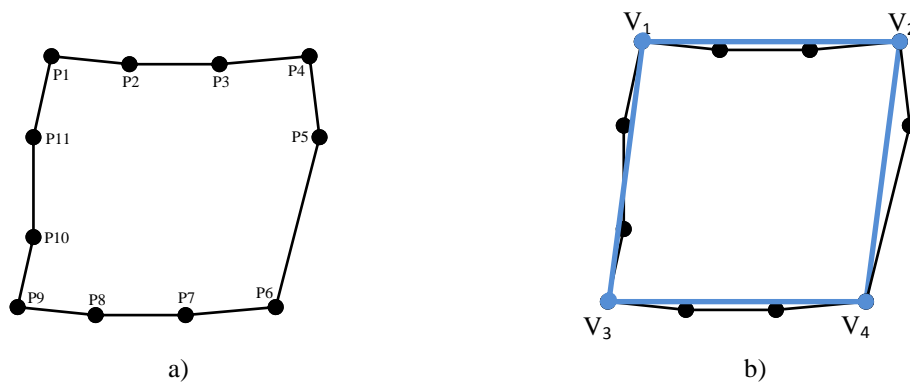


**Figura 3.2.5.1** – Diagrama de blocos do reconhecimento de retângulos.

Na etapa inicial do algoritmo a imagem capturada é realçada com recurso ao filtro *Unsharp Mask* (Russ, 2011b) e aplicado um detetor de contornos Canny (Canny, 1986) parametrizado com um filtro de tamanho  $\sigma = 1.0$  e valores de *threshold* inferior



$T_l = 0.3$  e superior  $T_h = 0.5$ . Detectados os contornos dos objetos são executados procedimentos que visam a extração de formas retangulares, modelo adaptado do OpenCV (OpenCV, 2003). O modelo de reconhecimento utiliza uma função nativa do OpenCV (*cvContour*) baseada no algoritmo de Susuki (Susuki e Abe, 1985; Bradski e Kaehler, 2008a) que devolve segundo uma estrutura em árvore, a sequência de pontos que constitui cada um dos contornos dos objetos (Fig. 3.2.5.2a), permitindo assim segmentar individualmente cada uma das regiões definidas. Ao conjunto de regiões extraídas é depois aplicada outra função, nativa do OpenCV (*cvApproxPoly*) baseada por sua vez no algoritmo de Ramer-Douglas-Peucker (Ramer, 1972; Douglas e Peucker, 1973; Bradski e Kaehler, 2008b), que permite obter um modelo simplificado contendo unicamente os vértices de maior significado da região (Fig. 3.2.5.2b).



**Figura 3.2.5.2** – Ilustração de exemplo dos algoritmos de Susuki e Abe, 1985 e Douglas e Peucker, 1973 aplicados a um retângulo. a) Pontos devolvidos da segmentação de um objeto utilizando o algoritmo de Susuki e Abe e b) modelo simplificado, do objeto segmentado, utilizando o algoritmo de Douglas-Peucker.

Conhecendo o número de vértices constituintes de um quadrado ou retângulo é analisado o tamanho das estruturas de pontos, devolvidas após simplificação, de cada uma das regiões segmentadas, sendo consideradas candidatas aquelas que tiverem 4 pontos e que definam uma estrutura convexa, através de um teste de convexidade utilizando o algoritmo *cvCheckContourConvexity* (Bradski e Kaehler, 2008c) disponibilizado pela livreria de visão OpenCV (OpenCV, 2012). No seguimento é examinado o ângulo interno formado entre os segmentos de reta do conjunto, verificando se formam ângulos aproximadamente retos entre si. Fazendo recurso à álgebra vetorial e considerando os segmentos de reta adjacentes, da Fig. 3.2.5.2b), compostos pelos vértices  $\{V_3(x, y), V_1(x, y), V_2(x, y)\}$ ,  $\{V_1(x, y), V_2(x, y), V_4(x, y)\}$  e  $\{V_2(x, y), V_4(x, y), V_3(x, y)\}$  como vetores, é possível determinar o ângulo definido por

cada um dos conjuntos individualmente, utilizando a fórmula para o cálculo do produto escalar entre vetores  $\bar{u}$  e  $\bar{v}$ , ou seja,  $\bar{u} \cdot \bar{v} = |\bar{u}| \times |\bar{v}| \times \cos(\varphi_{\bar{u} \wedge \bar{v}})$ . Considerando genericamente um conjunto formado pelos vértices  $\{V_a(x, y), V_b(x, y), V_c(x, y)\}$  temos que:

$$\begin{aligned} \cos(\varphi_{V_a V_b \wedge V_c V_b}) &= \frac{\overline{V_b(x, y)V_a(x, y)} \cdot \overline{V_b(x, y)V_c(x, y)}}{|\overline{V_b(x, y)V_a(x, y)}| \times |\overline{V_b(x, y)V_c(x, y)}|} \\ &= \frac{(x_{V_a} - x_{V_b})(x_{V_c} - x_{V_b}) + (y_{V_a} - y_{V_b})(y_{V_c} - y_{V_b})}{\sqrt{(x_{V_a} - x_{V_b})^2 + (y_{V_a} - y_{V_b})^2} \times \sqrt{(x_{V_c} - x_{V_b})^2 + (y_{V_c} - y_{V_b})^2}}. \end{aligned} \quad (3.2.5.1)$$

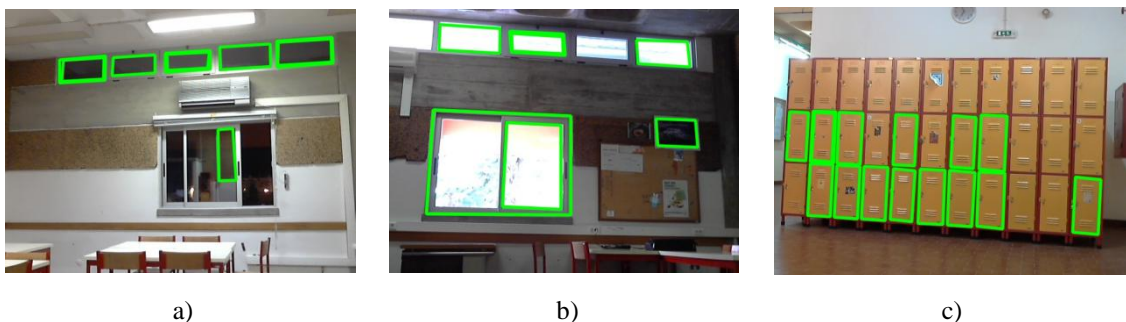
Fica provado a existência de um objeto retangular, caso se verifique para os três conjuntos de vértices, que os segmentos de reta são ortogonais  $\cos(\varphi_{V_a V_b \wedge V_c V_b}) < T_{ort}$ , considerando um valor de *threshold* para a validação de ortogonalidade ( $T_{ort}$ ) inferior a 0.3 (valor obtido experimentalmente). Extraídos os candidatos é averiguado se a área individual ( $A_{RA}$ ) de cada candidato respeita o critério exigido quanto à área mínima ( $T_{RA}$ ), isto é, se  $A_{RA} > T_{RA}$  ( $T_{RA}$  de valor igual a 10% da área total da imagem), sendo separados os candidatos em conjuntos distintos de acordo com as conclusões obtidas.

Os candidatos que foram assumidos inválidos na análise de área ( $A_{RA} \leq T_{RA}$ ) são sujeitos a uma análise posterior, sendo escolhidos os candidatos que possuam uma relação ( $R_{HvsW}$ ) entre a altura ( $H_{RC}$ ) e o comprimento ( $W_{RC}$ ), como demonstrado por  $R_{HvsW} = \frac{H_{RC}}{W_{RC}}$ , superior a um determinado valor de *threshold* ( $T_{HvsW}$ ) com valor igual a 1.2, isto é, são aceites os retângulos verticais que respeitem a relação  $R_{HvsW} > T_{HvsW}$ .

Filtrados os candidatos são procuradas similaridades entre as áreas com a finalidade de encontrar padrões que possam ser indício da presença de um cacifo no *frame* (imagem) capturada. Desta forma é executado um processo iterativo em que cada candidato  $ri$  do conjunto irá ser comparado com os restantes candidatos  $rj$  segundo a fórmula  $T_{RA_{inf}} \leq \frac{A_{ri}}{A_{rj}} \leq T_{RA_{sup}}$ , sendo que  $A_{ri}$  é a área do candidato  $ri$ ,  $A_{rj}$  a área de cada um dos restantes candidatos  $rj$  do subconjunto,  $T_{RA_{inf}}$  e  $T_{RA_{sup}}$  valores de *threshold* para validar os candidatos como similares ( $T_{RA_{inf}} = 0.5$  e  $T_{RA_{sup}} = 1.5$  nos ensaios realizados). Para cada iteração é contabilizado o número de candidatos com áreas semelhantes, sendo confirmada a existência de um cacifo quando contabilizados 4 candidatos idênticos (Fig. 3.2.5.3c).

Por outro lado o conjunto de candidatos validados pela análise da área mínima ( $A_{RA} > T_{RA}$ ) é sujeito a uma verificação quanto às relações de altura ( $H_{RC}$ ) e comprimento ( $W_{RC}$ ), através do método já utilizado anteriormente para caracterizar os objetos em retângulos verticais (altura maior que o comprimento  $R_{HvsW} > 1.2$ ), retângulos horizontais (comprimento maior que a altura  $R_{HvsW} < 0.8$ ) e quadrados (altura e comprimento semelhantes  $0.8 \leq R_{HvsW} \leq 1.2$ ).

Através do reconhecimento de retângulos na imagem é nos possível distinguir formas, tais como janelas, v.d., Fig. 3.2.5.3a) e b) e cacifos (Fig. 3.2.5.3c), contudo, não é possível afirmar com precisão se os retângulos detetados pertencem a janelas ou a outros objetos de características semelhantes.



**Figura 3.2.5.3** – Exemplos de retângulos detetados.

O reconhecimento de retângulos apesar de não fornecer informações claras quanto ao tipo de objeto, pode quando combinado com outros objetos reconhecidos no *frame* capturado fornecer informações importantes para a localização através de associações de proximidade com auxílio do SIG.

Deve-se ainda referir que a deteção de retângulos pode ainda ser combinada com a deteção e reconhecimento de objetos (próxima secção). De forma a simplificar e acelerar o reconhecimento de sinalização de alertas ou informativa, que normalmente em ambientes interiores está enquadrada dentro de “molduras” com esta forma. Do mesmo modo pode servir como método para validação dos mesmos.

### 3.2.6 Reconhecimento de Objetos

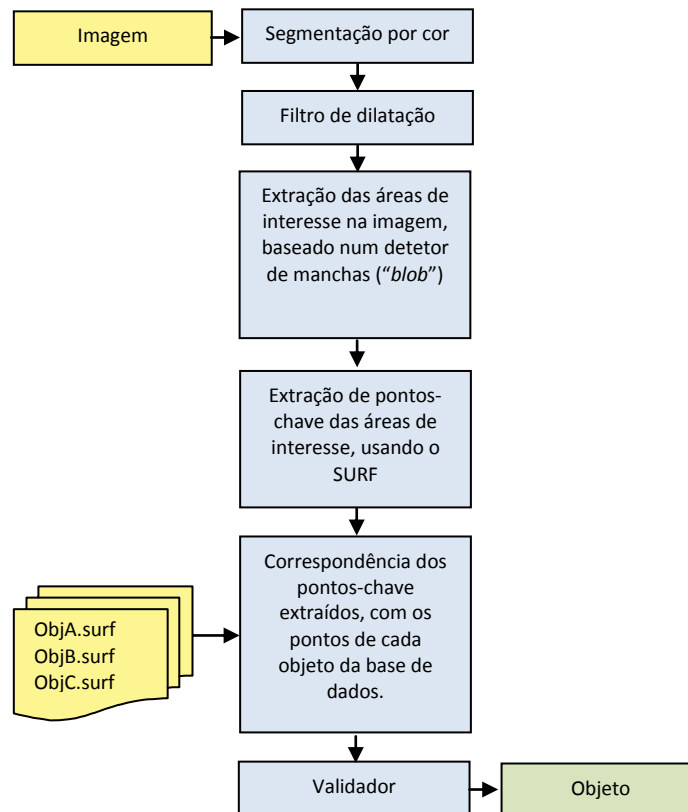
Como já abordado na secção 2.3, existe uma variedade de métodos para o reconhecimento de objetos. Devido a fiabilidade, velocidade de processamento e a facilidade de utilização escolhemos o algoritmo SURF (ver mais detalhes na secção

2.3). Este algoritmo permite identificar objetos, através da busca de correspondências numa imagem. O processo de correspondência tem em conta vários pontos de referência pré-adquiridos a partir de diferentes perspectivas de um objeto ou cena. Os atributos e pontos de interesse extraídos de um objeto são relativamente invariantes à escala, rotação, distorção, e diferença de luminescência, sendo possível a sua identificação numa imagem, independentemente da sua localização, posição ou rotação.

Como referido, para que um objeto seja possível de ser identificado numa imagem através do SURF, é necessário existir uma base de imagens (dados) constituída pelos pontos-chave do objeto. A construção da base de dados do objeto deve ter por base um número razoável de vistas, incidindo essencialmente nos ângulos mais relevantes do objeto.

Para a extração de pontos-chave de objetos com o algoritmo de SURF, numa imagem, é utilizada a biblioteca auxiliar OpenSurf (Evans, 2009). Esta biblioteca oferece ferramentas simples para extração e correspondência de pontos, sendo possível analisar rapidamente quais e quantos pontos foram identificados como correspondências corretas. Na Fig. 3.2.6.1 encontra-se representado o diagrama de blocos exemplificativo das várias etapas que compreendem a fase de reconhecimento de objetos. Este é constituído por (a) uma segmentação quanto à cor seguido da aplicação (b) de um filtro de dilatação para criar regiões em mancha. Depois são (c) identificadas as áreas das regiões segmentadas e são (d) extraídos os pontos-chave para (e) comparação com as bibliotecas de objetos e (f) aplicadas regras para validação.

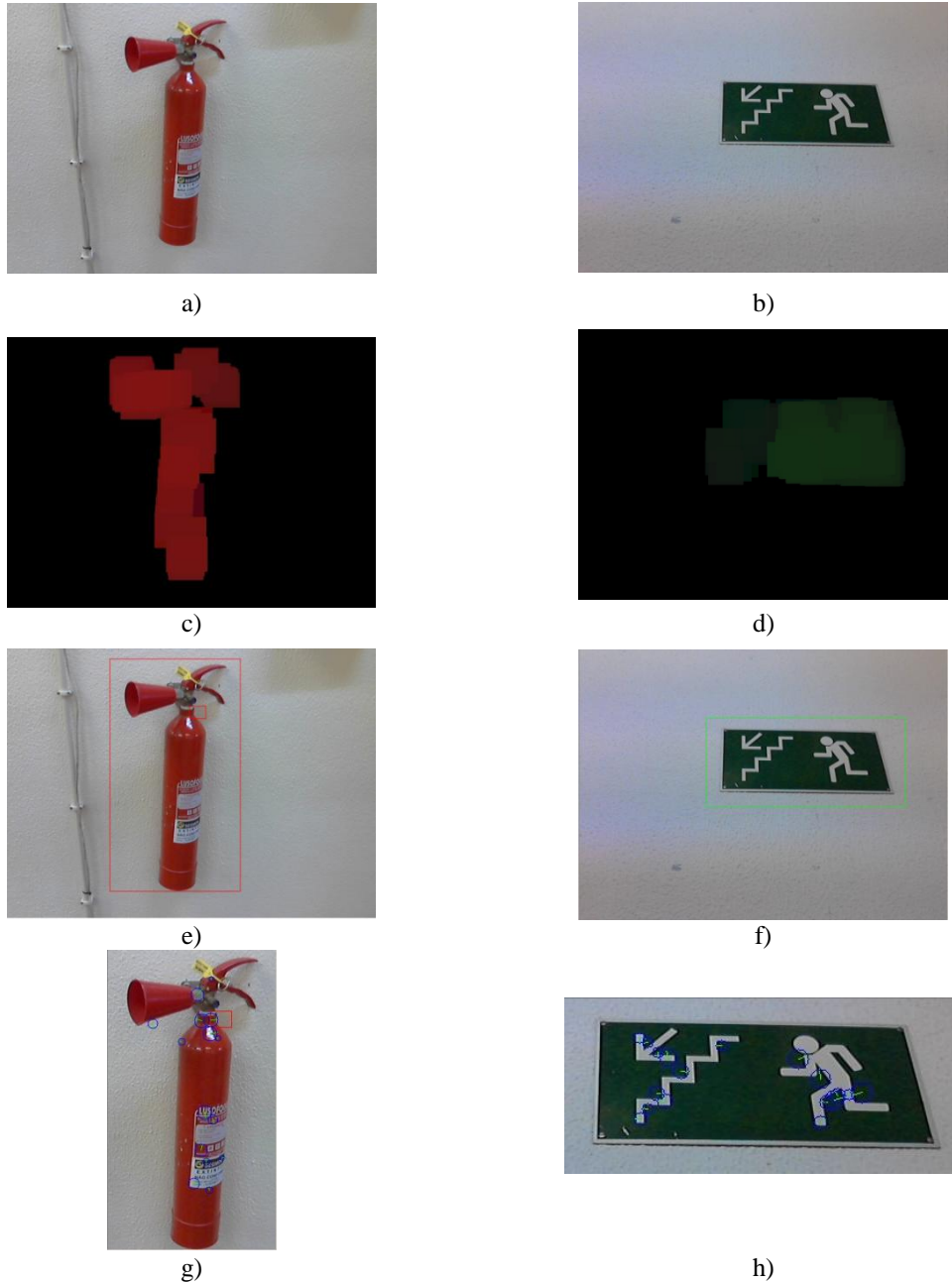
De modo a otimizar o processo de reconhecimento e diminuir a exigência computacional, ao mínimo possível, é utilizada uma técnica de segmentação que limita a procura de atributos a pequenas áreas de potencial interesse e não à imagem no seu todo. Tomando por princípio que os objetos a reconhecer são constituídos por cores praticamente uniformes, é possível efetuar uma segmentação por cor isolando as áreas com predominância de determinadas gamas de cor, seguido de um filtro de dilatação, com um período de 30 iterações, para dissolver pequenas áreas que apresentem cores diferentes e insignificantes quando comparadas com a cor principal do objeto, v.d. Fig. 3.2.6.2c) e d). A segmentação de cor é efetuada unicamente para as cores mais populares entre os objetos observados no ambiente de navegação indoor, como o vermelho, verde e azul não tendo sido utilizado nenhum algoritmo em específico para a segmentação de regiões.



**Figura 3.2.6.1** – Diagrama de blocos da identificação de objetos, tendo por base o algoritmo SURF.

Efetuada a segmentação por cor é utilizado uma função baseada no algoritmo de Susuki e Abe, 1985; Bradski e Kaehler, 2008a (v.d. secção anterior), para devolver o conjunto de pontos que formam os limites de cada uma das regiões segmentadas, extraíndo de cada um dos conjuntos os extremos para criação de uma área retangular de interesse, v.d. Fig. 3.2.6.2e), 3.2.6.2f).

Identificadas as áreas de interesse (RoI) é aplicado um algoritmo que descreva e detete os pontos-chave (SURF) de cada uma dessas regiões individualmente, na imagem original. Os pontos recolhidos são depois comparados com os pontos-chave de objetos referência guardados numa base de dados, sendo que o objeto é considerado como detetado caso haja correspondência efetiva em pelo menos 3 atributos SURF, v.d. Figura 3.2.6.2g) e h). Através da utilização do SURF e do algoritmo desenvolvido é possível reconhecer uma grande diversidade de objetos, adquirindo os preciosos pontos de referência necessários para a localização do individuo no meio *indoor*. Na figura 3.2.6.3 estão exemplificados alguns dos objetos possíveis identificar (*templates*) a partir da biblioteca de objetos criada.



**Figura 3.2.6.2** – Imagens capturadas de um a) extintor de incêndio e de b) um sinalizador de escadas. Imagens após a segmentação por cor e aplicação do filtro de dilatação c) e d). Resultado da determinação das áreas de interesse delimitando o e) extintor e o f) sinalizador. Pontos-chave em que foi verificada correspondência g) e h).

A utilização do SURF provou-se de grande eficácia, contudo deverá ser tido em especial atenção certos aspetos que podem interferir diretamente na eficácia de deteção, como o tamanho, uniformidade do objeto, o valor de *threshold* assumido para a resposta da matriz de *Hessian* e a distância máxima para deteção, que pelos testes práticos realizados não deverá ser superior a 2 metros, considerando as dimensões dos objetos amostra. A distância máxima está diretamente relacionada com o tamanho dos objetos e

com a sua contribuição em número de pontos-chave. Objetos definidos por áreas pequenas (fixando uma distância para deteção não inferior a 1 metro), devido ao seu tamanho, como por exemplo o da Fig. 3.2.6.3c), ou por ser baseado numa perspetiva afastada apresentam uma contribuição insignificante em número de pontos-chave, sendo de evitar o seu uso. Para efeitos futuros deverá ser criada uma pré-filtragem das áreas de interesse por área mínima, visto áreas pequenas não servirem como contributo válido.

De igual modo, objetos muito uniformes (v.d. Fig. 3.2.6.3d)) são pouco produtivos em número de pontos e desaconselhados como pontos de referência para a navegação. O valor escolhido para *threshold* ( $T_{hess}$ ) da resposta da matriz de *Hessian* é de extrema importância, visto ser com base no seu valor que irá ser realizada a filtragem dos pontos de interesse, sendo validados aqueles que obtenham picos de resposta acima do valor escolhido para *threshold*. Valores baixos permitem a extração de um maior número de pontos, contudo muitos deles de baixo nível de detalhe, resultando posteriormente num aumento de deteções incorretas. Por outro lado, o aumento do valor de *threshold* aumenta o nível de detalhe dos pontos, sendo escolhidos os máximos mais fortes, o que resulta consequentemente na extração de um número inferior de pontos, dificultando na identificação dos objetos. Como tal a escolha do valor de *threshold* a utilizar é essencial, estando o seu valor diretamente ligado à eficiência de deteção. Pelos testes efetuados, com vários objetos referência, foi escolhido um valor de  $T_{hess} = 0.0015$ , podendo no entanto o seu valor variar dentro de um intervalo  $T_{hess} \in [0.0025;0.0005]$ .



a)



b)



c)



d)



e)



f)

**Figura 3.2.6.3** – Exemplos de objetos detetados. a) Sinalizador de Extintor de PO, b) extintor de PO, c) botoneira de emergência, d) Armário de extinção de incêndio, e) sinalizador de zona wireless e f) sinalizador de elevador.

Detetados os objetos no meio é necessário situá-los no SIG e determinar com base nos mesmos qual a localização do indivíduo. Esse tema será abordado sucintamente na próxima secção.

### **3.3 Processo de Localização e Navegação**

Existem diversas técnicas e teorias desenvolvidas visando a localização e navegação de robôs em ambientes *indoor* através de modelos matemáticos, muitos dos quais com provas reconhecidas e de potencialidade comprovada. Contudo, ao contrário de um robô, em que os movimentos exigidos são cumpridos na íntegra, no humano o seu poder de imprevisibilidade e capacidade de auto decisão pode resultar em falhas irreversíveis, quando ignoradas as ordens de movimento indicadas pelo sistema. O que veio acrescentar a necessidade da criação de um modelo robusto para a localização automática no meio *indoor*, quando o utilizador se desvia da rota determinada.

A localização e navegação usualmente são apoiadas por mapas elaborados manualmente ou automaticamente numa fase de aprendizagem durante a navegação, sendo a sua utilização instrumento imprescindível. O rigor e a pluralidade de informação disponível, por seu lado, são vitais para a eficácia da determinação da localização, sendo que quanto maior for o número de pontos de referência distintos, maior será a probabilidade de determinar a localização correta. A diversidade de informação deverá, sempre que aplicável, ser escolhida em detrimento da quantidade, sendo preferível a utilização de poucos pontos de referência mas que permitam uma localização imediata, do que muitos pontos de referência vulgares e consequentemente de pouco significado.

Para esta dissertação a localização e navegação têm por base mapas de vetores do Instituto Superior de Engenharia da Universidade do Algarve. A leitura dos dados geográficos do mapa de vetores é realizada fazendo recurso ao GDAL 1.9.0 (*Geospatial Data Abstraction Library*) (GDAL, 2012) que é uma livraria de interpretação de formatos de dados geográficos distribuída pela *Open Source Geospatial Foundation* (OSGeo, 2013).

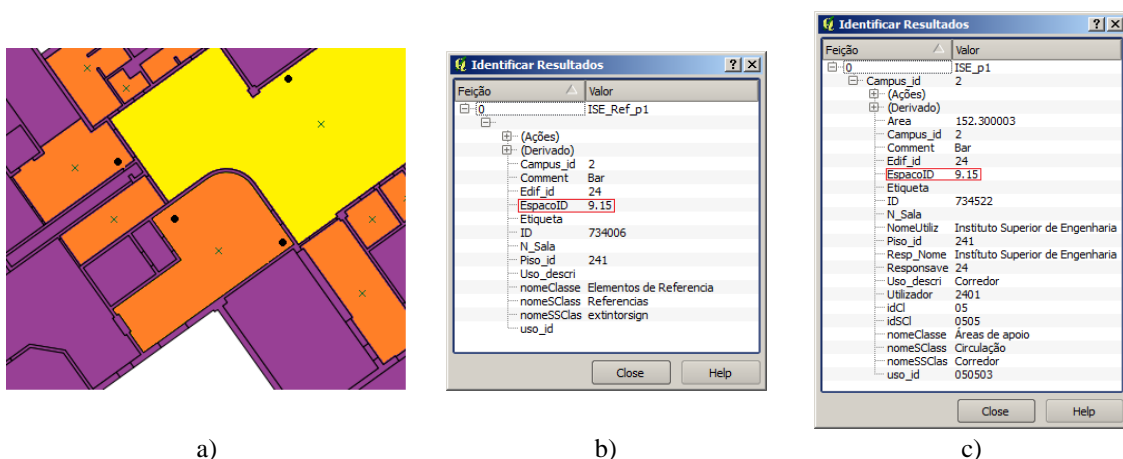
Utilizando o GDAL é possível extrair as informações geográficas das divisões, pontos de referência e estruturas tais como paredes, portas e janelas. Estas informações são lidas e armazenadas em memória criando uma base de dados onde constam as informações necessárias para a navegação, tais como o tipo de estrutura (sala, parede,



janela, porta) ou objeto para referência (extintor, indicador, cacifo, etc.), sua localização geográfica através do centróide geométrico e relações de vizinhança entre cada um dos dados extraídos. As relações de vizinhança são de extrema importância para o planejamento do percurso de navegação e determinação da localização atual.

Os objetos para ponto de referência foram introduzidos nos mapas de vetores do SIG, através da ferramenta de edição Quantum GIS 1.7.1 (QGIS, 2011) referenciando esses pontos quanto à sua posição global, através de coordenadas geográficas, e região a que pertence através de um identificador partilhado pelo ponto de referência e pela região onde está inserido, como são exemplo corredores, salas de aula etc. Por sua vez, as regiões estão agrupadas espacialmente formando relações de vizinhança entre si através de paredes, janelas, portas e escadas. Essas relações permitem determinar as vizinhanças entre regiões através da análise de contiguidade entre estruturas de acesso como portas ou escadas comuns entre regiões.

Na Fig. 3.3.1 a), encontra-se representado o exemplo de uma região (assinalada a amarelo) e regiões vizinhas (assinaladas a laranja) que podem ser acedidas através de uma porta e respetivos centróides geométricos (indicados por uma cruz) que fornecem as informações de posicionamento geográfico. Na mesma figura, ainda pode ser visualizado os pontos de referência inseridos nas regiões (assinalados pelos pontos pretos) e que estão referenciados às mesmas através de um identificador comum que permite relacionar o ponto à região a que lhe pertence, como pode ser visualizado na figura 3.3.1 b) pelo atributo *EspacoID* (destacado a vermelho) do ponto de referência e na figura 3.3.1 c) pelo mesmo atributo mas agora referente à região.



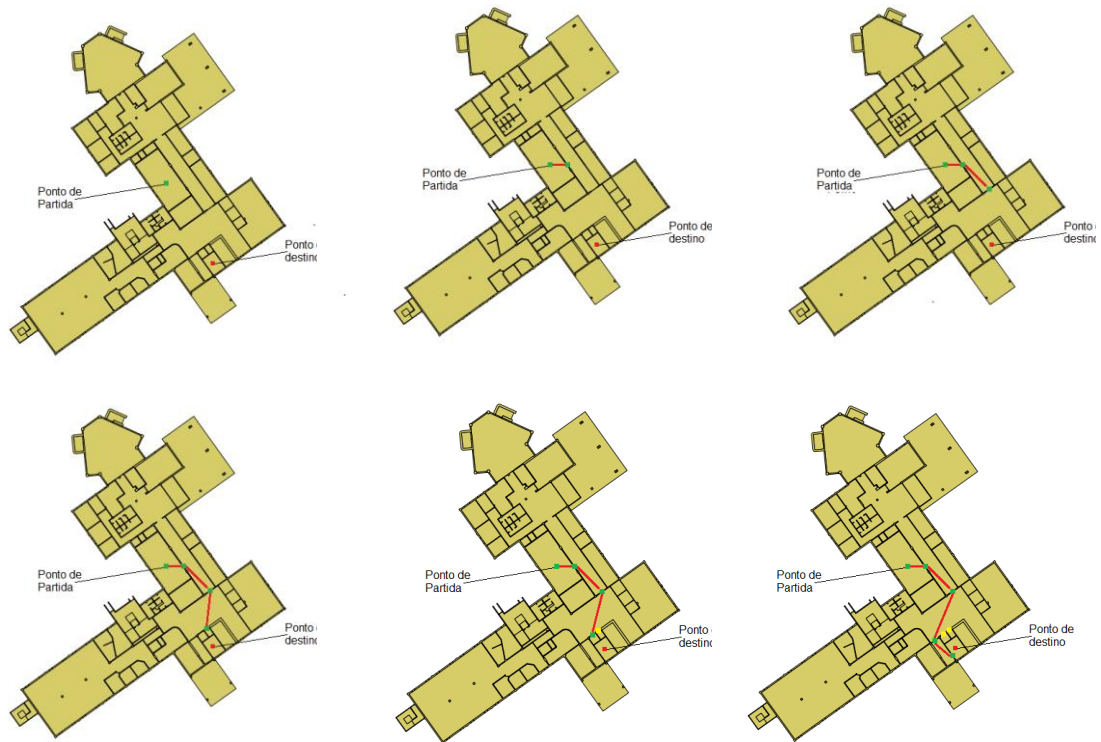
**Figura 3.3.1** – Representação gráfica, em pormenor, retirada do SIG apresentando a) uma região (assinala a amarelo) e regiões vizinhas (assinaladas a laranja) que partilham as mesmas portas e pontos referência pertencentes a cada uma das regiões (assinalados por pontos). Visualização dos campos constituintes de b) um ponto de referência e da c) região onde o mesmo está inserido.

Os relacionamentos referidos permitem construir um mapa matricial, em formato binário, indicando as vizinhanças entre regiões e entre estas e as estruturas de acesso e objetos para referência. Esse mapa matricial permite posteriormente uma rápida análise para os cálculos de determinação da localização e planeamento do percurso de navegação, que irá ser a seguir referido.

Para que seja possível navegar entre dois pontos num mapa é condição essencial estabelecer uma rota de navegação que permita traçar um caminho válido até ao destino desejado, envolvendo o menor esforço possível, isto é, considerando o menor deslocamento possível. Deste modo, o cálculo do percurso é obtido tendo em conta o percurso com menor custo de distância, entre as regiões (divisões) até ao seu objetivo final e considerando como pontos de transição as portas e escadas nas suas fronteiras.

O algoritmo para cálculo da rota de navegação é um processo iterativo que avalia em cada passo, qual a direção do ponto de destino e verifica no seu mapa matricial, com as informações de vizinhança, qual é a estrutura de acesso (porta ou escada) que se encontra mais próxima de si na direção do mesmo. Conhecida a estrutura de acesso é verificado qual a região, ou divisão, a que lhe dá acesso sendo calculado novamente a direção a que se encontra o ponto de destino e analisada qual a estrutura de acesso (porta ou escada) que lhe fica mais próxima na direção indicada. Este processo é repetido até que seja encontrado o ponto de destino, devolvendo em formato de vetor, o percurso que considera o mais apropriado. Sempre que seja encontrada uma divisão a qual não haja progressão possível, o mapa de vizinhanças é atualizado e são marcadas como inválidas para o percurso pretendido, a região e respetivas estruturas de acesso (portas e escadas), sendo iniciado a partir do ponto de partida o cálculo de uma nova rota, com o mapa de vizinhanças atualizado. Na Fig. 3.3.2 podem ser visualizadas algumas das iterações do processo da determinação da rota de navegação entre dois pontos de um mesmo piso do ISE. Fixando como ponto de partida a região marcada pelo ponto verde e o ponto de destino a região marcada pelo ponto vermelho (1ª linha à esquerda) é verificada qual a porta que se encontra mais próxima segundo a direção do ponto de destino, porta a qual por sua vez dá acesso a uma nova região. Após descoberta uma nova região é verificada novamente (1ª linha ao centro) a porta que segundo a mesma direção dá acesso a uma nova região. Processo idêntico é realizado na iteração seguinte (1ª linha à direita) procurando sempre a porta com menor custo associado na direção do ponto de destino. Tendo chegado a uma região cuja progressão

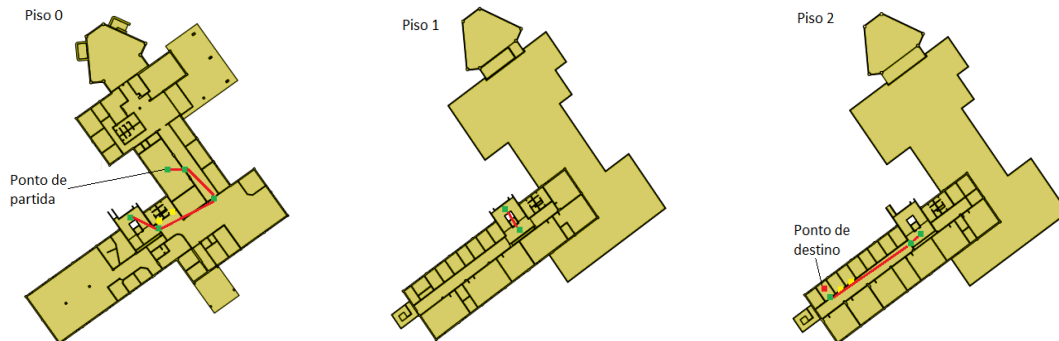
não é possível (2ª linha à esquerda e ao centro) é marcado no mapa de vizinhanças a região e respectivas portas como não transitáveis para o percurso requerido e é iniciado novamente todo o processo a partir do ponto de partida. O processo é repetido até que seja encontrada a região final (2ª linha à direita).



**Figura 3.3.2** – Planeamento de rota de navegação entre dois pontos de um mesmo piso.

Contudo a navegação em meios *indoor* não deve ser vista como estando limitada unicamente a um piso, devendo possibilitar o planeamento de rotas entre regiões pertencentes a pisos diferentes. Essa necessidade é solucionável através de uma interpretação faseada de cada piso, fixando como ponto temporário de destino a estrutura de acesso (escada) que dá acesso a um piso diferente e na mesma direção do ponto de destino. Na figura 3.3.3 pode ser visualizado o caminho planeado entre um ponto de partida numa região do piso 0 (indicado na Fig. 3.3.3 à esquerda) e um ponto de destino no piso 2 (indicado na Fig. 3.3.3 à direita). Inicialmente é verificada qual a escada pertencente ao mesmo piso que o ponto de partida, mais próxima segundo a direção do ponto destino (Fig. 3.3.3 à esquerda). De seguida é calculado o percurso com menor custo de distância através do mesmo processo de passos explicado para a figura 3.3.2. Transitando para um novo piso (Fig. 3.3.3 ao centro) é verificado que o mesmo não é idêntico ao piso do destino final e é novamente analisada qual a escada mais

próxima segundo a direção do destino e calculado o percurso entre as duas escadas. Finalmente encontrando-se no mesmo piso que o ponto de destino desejado (Fig. 3.3.3 à direita), é traçado o caminho até o objetivo ser atingindo, uma vez mais utilizando o processo iterativo já mencionado.



**Figura 3.3.3** – Planejamento de rota de navegação, tendo como ponto de partida uma sala no piso 0 e como destino uma sala no 2º Piso.

O trajeto planejado é armazenado sobre o formato de vetor, onde constam os ponteiros para o SIG de cada uma das regiões e estruturas de acesso (portas e escadas) organizados sequencialmente do ponto de partida para o ponto de destino. As informações armazenadas por sua vez permitem que o sistema auxilie, o utilizador a deslocar-se através do meio *indoor* orientando o mesmo a procurar portas ou escadas que lhe dê acesso às regiões traçadas para a rota de navegação.

Para orientar o utilizador ao longo do percurso de navegação é fundamental determinar a sua localização, para poder fornecer as indicações de movimento corretas para seguir o trajeto definido, ou corrigir o seu movimento.

Através das informações extraídas por visão (secção 3.2), é possível desenvolver um sistema de localização em tempo real que tendo por base os dados capturados do meio, calcule uma posição estimada para o utilizador, e permita a localização automática do mesmo partindo de um ponto desconhecido.

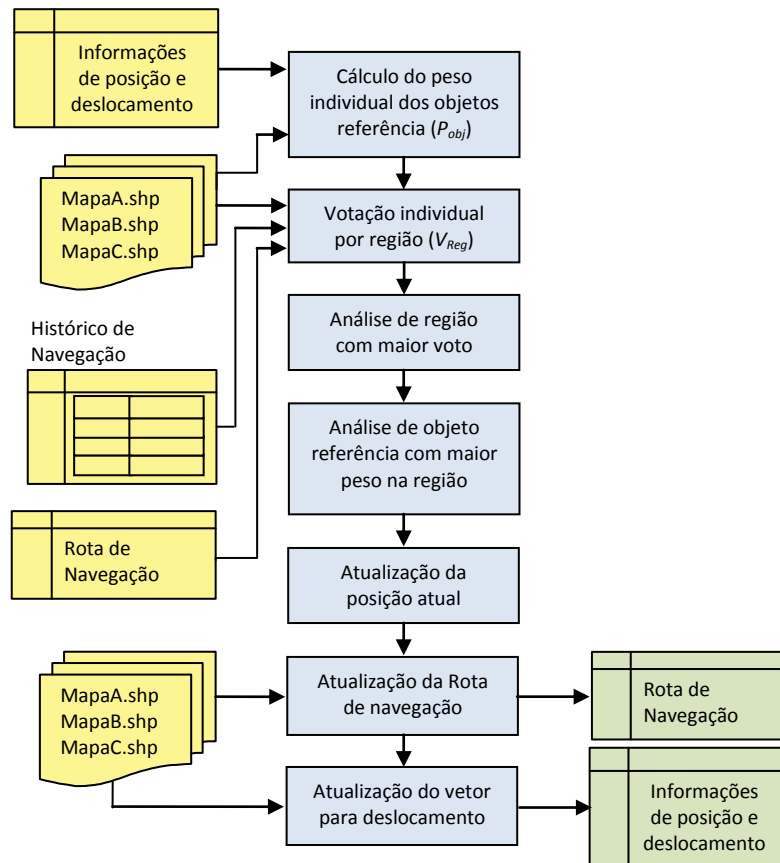
O processo desenvolvido está representado no diagrama de blocos da figura 3.3.4 e tem como objetivo permitir essa mesma localização apoiado nos dados obtidos a partir de um histórico de navegação do tipo FIFO (*First In First Out*) que armazena as informações sobre os quatro últimos objetos detetados. O valor para o tamanho do histórico foi escolhido de forma a permitir englobar o maior número possível de objetos detetados, sem que fosse comprometida a rapidez de execução do algoritmo devido às operações necessárias para a determinação da localização. Valores inferiores resultam

num maior grau de incerteza devido à associação entre três ou menos objetos ser mais comum entre regiões vizinhas.

O processo é executado sempre que ocorre uma nova detecção, verificando a possibilidade de uma nova localização. Num primeiro passo do processo (a) são criados valores de peso para todos os objetos referência inseridos no SIG sendo atribuídos valores segundo critérios relativos às informações estimadas do vetor para deslocamento, caso exista, e piso que se situam, realçando os objetos referência, i.e. com maior viabilidade para detecção. De seguida é (b) calculado por cada uma das regiões do SIG (corredores, salas, casas de banho, áreas de estudo, etc.) um valor de voto tendo por base a distribuição quantitativa de objetos, idênticos aos armazenados no histórico de navegação, por cada uma das regiões e seus respetivos vizinhos.

Essa análise é posteriormente sujeita a um sistema de pesos ponderados que tem por base as informações retiradas do cálculo dos pesos dos objetos (calculados no passo anterior (a)), que sejam iguais ao último detetado e que pertençam à região analisada, conjuntamente com as informações de posição, percurso de navegação e relacionamentos de proximidade, na região, entre objetos do histórico que partilhem o mesmo identificador. Findo o processo de votação descrito é (c) verificado qual a região que obteve maior voto, sendo a mesma escolhida como nova posição, caso possua um valor de voto que seja único. Encontrada a possível região de localização é (d) verificado se é possível dentro da mesma encontrar o posicionamento exato, analisando qual o objeto referência, igual ao detetado e pertencente à região, com maior peso. De igual modo é assumida uma nova localização coincidente com o objeto com maior expressão, caso se verifique que seu valor é único entre objetos homólogos dentro da região. Ocorrida a estimação de uma nova posição são atualizados os dados de (e) posição atual e (f) realizado o replaneamento do percurso de navegação, seguido da (g) estimação do vetor para o próximo deslocamento.

Como já referido, o sistema de localização tem por base um conjunto de procedimentos baseados nas informações de posicionamento e movimento esperado do utilizador com a função de isolar a nova posição a um conjunto de regiões ou pontos referência. Tais procedimentos são apoiados num sistema de pesos ponderados que devem ser atenciosamente calibrados, pois destes depende o bom funcionamento do sistema de localização e os quais seguidamente iremos abordar.



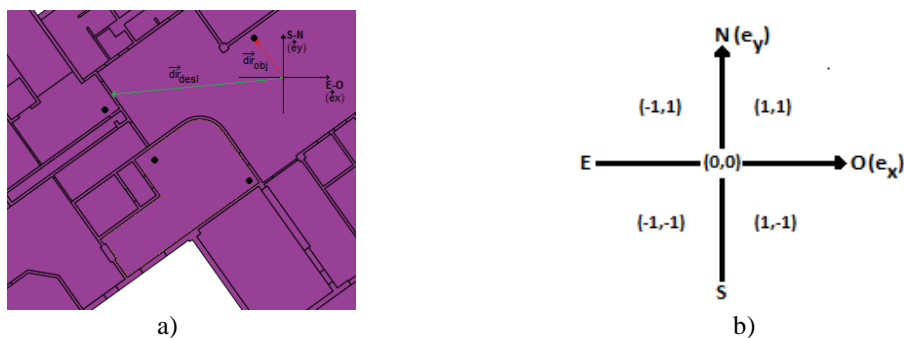
**Figura 3.3.4** – Diagrama de blocos de localização e planejamento de rota.

A primeira etapa do algoritmo, passa por destacar os objetos do SIG, com maior probabilidade para detecção através da pesagem dos mesmos. O cálculo do peso do objeto ( $P_{obj}$ ) é aplicado a cada um dos objetos do SIG tendo em consideração um peso relativo ao vetor para deslocamento ( $P_{dir}$ ) e um peso relativo à possibilidade de transição de piso ( $P_f$ ), considerando o percurso definido e a detecção recente de uma escada, resultando na fórmula  $P_{obj} = P_{dir} + P_f$ .

O Peso relativo ao vetor para deslocamento ( $P_{dir}$ ) permite atribuir maior destaque aos objetos do SIG, que estejam segundo a direção estimada para o próximo deslocamento do utilizador em relação à última posição conhecida. Entendamos como vetor para deslocamento o vetor formado entre a posição atual (última determinada antes da presente detecção) e a próxima posição de destino, porta ou escada, verificada através das informações armazenadas de rota de navegação. Na falta da existência de uma rota de navegação o vetor para deslocamento é determinado com base nas informações referentes à penúltima e última posições determinadas, considerando que o

indivíduo continua a mover-se segundo a mesma direção. Este vetor bidimensional é composto por componentes segundo os eixos Sul-Norte ( $\vec{e}_y$ ) e Oeste-Este ( $\vec{e}_x$ ) do SIG.

Para facilitar o cálculo estas componentes tomam valores unitários positivos, negativos ou nulos, consoante o sentido de orientação, isto é, considerando o eixo cartesiano e as referências geográficas Norte, Sul, Este e Oeste. Na Fig. 3.3.5a) pode ser visualizado o eixo cartesiano centrado numa região e o vetor de direção para deslocamento (a verde) e direção do objeto (a vermelho). Na figura 3.3.5b) por sua vez, encontra-se esquematizado os valores das componentes  $\vec{e}_x$  e  $\vec{e}_y$ , para cada um dos quadrantes de orientação sobre a forma de um vetor ( $\vec{v}$ ) genérico  $\vec{v}(e_x, e_y)$ . Aplicando os valores das componentes apresentados na figura 3.3.5b) aos vetores ilustrados na figura 3.3.5a) temos que,  $\overrightarrow{dir_{obj}}(e_x, e_y) = -e_x + e_y$  e  $\overrightarrow{dir_{desl}}(e_x, e_y) = -e_x - e_y$ , sendo fácil constatar que ambos os vetores se encontram segundo a mesma direção segundo o eixo Oeste-Este ( $e_x$ ) com o ponto de origem considerado.



**Figura 3.3.5** – Representação gráfica a) do vetor para deslocamento ( $\overrightarrow{dir_{desl}}$ ) e do vetor direção do objeto ( $\overrightarrow{dir_{obj}}$ ) considerando o utilizador posicionado no centro da região. Em b) encontra-se representado os valores para cada uma das componentes vertical e horizontal nos quadrantes de orientação para um vetor genérico  $\vec{v}(e_x, e_y)$ .

Comparando individualmente cada uma das componentes de os vetores da direção para deslocamento ( $\overrightarrow{dir_{desl}}$ ) e de direção do objeto ( $\overrightarrow{dir_{obj}}$ ) é possível conferir o valor de peso mediante as semelhanças encontradas, entre as direções vertical e horizontal de ambos (segundo os eixos  $\vec{e}_y$  e  $\vec{e}_x$  respetivamente). O valor máximo de peso é atribuído aos objetos que apresentam a mesma direção segundo ambos os eixos horizontal e vertical com o vetor para deslocamento, indicando-os como mais favoráveis. Aos objetos que apresentam concordância segundo uma única direção (vertical ou horizontal) relativamente ao vetor para deslocamento será conferido um valor inferior ao anterior. O menor peso será atribuído aos objetos que fiquem em sentido oposto ao

do vetor para deslocamento. Esse valor, contudo, deverá ser superior a 0 para que não sejam totalmente descartados. Considerando um valor mínimo de 1 para os objetos referência que se encontrem em oposição ao vetor para deslocamento, e restantes valores dependentes deste e segundo os critérios acima descritos, temos que:

$$P_{dir} = \begin{cases} 3, & \text{se } (\overrightarrow{dir}_{dest}(e_x) = \overrightarrow{dir}_{obj}(e_x)) \wedge (\overrightarrow{dir}_{dest}(e_y) = \overrightarrow{dir}_{obj}(e_y)) \\ 2, & \text{se } (\overrightarrow{dir}_{dest}(e_x) = \overrightarrow{dir}_{obj}(e_x)) \vee (\overrightarrow{dir}_{dest}(e_y) = \overrightarrow{dir}_{obj}(e_y)) \\ 1, & \text{se } (\overrightarrow{dir}_{dest}(e_x) \neq \overrightarrow{dir}_{obj}(e_x)) \wedge (\overrightarrow{dir}_{dest}(e_y) \neq \overrightarrow{dir}_{obj}(e_y)). \end{cases} \quad (3.3.1)$$

Contudo, considerando um ambiente *indoor* constituído por vários pisos, é necessário ponderar a possibilidade da existência de vários objetos referência iguais na mesma direção, mas situados em pisos diferentes, o que fomenta a necessidade de diferenciá-los de acordo com o piso em que se inserem. A diferenciação terá que ser sensível, no entanto, à transição entre pisos.

De forma a solucionar problemas de localização na utilização de uma filosofia multi-piso, foi criado um peso adicional, peso do piso ( $P_f$ ) com a função de realçar os objetos do SIG relativamente ao seu piso de origem. Este peso tem um valor nulo ( $P_f = 0$ ) durante a fase de auto-localização, em que a posição do indivíduo ainda não é conhecida e sempre que a última posição estimada seja coincidente com uma escada, de forma a não inibir uma possível transição de piso por parte do utilizador. Deste modo são dados pesos ( $P_f$ ) idênticos aos objetos, independentemente do piso a que pertencem, ficando posteriormente a decisão de posição para a região que reunir maior consenso através do sistema de pesos para região abordado mais à frente.

No entanto, sendo a posição conhecida e não coincidente com uma escada, é fator indicador que provavelmente o indivíduo ainda se mantém no mesmo piso, excetuando uma falha de deteção de uma escada, pelo que é necessário conferir maior destaque aos objetos situados no mesmo piso, mesmo que estes estejam segundo uma orientação oposta ao vetor para deslocamento ( $P_{dir} = 1$ ) em relação a objetos de outros pisos com o peso máximo ( $P_{dir} = 3$ ).

Deste modo, considerando  $P_f'$  como sendo relativo ao peso a atribuir a um objeto do mesmo piso (que o da posição atual), e  $P_f''$  como peso a atribuir a um objeto de um piso diferente temos que  $1 + P_f' > 3 + P_f''$ , o que resulta em  $P_f' > 2 + P_f''$ . Atribuindo um peso nulo para  $P_f''=0$  ficamos com  $P_f' > 2$ , pelo que podemos aplicar para os



objetos pertencentes ao mesmo piso que o último estimado  $P_f = P'_f = 3$  e para os objetos dos restantes pisos  $P_f = P''_f = 0$ .

Calculados os pesos dos objetos referência é iniciado um processo de votação que procura isolar as regiões com maior probabilidade para a localização. Esta votação é realizada com base numa análise individual por cada uma das regiões do SIG (corredores, salas, casas de banho, áreas de estudo, etc.), quantificando percentualmente a distribuição dos objetos, armazenados no histórico de navegação, pertencentes à região em análise ( $R_p$ ) e suas regiões vizinhas ( $R_v$ ). Por definição, uma região vizinha é considerada como tal, se tiver em comum com a região em análise uma estrutura de acesso como uma porta ou escada.

A votação para as regiões ( $V_{Reg}$ ) é calculada com base na equação  $V_{Reg} = P_{Reg} \times R_p + (P_{Reg} - 1) \times R_v$ , com  $R_p = \frac{N_p}{N_{obj}}$  e  $R_v = \frac{N_v}{N_{obj}}$ , em que  $N_p$  é o número de objetos referência do histórico de navegação possíveis encontrar na região principal (região em análise),  $N_v$  o número dos restantes objetos do histórico, não encontrados na região principal, que pertencem a regiões vizinhas da mesma e  $N_{obj}$  o número total de objetos do histórico. A cada uma das componentes percentuais é ainda aplicado um fator multiplicativo ( $P_{Reg}$ ) diferenciado, de forma a destacar as regiões com objetos detetados na região principal, quando comparadas com outras regiões que tenham igual número de objetos detetados em regiões vizinhas. O fator multiplicativo  $P_{Reg}$  é um peso individual atribuído a cada uma das regiões do SIG assente em um conjunto de critérios tais como: (a) o voto do objeto idêntico ao último detetado, pertencente à região e com maior peso ( $P_{max_{obj}}$ ); (b) a relação de vizinhança com a região estimada na última localização ( $P_{pos}$ ), caso exista; (c) a relação de vizinhança com o percurso planeado para navegação ( $P_{perc}$ ), caso exista; d) relações de proximidade entre objetos do histórico pertencentes à região e com identificador igual ( $P_{ID}$ ); resultando na seguinte função de cálculo,  $P_{Reg} = P_{max_{obj}} + P_{pos} + P_{perc} + P_{ID}$ .

O peso atribuído ao último objeto detetado ( $P_{max_{obj}}$ ) é avaliado através da análise individual de cada um dos objetos semelhantes ao reconhecido e pertencentes à região, sendo o seu valor igual ao máximo dos valores calculados, isto é  $P_{max_{obj}} = \max(P_{obj})$ , sendo o objeto “obj” pertencente à região e igual ao detetado.

Por sua vez, o peso da última posição determinada ( $P_{pos}$ ) pretende destacar positivamente as regiões que se situem na proximidade da mesma, indicando-as como mais favoráveis à nova localização quando comparadas com outras regiões mais afastadas. O seu valor de peso é calculado considerando que um objeto pertencente a uma região vizinha, à região da última localização (inclusive), segundo pelo menos uma das componentes (horizontal ou vertical) do vetor para deslocamento ( $P_{dir} = 2$ ) deve ter um peso superior a um mesmo objeto numa região mais afastada mas com o peso máximo ( $P_{dir} = 3$ ). Desta forma temos que  $2 + P_{pos} > 3$ , ou seja,  $P_{pos} > 1$ . Nos ensaios realizados foi utilizado um  $P_{pos} = 2$ .

O peso atribuído às regiões pertencentes ao percurso planeado ( $P_{perc}$ ) tem por objetivo destacar as regiões pertencentes à rota de navegação em relação às restantes, quando estas possuam objetos idênticos ao reconhecido. Para a determinação do seu valor é considerado que uma região do caminho que possua um objeto segundo uma direção oposta (devido à circulação em escadas de formato em U) ao vetor para deslocamento ( $P_{dir} = 1$ ) deverá possuir, um valor superior que permita anular a deteção de um objeto idêntico noutra região fora da rota de navegação mas segundo a direção estimada para o próximo movimento ( $P_{dir} = 3$ ), isto é  $1 + P_{perc} > 3$  resultando num valor de  $P_{perc} > 2$ , tendo sido assumido o valor  $P_{perc} = 3$ . No entanto, durante a navegação também é possível que o indivíduo entre numa divisão vizinha ao caminho planificado, sendo também necessário atribuir um peso a essas divisões. Chamemos  $P'_{perc}$  ao peso a atribuir a uma região vizinha do caminho. O valor para  $P'_{perc}$  considerando a existência de um objeto, igual ao detetado, segundo pelo menos uma das direções do vetor para deslocamento ( $P_{dir} = 2$ ) deverá ter valor inferior que um mesmo objeto no caminho traçado e segundo as mesmas condições ( $P_{dir} = 2$ ), contudo deverá ter um valor superior a um mesmo objecto, que tenha o peso máximo ( $P_{dir} = 3$ ), pertencente a uma outra região que não pertença ao caminho nem seja vizinho do mesmo o que resulta em,  $P_{perc} + 2 > 2 + P'_{perc} > 3$  ou seja  $P_{perc} > P'_{perc} > 1$ . Nos ensaios de navegação realizados foi utilizado um valor de peso de percurso para as regiões vizinhas ao caminho igual a  $P_{perc} = P'_{perc} = 1.5$ .

Por fim, existe ainda um peso relativo a possíveis associações de objetos pertencentes a um mesmo *frame* ou cenário visual ( $P_{ID}$ ) que segundo relações de proximidade, verificadas no SIG, possam ser indicadoras da posição do indivíduo. A

análise é efetuada através da verificação do posicionamento, no SIG, de objetos pertencentes ao histórico de navegação com o mesmo identificador, sendo aceites as regiões que possuam objetos com distâncias não superiores a 1 metro. Deste modo para o cálculo do seu valor é considerado que qualquer região que tenha pelo menos uma das direções ( $P_{dir} = 2$ ) de movimento, objetos cujos os critérios atrás descritos se verificarem, terá que ter maior peso que uma mesma região que possua um objeto igual ao reconhecido segundo o peso máximo de direção ( $P_{dir} = 3$ ). Deste modo temos que,  $2 + P_{ID} > 3$  o que resulta em  $P_{ID} > 1$ , tendo sido assumido o valor de  $P_{ID} = 2$ .

O método desenvolvido pretende oferecer robustez na localização mesmo nos casos em que é difícil diferenciar as regiões, com base nos objetos de referência que lhe pertencem, devido a serem muito semelhantes. Do mesmo modo através de um histórico de navegação é possível efetuar a auto-localização, sendo escolhida a região que reunir maior número de objetos do histórico presentes nas proximidades. Contudo sendo o algoritmo apoiado nas informações de direção estimadas para o próximo movimento (vetor para deslocamento), pode resultar em falhas de localização quando o mesmo inverte o sentido do seu movimento em relação ao esperado (em relação ao vetor para deslocamento), sendo esta restabelecida quando existirem objetos suficientes no histórico, ou um objeto único, que permita identificar uma nova posição. Por outro lado um reconhecimento de um falso positivo também pode interferir negativamente, necessitando que seja criado futuramente uma técnica de verificação que possa descartar falsos positivos analisando as relações de proximidade com os restantes objetos do histórico. Isto é, caso o objeto detetado não apresente relações de proximidade com os restantes ou parte deles, é inviabilizada a determinação de localização, contudo o mesmo deverá ser armazenado no histórico para cálculo futuro de uma nova localização, visto poder não ser um falso positivo.

### **3.3.1 Ordens de Navegação**

As notificações de reconhecimento e as ordens de navegação pretendem informar o indivíduo da deteção de objetos e orientá-lo no trajeto até o seu destino, através de comandos de voz ditados através de um *script* (Bennett, 2010) que utiliza o reconhecimento de voz do Windows. Sempre que ocorre o reconhecimento de um novo objeto, essa informação é comunicada ao utilizador indicando o nome do objeto (porta, escada, extintor, indicador de escadas, texto extraído por OCR, etc.) e a sua zona de

deteção na imagem capturada, considerando a imagem dividida numa matriz 3×3 como a demonstrada na Fig. 3.3.1.1.

Após efetuado o reconhecimento de um novo objeto e findo o processo de estimação da nova localização é necessário orientar o indivíduo, informando-o em que direção se deverá deslocar para que possa encontrar a porta ou escada que dá acesso à próxima região, segundo o caminho planeado (as informações do caminho são armazenadas num vetor, possibilitando saber qual a próxima estrutura de acesso (porta ou escada) e qual a próxima região de acesso). Sendo o sistema vocacionado para pessoas com dificuldades visuais, a orientação deve ser de fácil interpretação e execução. Uma das mais conhecidas formas de orientação é baseada nos ponteiros do relógio devido à sua facilidade de interpretação, bem como ser mundialmente reconhecida.

	Esquerda	Centro	Direita
Superior			
Centro			
Inferior			

**Figura 3.3.1.1** – Representação matricial das zonas de deteção de uma imagem.

A tarefa de orientação tem alguma complexidade acrescida, visto não ser conhecida a orientação do indivíduo relativamente aos pontos cardeais Norte, Sul, Oeste e Este. Uma forma de estimar a orientação é através da análise do vetor diretor de movimento, assumindo que o indivíduo se encontra de frente para o objeto. Utilizando o vetor diretor como referência de orientação e o sistema horário, v.d. figura 3.3.1.2a), podemos fornecer coordenadas de movimento, segundo os ponteiros do relógio.

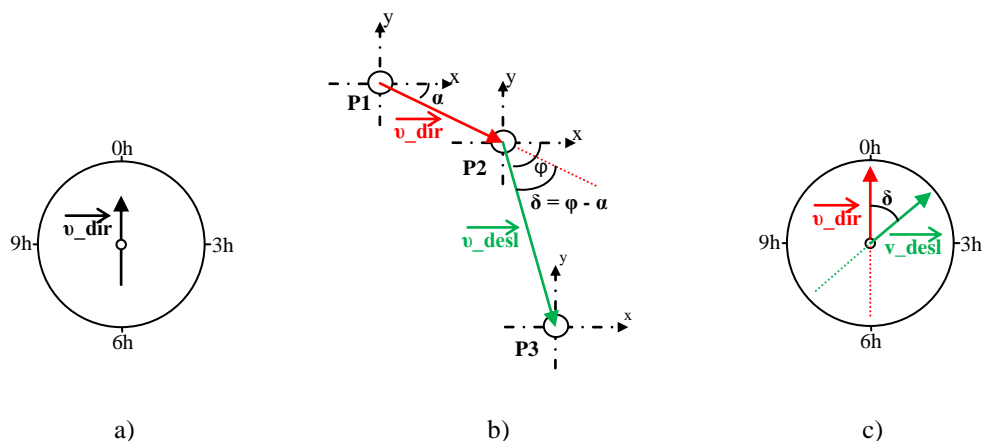
Para determinar em que direção fica o próximo destino é necessário recorrer ao vetor diretor e ao vetor para deslocamento de forma a verificar qual a diferença de ângulo que existe entre ambos. O vetor diretor caracteriza o movimento realizado pelo utilizador desde a última posição até à posição atual, enquanto o vetor para deslocamento caracteriza o movimento esperado entre a posição atual e o próximo ponto de destino (porta ou escada) verificado no caminho planeado. Na figura 3.3.1.2b), encontra-se esquematizado um vetor diretor compreendido entre os pontos P1 (última posição) e P2 (posição atual) e um vetor para deslocamento constituído pelos pontos P2 (posição atual) e P3 (próxima posição objetivo).

Considerando a diferença angular ( $\delta$ ) entre o vetor para deslocamento (com ângulo  $\varphi$ ) e o vetor diretor de movimento (com ângulo  $\alpha$ ), calculado segundo  $\delta = \varphi - \alpha$ , é possível conhecer qual a direção do próximo objetivo, tendo em conta o movimento anteriormente realizado. Conhecido o ângulo ( $\delta$ ) a que fica o próximo objetivo é necessário efetuar a sua conversão para um sistema de coordenadas horárias, v.d. Fig. 3.3.1.c), considerando para tal o vetor diretor a apontar para as 0 horas, Fig. 3.3.1.2a), e a seguinte fórmula de tradução:

$$\text{hora}(\delta) = \begin{cases} \frac{12 \times (360^\circ - \delta)}{360^\circ}, & \delta > 0 \\ \frac{12 \times |\delta|}{360^\circ}, & \delta \leq 0. \end{cases} \quad (3.3.1.1)$$

Este método, baseado no cálculo de direção pelos vetores diretor de movimento e para deslocamento, só é aplicável caso já tenha sido determinada a posição e o vetor para deslocamento. Caso ainda não seja conhecida a localização a ordem de comando é unicamente a de continuar o seu movimento, até que consiga ser estimada a posição.

O sistema de orientação também só é válido caso haja alguma rota definida até a um destino, pois caso contrário perde qualquer sentido o fornecimento de ordens de movimento.



**Figura 3.3.1.2** – Processo de determinação da orientação para movimento. a) Vetor diretor referenciado ao sistema horário de coordenadas. b) Cálculo do ângulo ( $\delta$ ) entre o vetor diretor de movimento ( $\alpha$ ) e o vetor para deslocamento ( $\varphi$ ) e c) representação considerando um referencial de coordenadas baseado no sistema horário.

## 4. Resultados

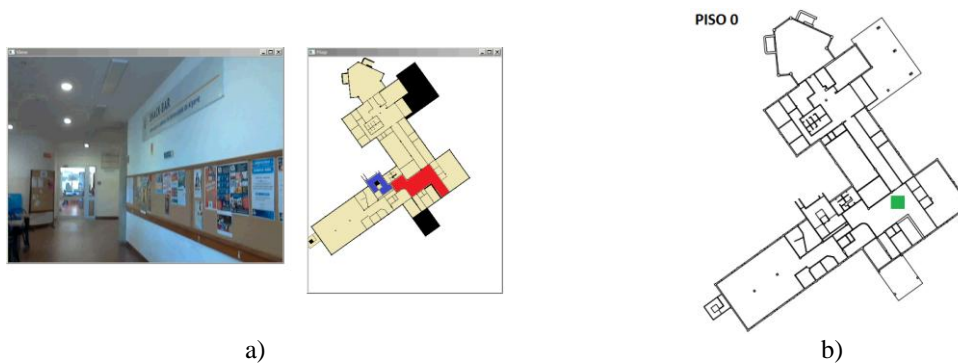
De forma a comprovar o funcionamento do sistema, englobando os tópicos abordados para reconhecimento e localização, foram desenvolvidos alguns testes no Instituto Superior de Engenharia, os quais seguem sucintamente descritos no subcapítulo seguinte.

### 4.1 Testes

Foram realizados vários testes em várias localizações dentro do ISE, aqui apresentamos três destes testes, devido a sua complementaridade, e discutimos os respetivos resultados. O primeiro teste visa ensaiar a auto localização, partindo de um ponto desconhecido para o sistema. O segundo ensaio, partindo de uma posição já determinada, pretende ilustrar o funcionamento da estimação de posição em várias etapas de reconhecimento de objetos e respetivas ordens de movimento comunicadas ao utilizador para o auxílio na navegação. O terceiro teste, é compreendido por um desvio intencional do utilizador de forma a analisar o desempenho da recuperação de posição.

Para os testes acima mencionados, foi utilizado um computador com um processador *Intel Core I3* a 2.13GHz e 4GB de memória DDR3 e uma câmara *Logitech C270* com resolução de 3 Megapixel para a captura de imagens de dimensões 640x480 pixel.

As imagens como a Fig. 4.1.1a) aqui utilizadas para ilustração são retiradas diretamente da aplicação desenvolvida. A aplicação retorna como “*outputs*” a imagem capturada com os objetos identificados (Fig. 4.1.1a) à esquerda), se houver, e um mapa produzido automaticamente (Fig. 4.1.1a) à direita), onde são registradas as informações de posição (marcado a vermelho a região da posição atual) e do caminho planeado (marcadas a azul as regiões que fazem parte do caminho), no mesmo piso que a posição atual determinada. Por sua vez, a imagem do mapa SIG apresentado na figura 4.1.1b) é meramente auxiliar e aqui utilizada para complementar a informação e fornecer dados mais detalhados.

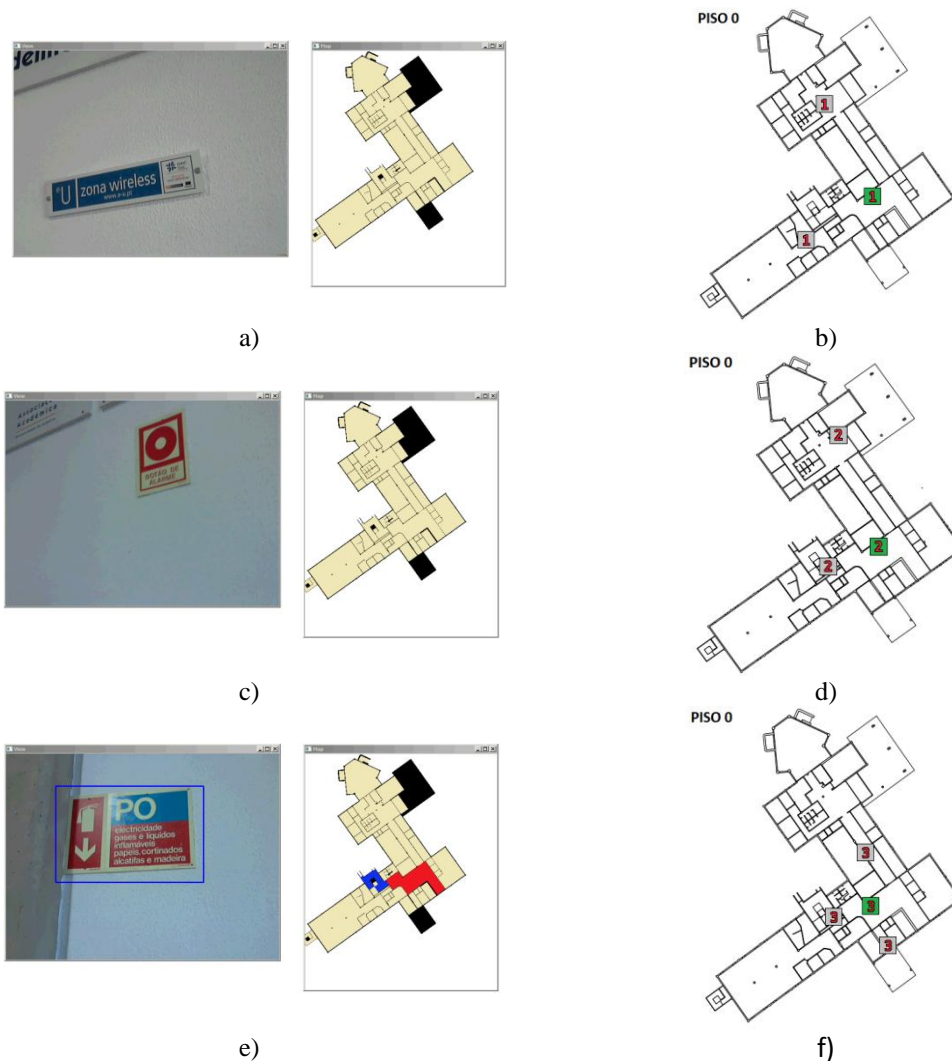


**Figura 4.1.1** – Exemplo de a) imagem retirada do sistema de localização em tempo real desenvolvido com a imagem capturada à esquerda e as informações de localização e navegação à direita. Em b) um mapa auxiliar para complementar a informação não exibida pelo sistema. O quadrado verde indica a posição real do utilizador.

O *primeiro teste* realizado tem como objetivo demonstrar a localização automática, partindo de um ponto desconhecido, através dos objetos referência reconhecidos. Na fase inicial o sistema desconhece qual a sua localização atual, pelo que é comunicado verbalmente ao indivíduo que deverá continuar o seu movimento e tentar encontrar pontos para referência. Na figura 4.1.2a) à esquerda encontra-se o primeiro objeto reconhecido, através de OCR, com a informação textual de “zona wireless”. Após o reconhecimento é executado o processo de localização na tentativa de obter a posição correta. No entanto este objeto como pode ser visto no SIG representado na Fig. 4.1.2b), respetivo ao piso 0 do ISE, encontra-se em 3 regiões distintas (assinaladas nos quadrados com o número 1, em que o quadrado verde é a posição real do utilizador), sendo impossível determinar em qual das regiões, com base nos dados até aqui adquiridos, o indivíduo se encontra. Não tendo dados adicionais, não é comunicada nenhuma ordem ao utilizador, esperando que este prossiga o seu movimento.

Durante o movimento executado é reconhecido um segundo objeto, representado na Fig. 4.1.2c) à esquerda, identificando um sinalizador com a informação textual (extraída por OCR) “botão de alarme”. Realizada uma nova tentativa de obter localização é verificado que existem duas regiões, como pode ser visualizado na figura 4.1.2d), com a mesma combinação de objetos encontrada até ao momento, isto é, as informações de “zona wireless” (assinalado pelo número 1 na Fig. 4.1.2b)) e “botão de alarme” (assinaladas pelo número 2 na Fig. 4.1.2d)). Deste modo apesar da deteção de mais um objeto de referência é ainda impossível conhecer a posição do indivíduo, aguardando a resolução da localização por deteções futuras.

O terceiro objeto (Fig. 4.1.2e) à esquerda delimitado por um retângulo azul) reconhecido é um sinalizador de extintor, obtido pela extração e comparação de pontos com recurso ao SURF. Neste momento recorrendo novamente ao histórico de objetos identificados, é verificado que existe uma combinação única numa das regiões (combinação dos objetos assinalados na Fig. 4.1.2 b), d) e f) por os números 1,2 e 3 respetivamente), pelo que será validada como posição atual a mesma posição que a do objeto (sinalizador de extintor) dentro da região do piso 0 (a região da posição atual é assinalada a vermelho no SIG construído pela aplicação e ilustrado na Fig. 4.1.2e) à direita da imagem capturada). Após ter sido efetuada a localização, é criada a rota de navegação tendo como ponto de destino um gabinete no piso 2 (a rota de navegação é assinalada a azul no SIG construído e representado na Fig. 4.1.2e) à direita da imagem capturada).

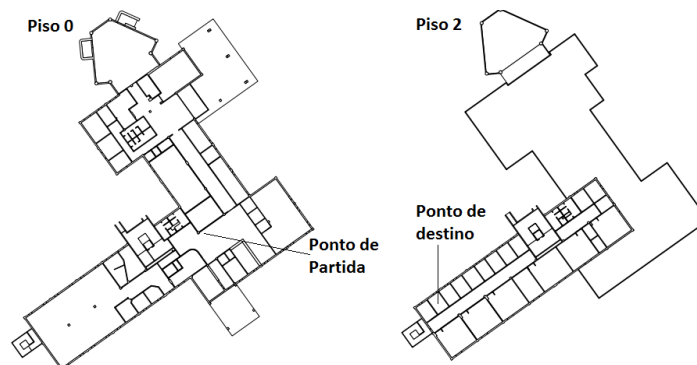


**Figura 4.1.2** – Figura ilustrativa da sequência de objetos reconhecidos e suas posições no SIG, na fase de auto localização.



Como verificado, a localização num meio interior tendo por base objetos comuns a diversas divisões pode ser difícil e só possível se forem armazenados a sequência de objetos reconhecida, na tentativa de através de associações determinar a posição correta.

O *segundo teste* é pretendido, partindo da posição já determinada no primeiro teste do piso 0 e tendo como ponto de destino um gabinete no piso 2 (v.d. Fig. 4.1.3), aferir o desempenho do sistema na orientação do utilizador até que seja atingido um piso diferente do piso 0.

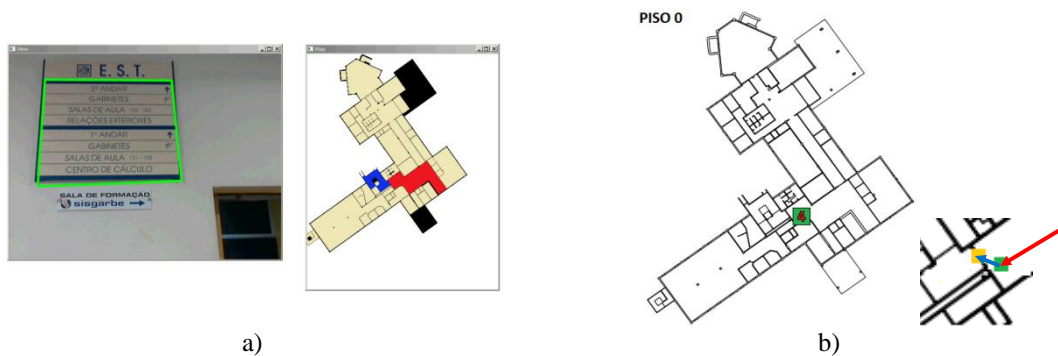


**Figura 4.1.3** – Representação, à esquerda, do ponto de partida no piso 0 e à direita a representação do ponto de destino no piso 2.

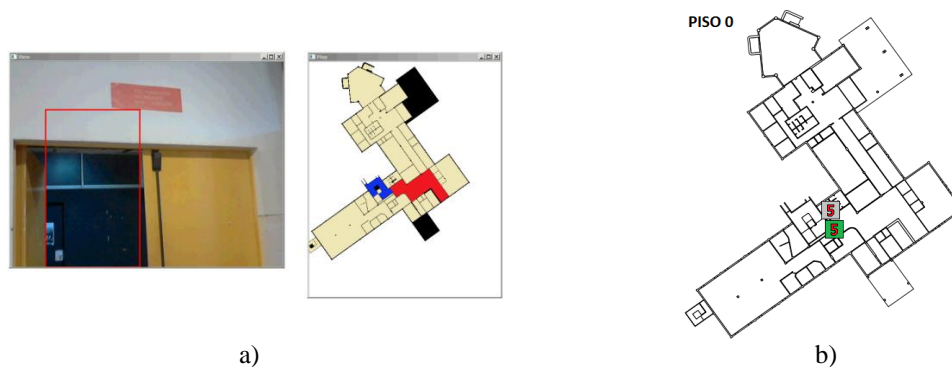
No segundo teste o primeiro objeto detetado foi um retângulo (Fig. 4.1.4a) à esquerda destacado a verde) com a informação adicional “salas de aula” por OCR, verificando no SIG uma nova posição dentro da mesma região atual, marcada através de um quadrado verde numerado com o número 4 no mapa representativo na figura 4.1.4b). Neste caso é conhecido o movimento que o utilizador realizou, pelo que é possível fornecer informações mais detalhadas quanto ao próximo movimento a realizar. Tendo em conta o vetor diretor (seta vermelha Fig. 4.1.4b) à direita entre a última posição conhecida e a agora determinada, representada através de um quadrado verde) e o vetor para deslocamento (seta azul Fig. 4.1.4b) à direita entre a posição determinada e uma porta aqui representada por um quadrado amarelo) até ao próximo ponto de destino é comunicado ao utilizador que se deve deslocar para as suas 2 horas e procurar uma porta perto de si.

Durante a deslocação foi detetada uma porta (delimitada pelo retângulo vermelho na fig. 4.1.5a) à esquerda), no entanto segundo a mesma direção que o vetor deslocamento existe mais que uma porta, como ilustrado pelo mapa representado na figura 4.1.5b) pelos quadrados com o número 5 (porta capturada representada pelo quadrado verde

com o número 5), pelo que o sistema de localização dá como indeterminada uma nova posição e nenhuma ação é comunicada.



**Figura 4.1.4** – Imagem com a) identificação de um retângulo (destacado a verde) à esquerda e respetivo mapa SIG, com as informações de posição (região assinalada a vermelho) e caminho planeado (assinalado a azul) à direita. Em b) no mapa, à esquerda, representação da posição real (quadrado verde) e posição do objeto detetado (indentificado com o numero 4). Na mesma imagem em b), à direita, detalhe da posição estimada (quadrado verde) e da próxima posição objetivo (quadrado amarelo) bem como a representação do vetor diretor (seta vermelha) e do vetor para deslocamento (seta azul).

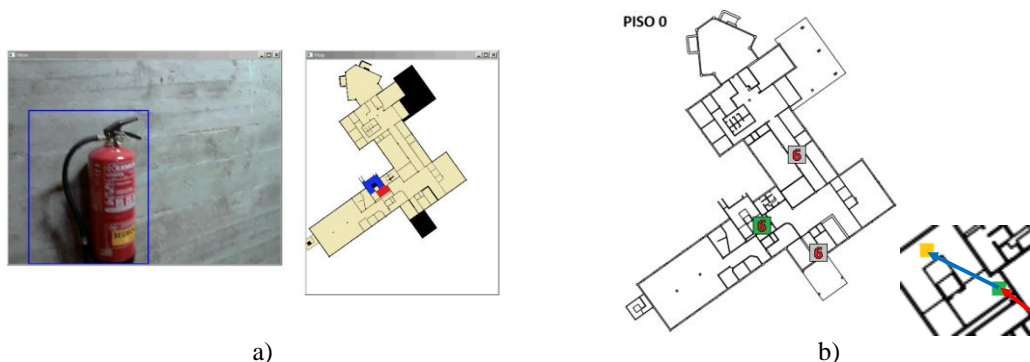


**Figura 4.1.5** – Imagem com a) identificação de uma porta (destacada a vermelho) à esquerda e respetivo mapa SIG, com as informações de posição (região assinalada a vermelho) e caminho planeado (assinalado a azul) representados à direita. Em b) representação da posição das portas candidatas, identificadas pelos quadrados numerados com o número 5, sendo que o quadrado verde é representativo da posição real da porta detetada.

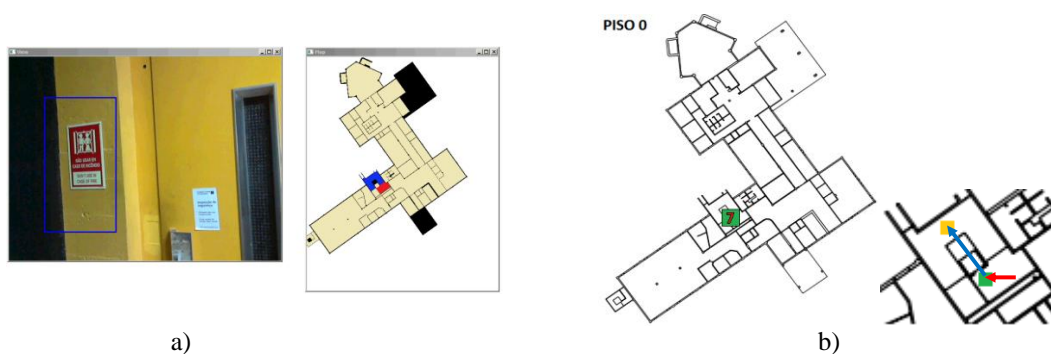
Continuando a navegação é reconhecido um extintor (fig. 4.1.6a) à esquerda), através do SURF, resultando numa nova estimacão de posição coincidente com o extintor representado pelo quadrado verde e o número 6 no SIG da figura 4.1.6b), pertencente a uma nova região vizinha da anterior. No mapa representado pela figura 4.1.6b) é possível verificar que existem vários vizinhos da região anterior que possuem um extintor (quadrados cinza numerados com o número 6), no entanto a nova posição foi determinada tendo por base a técnica de pesos utilizada que permitiu atribuir um maior destaque à nova região e objeto segundo as informações respeitantes quanto ao caminho planeado e vetor para deslocamento esperado. Através da diferença calculada entre o vetor diretor (movimento realizado) e o vetor para deslocamento (movimento a

realizar), representados na figura 4.1.6b) à direita, é comunicado que deve ser procurada uma escada às 11 horas. A informação de movimento comunicada no entanto é errada, na verdade o próximo movimento a executar deverá ser para as 2 horas e não para as 11 horas como reportado pelo sistema. Este erro deveu-se ao facto de ter falhado a localização no passo anterior, aquando do reconhecimento da porta.

O próximo objeto detetado foi um sinalizador de elevador, mais uma vez com recurso ao SURF, identificado por um retângulo azul na figura 4.1.7a) à esquerda, na mesma região informando que deverá ser procurado umas escadas às 2 horas (v.d Fig. 4.1.7b) à direita).

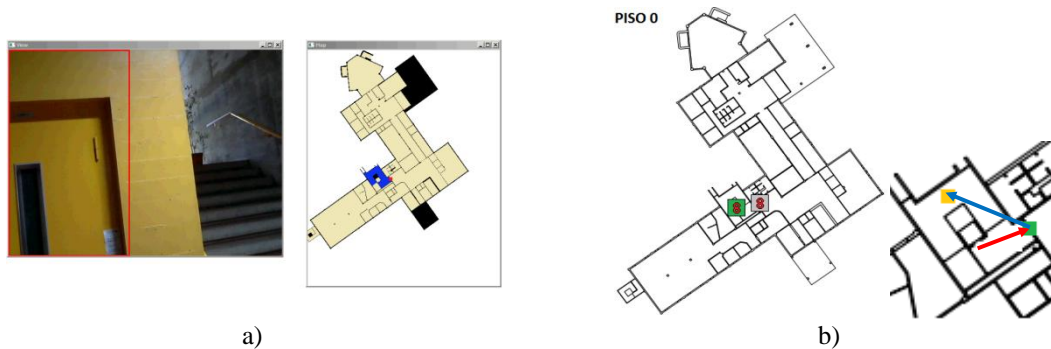


**Figura 4.1.6** – Imagem com a) identificação de um extintor (destacada a azul) à esquerda e respetivo mapa SIG, com as informações de posição (região assinalada a vermelho) e caminho planeado (assinalado a azul) à direita. Em b) à esquerda, encontra-se representada a posição real (quadrado verde com o número 6) e a posição de outros extintores candidatos (identificados pelos quadrados cinza com o número 6). Na mesma imagem à direita encontra-se a representação do vetor director de movimento (seta vermelha) e do vetor para deslocamento (seta azul), sendo o quadrado verde coincidente com a posição estimada e o quadrado amarelo com a próxima posição objetivo, neste caso uma escada.



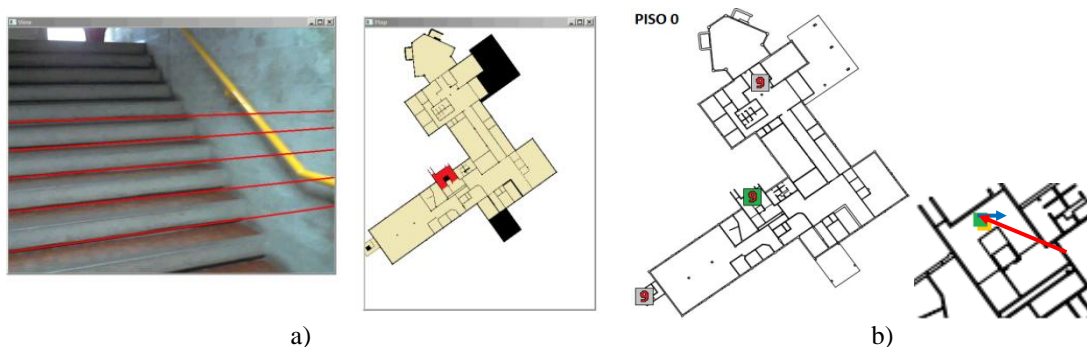
**Figura 4.1.7** – Imagem com a) identificação de um sinalizador de elevador (destacado a azul) à esquerda e respetivo mapa SIG, com as informações de posição (região assinalada a vermelho) e caminho planeado (assinalado a azul) à direita. Em b) à esquerda, encontra-se representada a posição real do objeto identificado, através do quadrado verde com o número 7. Na mesma imagem à direita encontra-se a representação do vetor director de movimento (seta vermelha) e do vetor para deslocamento (seta azul), sendo o quadrado verde coincidente com a posição estimada e o quadrado amarelo com a próxima posição objetivo, neste caso uma escada.

Durante o movimento executado, à procura de uma escada, foi identificado uma porta, circunscrita através de um retângulo vermelho na figura 4.1.8a) à esquerda, pertencente ao elevador. No entanto a porta do elevador (assinalada pelo quadrado verde numerada com o número 8 no SIG da Fig. 4.1.8b) à esquerda) não se encontrava representada no Sistema de Informação Geográfica, tendo sido assumida como posição atual uma outra porta na direção do movimento (assinalada pelo quadrado cinza numerado com o número 8 do mesmo SIG da fig. 4.1.8b)). Com a determinação de uma nova posição (destacada a vermelho no SIG da fig. 4.1.8a) à direita), foi criado um novo caminho para navegação (regiões assinaladas a azul no SIG da fig. 4.1.8a) à direita) e fornecidas informações de movimento reportando ao utilizador que deveria corrigir o movimento para as suas 7 horas. Devido a ter sido efetuado uma estimativa de posição errada, todas as informações compiladas acabam por ser igualmente erradas.



**Figura 4.1.8** – Imagem com a) identificação de uma porta (destacada a vermelho) à esquerda e respetivo mapa SIG, com as informações de posição (região assinalada a vermelho) e caminho planeado (assinalado a azul) à direita. Em b) no mapa à esquerda, encontra-se representada a posição da porta detetada, identificada pelo quadrado verde com o número 8 e a posição da única porta candidata na região (segundo o SIG), identificada pelo quadrado cinza. Na mesma imagem à direita encontra-se a representação do vetor director de movimento (seta vermelha) e vetor para deslocamento (seta azul), sendo o quadrado verde coincidente com a posição estimada e o quadrado amarelo com a próxima posição objetivo, neste caso uma escada.

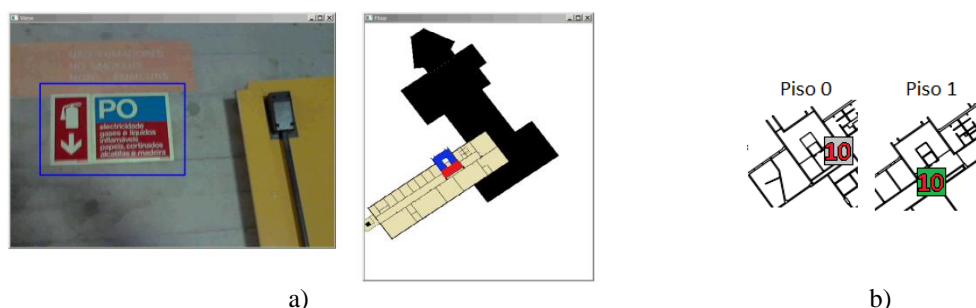
Rodando a câmara foi detetado as escadas pretendidas (identificada pelas linhas vermelhas da fig. 4.1.9a) à esquerda), sendo atualizada a posição atual (v.d. região destacada a vermelho no SIG da fig. 4.1.9a) à direita) do indivíduo para as escadas que dão acesso ao piso 1, representadas pelo quadrado verde com o número 9 no SIG da figura 4.1.9b). É ainda calculado um novo caminho para navegação e fornecidas novas ordens de movimento para seguir pelas escadas e encontrar novamente umas escadas às 5 horas (v.d. figura 4.1.9b) à direita do SIG), escadas essas pertencentes ao piso 1 e que dão acesso ao piso 2.



**Figura 4.1.9** – Imagem com a) detecção de uma escada (identificada pelas linhas vermelhas) na imagem à esquerda e respetivo mapa SIG, com as informações de posição (região assinalada a vermelho) na imagem à direita. Em b) representação real da posição das escadas detetadas (quadrado verde com o número 9) e posição de outras escadas candidatas nas regiões (identificadas pelos quadrados cinza com o número 9) à esquerda. Na mesma imagem à direita encontra-se a representação do vetor director de movimento (seta vermelha) e de deslocamento (seta azul), sendo o quadrado verde coincidente com a posição estimada e o quadrado amarelo com a próxima posição objetivo.

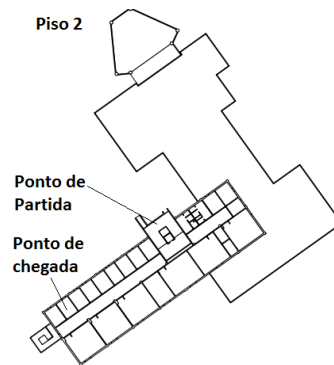
Seguindo as escadas, foi detetado através do SURF, um sinalizador de extintor (isolado pelo retângulo azul na fig. 4.1.10a) à esquerda). O algoritmo de localização atribui-o corretamente a nova localização como pertencente ao piso 1 (objeto identificado pelo quadrado verde com número 10 no SIG do piso 1 na fig. 4.1.10b) à direita). A transição foi concretizada devido ao peso atribuído às regiões pertencentes ao caminho, o que validou no SIG a região e respetivo ponto de referência referente a esta posição.

Neste teste foi possível verificar o funcionamento do sistema durante a navegação do utilizador entre as diferentes divisões de um piso e entre pisos, concluindo-se que a sua eficácia é influenciada diretamente pelos valores estimados do vetor para deslocamento, podendo ser indicadas coordenadas de movimento erradas quando existe falhas de determinação de posição ou objetos referência mal introduzidos.



**Figura 4.1.10** – Imagem com a) identificação de um sinalizador de extintor (delimitado pelo retângulo azul) na imagem à esquerda e respetivo mapa SIG, com as informações de posição (região assinalada a vermelho) e caminho planeado (região assinalada a azul) na imagem à direita. Em b) no mapa à direita, representação da posição do sinalizador reconhecido no piso 1 (quadrado verde com o número 10) e posição de um mesmo sinalizador candidato mas pertencente ao piso 0 (identificado pelo quadrado cinza com o número 10) à esquerda.

No *terceiro teste* efetuou-se um desvio intencional, da rota de navegação estabelecida, de forma a verificar a recuperação de posição por parte do sistema. O teste foi realizado tendo como ponto de partida a escada do piso 2 do ISE e ponto de destino o gabinete 172 do mesmo piso. Na figura 4.1.11 encontram-se ilustrados os pontos de partida e chegada referidos.

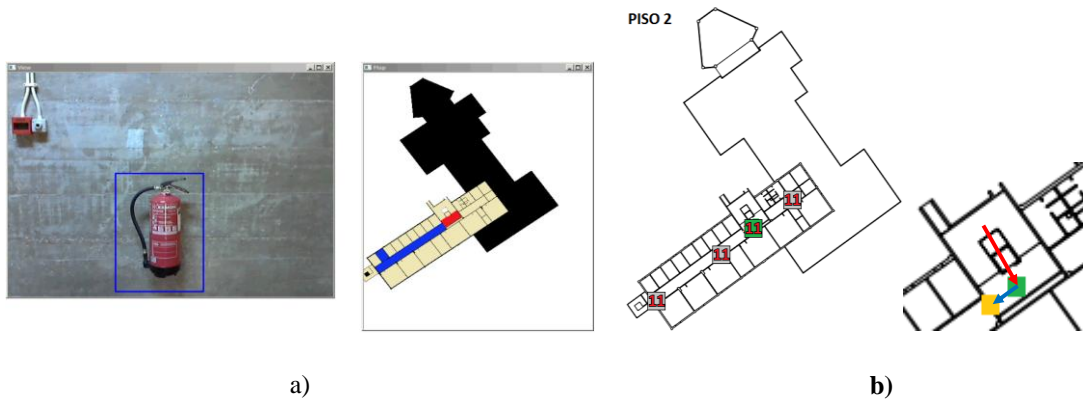


**Figura 4.1.11** – Representação do ponto de partida e destino no piso 2, assumidos para o teste.

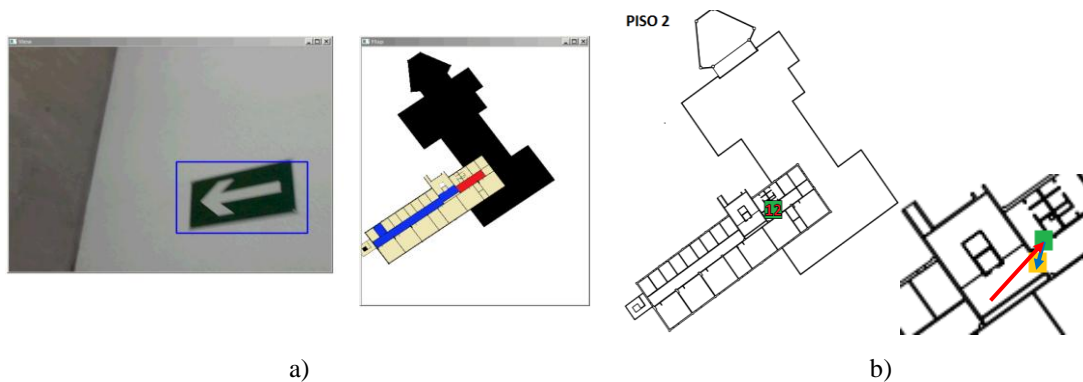
O primeiro objeto identificado no ensaio foi um extintor de incêndio, delimitado pelo retângulo azul na figura 4.1.12a) à esquerda. Através do SIG, representado na figura 4.1.12b) à esquerda, é possível verificar que existem 4 extintores (posições assinaladas pelos quadrados numerados com o número 11) em 3 regiões distintas, contudo, foi escolhido corretamente como nova localização a posição coincidente com o extintor, assinalado com quadrado verde número 11 no SIG (fig. 4.1.12b) à esquerda), presente no *hall* de entrada do piso. Determinada a região foi calculada a rota de navegação até ao objetivo (caminho marcado a azul na figura 4.1.12a) à direita) e dadas indicações da posição atual e instruções para mover para as 3 horas até encontrar uma porta.

No entanto, para efeitos de teste, foram ignoradas as ordens de navegação comunicadas e foi realizado o movimento oposto, sendo encontrado de seguida um sinalizador de indicação de saída, circunscrito pelo retângulo azul na figura 4.1.13a) à esquerda. Neste caso foi validada a posição numa das regiões vizinhas da anterior e dadas ordens para corrigir o movimento para procurar uma porta próxima às 4 horas.



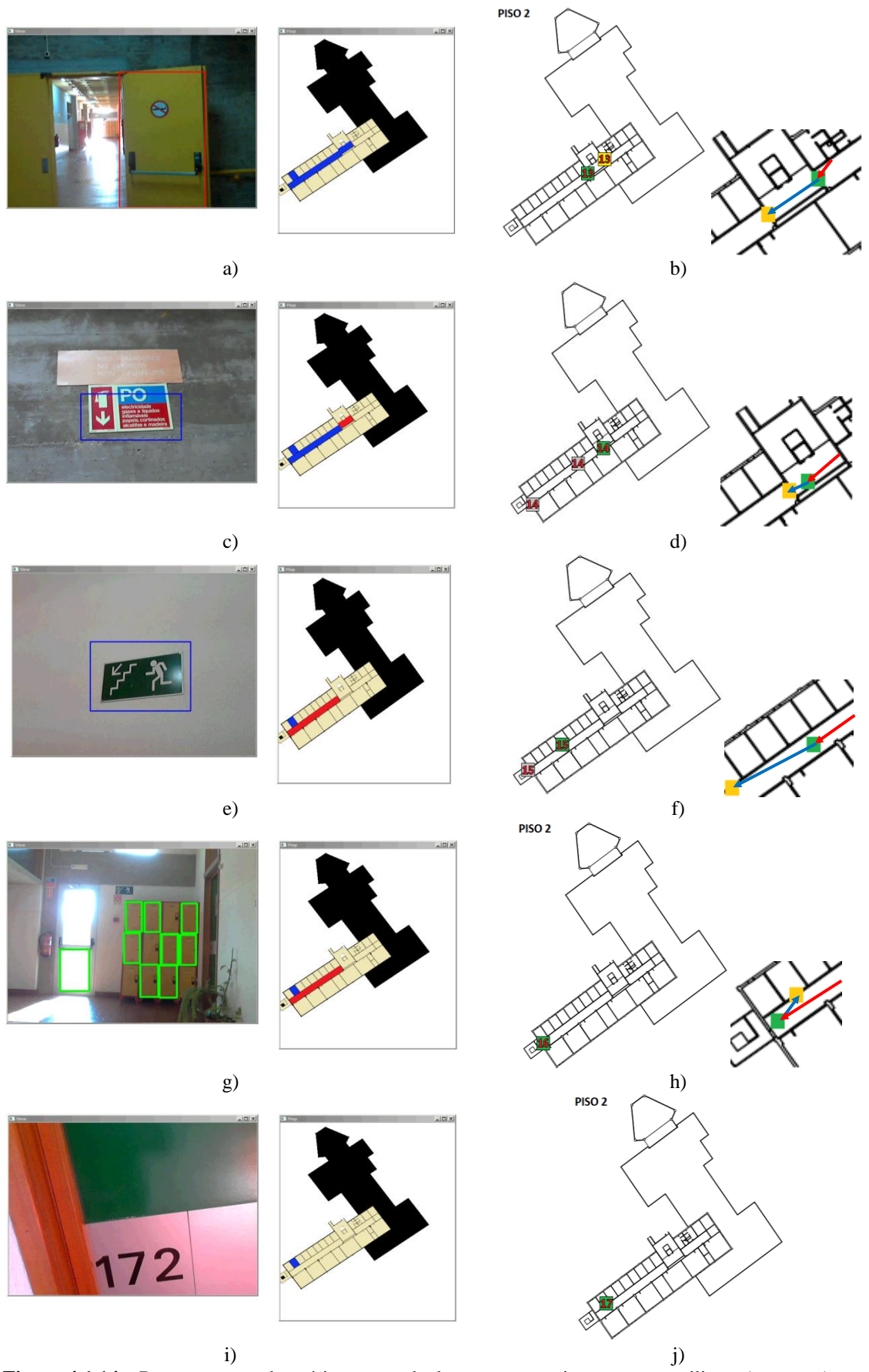


**Figura 4.1.12** – Imagem com a) identificação de um extintor (delimitado pelo quadrado azul) à esquerda e respetivo mapa SIG, com as informações de posição (região assinalada a vermelho) e caminho planeado (assinalado a azul) à direita. Em b) à esquerda, encontra-se representada a posição real do objeto identificado, através do quadrado verde com o número 11 e outros candidatos, representados pelos quadrados cinza. Na mesma imagem à direita encontra-se a representação do vetor director de movimento (seta vermelha) e vetor para deslocamento (seta azul), sendo o quadrado verde coincidente com a posição estimada e o quadrado amarelo situado na porta referente à próxima posição objetivo.



**Figura 4.1.13** – Imagem com a) identificação de um sinalizador de saída (delimitado pelo quadrado azul) à esquerda e respetivo mapa SIG, com as informações de posição (região assinalada a vermelho) e caminho planeado (assinalado a azul) à direita. Em b) à esquerda, encontra-se representada a posição real do objeto identificado, através do quadrado verde com o número 12. Na mesma imagem à direita encontra-se a representação do vetor director de movimento (seta vermelha) e vetor para deslocamento (seta azul), sendo o quadrado verde coincidente com a posição estimada e o quadrado amarelo situado na porta referente à próxima posição objetivo.

Realizado o movimento foi detetada uma porta (no retângulo vermelho da Fig. 4.1.14a) à esquerda), tendo sido assumido pelo sistema que se trataria da porta procurada (assinalada pelo quadrado amarelo com o número 13 no SIG da Fig. 4.1.14b)). No entanto na realidade, ao contrário do assumido pelo sistema, a porta detetada pertence a outra região, que se encontra referenciada pelo quadrado verde com o número 13 no SIG representado na figura 4.1.14b).



**Figura 4.1.14** – Representação das várias etapas de deteção e respetivos mapas auxiliares (ver texto).



De seguida foi reconhecido um sinal de extintor, Fig. 4.1.14c), indicando que se encontrava numa nova posição e que deveria prosseguir movimento para a 1 hora. Na continuação foi encontrado um sinal de indicação de proximidade de escadas, Fig. 4.1.14e), neste caso como pode ser visualizado no SIG apresentado na figura 4.1.14f), existem 2 sinalizadores de escadas pertencentes à mesma região, não sendo possível determinar qual deles foi detetado na realidade. Contudo, ambos pertencem à mesma região e como tal é garantido que o utilizador se encontra nessa região, apesar de não ser conhecida a posição exata dentro dela é assumido como posição atual o centro da região e reportado ao utilizador para continuar a deslocação e encontrar uma porta às suas 0 horas. Seguindo as instruções fornecidas foi encontrado um cacifo, Fig. 4.1.14g), sendo indicado que deveria inverter o movimento para as 5h, até encontrar uma porta com a numeração, extraída por OCR - “172”, Fig. 4.1.14i), comunicando por fim que havia chegado ao destino pretendido.

Este ensaio permitiu aferir a capacidade do sistema se adaptar a um possível desvio do utilizador da rota planeada, informando o mesmo de qual a direção correta do destino pretendido. Apesar de ter havido momentos em que o sistema assumiu posições erradas em relação à realidade foi possível auxiliar o utilizador a atingir o objetivo. Um filme detalhado destes testes pode ser visualizado em <http://www.youtube.com/watch?v=WoBcsnEGiB8&feature=youtu.be>.

## **4.2 Discussão de resultados**

Os testes permitiram comprovar o funcionamento do sistema de localização em tempo real desenvolvido, focando a navegação num meio interior, através dos vários desafios colocados visando a localização automática, a navegação entre pisos e recuperação de posição quando o utilizador se desvia de uma rota pré-determinada. Através destes testes foi claro que a localização em meios interiores é extremamente delicada e difícil, principalmente quando os objetos para referência são comuns a várias áreas, aspeto o qual característico na maioria dos edifícios públicos. Esta homogeneidade conduz a uma localização imprecisa, sendo muitas vezes impossível determinar a posição correta devido à semelhança que as divisões apresentam em termos de objetos e sinalética. Para que seja permitido uma navegação nestes meios é necessário criar métodos complementares que têm por base o movimento do utilizador, para um determinado objetivo, sendo assim possível isolar um grupo de candidatos para

estimar a localização. No entanto esta abordagem pode resultar em determinações erradas e conseqüentemente em ordens de movimento que afastem ainda mais o indivíduo do seu objetivo final. Através dos resultados obtidos, também foi claro a necessidade de ser criada alguma robustez aquando da deteção de falsos positivos de forma a não serem determinadas posições erradas. Uma das resoluções possíveis futuramente será incluir análises à própria arquitetura das divisões, de forma a permitir diferenciá-las e servir de complemento ao reconhecimento dos objetos.

Com vista a uma futura otimização e de forma a identificar detalhadamente os processos que necessitam de melhorias foi criado um registo de tempos de execução para aferir o tempo médio e máximo em segundos despendidos em cada um dos processos que envolvem o sistema desenvolvido, os quais encontram-se descritos na tabela 4.2.1 Os tempos aqui mencionados foram obtidos durante os testes descritos na secção anterior. Ao analisar a tabela é constatado que os processos que apresentam tempos mais longos de execução são os responsáveis pelo fornecimento de informações ao utilizador, tanto através das notificações através do sintetizador de voz como pela construção gráfica do SIG com as informações de localização e caminho planeado. As notificações de voz são encarregues de fornecer as informações dos objetos detetados, posição em que os mesmos foram detetados na imagem (à esquerda, à direita, no centro, em baixo, em cima, etc.), localização atual do utilizador (Bar, Sala nº x, Corredor y, etc.) e ainda as coordenadas de movimento, o que no seu todo justifica os tempos obtidos. Este procedimento pode ser acelerado, “acelerando” a velocidade da fala, bem como pode ser substituído (pelo menos parcialmente) por outro interface, como por exemplo uma pulseira no pulso ou no tornozelo de dê pequenos choques elétricos em função da direção onde se pretenda efetuar o movimento (du Buf et al., 2011).

A construção gráfica do SIG é realizada por intermédio de uma aplicação externa “*gdal\_rasterize*” pertencente à biblioteca do GDAL (GDAL, 2012), que permite criar a partir de uma mapa de vetores um mapa em formato de imagem (*raster*). Os tempos registados são justificados pelo facto de ser necessário chamar faseadamente a aplicação para realizar a coloração de cada uma das regiões para a identificação gráfica da posição atual e caminho calculado, junto ao facto de todas estas operações ocorrerem por intermédio do disco rígido que apresenta limitações mais restritas do que se o mesmo processo fosse executado através da memória. Contudo, para um utilizador invisual,

esta visualização não faz sentido, ela está a ser usada neste momento apenas para ilustrar a “prova do conceito”.

**Tabela 4.2.1** – Tempos médios e máximos em segundos despendidos em cada um dos processos que envolvem o sistema desenvolvido.

<b>Processo</b>	<b>Tempo médio (s)</b>	<b>Tempo máximo (s)</b>
Contextualização visual	0.062s	0.071s
Reconhecimento de portas	0.180s	0.282s
Reconhecimento de escadas	0.157s	0.218s
Reconhecimento de retângulos	0.216s	0.437s
Reconhecimento de caracteres	1.335s	4.903s
Reconhecimento de objetos (SURF)	1.495s	2.336s
Localização	0.554s	1.120s
Planeamento de rota	0.322s	0.510s
Ilustração gráfica do SIG	3.260s	8.211s
Notificações de voz	16.589s	25.194s

Dos processos encarregues pelo reconhecimento, um dos que apresentou piores performances foi o do reconhecimento ótico de caracteres, este processo é executado por uma aplicação externa, Tesseract-OCR (Tesseract-OCR, 2007), sendo passada por argumento a imagem capturada em níveis cinza após ser aplicado um filtro de realce. O tempo para processamento está relacionado com o fato de ter que efetuar o tratamento de uma imagem completa de dimensões 640x480, esperando-se melhores resultados se fosse efetuado um pré-tratamento que permitisse segmentar a imagem numa região de interesse de menores dimensões. O processo de reconhecimento de objetos por SURF, por seu lado, também apresentou tempos elevados para além do esperado, apesar de terem sido criados métodos de segmentação para isolar as áreas de potencial interesse, contudo, a sua performance está igualmente relacionada com o número de pontos para comparação e com o número de objetos para comparação na base de dados, aliado ao fato de poder existir situações em que são criadas múltiplas áreas de interesse ou uma única mas que engloba grande percentagem da área da imagem, resultando em tempos de execução mais longos. Os restantes processos para deteção de portas, escadas e retângulos apresentaram tempos bons de execução para a máquina utilizada, visto serem baseados em métodos simples e de rápida execução.

O algoritmo criado para a localização apresenta algumas limitações temporais criadas pelo número de operações que têm que executar quando são detetados objetos muito comuns no mapa, como são exemplo as portas que se encontram espalhadas nos

mapas dos 3 pisos considerados para ensaio, o que leva a um maior tempo de análise. Nos ensaios realizados foram registados tempos de processamento muito lentos, sendo processado em média 1 *frame* em cada 3 segundos, o que reforça a necessidade de melhorias significativas no desempenho geral do sistema, com foco nos algoritmos de reconhecimento de objetos com o SURF e no reconhecimento de caracteres.

Resumindo, através dos tempos de execução que cada um dos processos é de assinalar que os métodos criados para a deteção de portas, escadas e retângulos, sendo baseados em detetores de contornos, detetores de linhas e em associações matemáticas simples resultam em tempos de execução baixos. Contudo será importante referir, que esses tempos estão inteiramente associados à sensibilidade requerida na deteção dos contornos dos objetos, através do algoritmo de Canny, pois quanto maior quantidade de informação existir maior será o tempo a processá-la. Por outro lado, os métodos para reconhecimento de caracteres e objetos utilizando o SURF apresentaram tempos de execução mais preocupantes. No SURF, apesar dos métodos de segmentação criados para isolar as possíveis áreas de interesse, obteve-se tempos de execução altos, obtidos certamente devido à ocorrência de múltiplas áreas de interesse ou áreas demasiado grandes para análise, possivelmente devido aos filtros de cor estarem a abranger gamas de cor muito largas, contudo devido às deficiências de iluminação foi necessário alargar os *thresholds* de filtragem de cor de forma a aumentar a eficácia de reconhecimento de objetos. Para obter melhores performances, será de equacionar futuramente a utilização de mapas de saliência (discutidos na secção 2.3), de forma a isolar concretamente as áreas de interesse. O OCR, tal como o reconhecimento de objetos por SURF, obteve tempos de execução preocupantes. Da mesma forma será de equacionar a utilização de mapas de saliência (discutidos na secção 2.3) que permitam restringir a análise de texto unicamente a áreas de potencial interesse.

No capítulo seguinte serão referidas as conclusões obtidas nesta dissertação bem como enunciados para efeitos futuros, alguns dos trabalhos ainda necessários desenvolver para a criação de um sistema de localização em tempo real de maior rapidez de execução e de maior fiabilidade.

## 5. Conclusões e Trabalhos Futuros

Existe um cada vez mais crescente número de sistemas de localização em tempo real, que procuram tornar-se uma ferramenta essencial na mobilidade das populações. Muitas das aplicações não são inteiramente dedicadas a pessoas com deficiência visual, mas a servir toda a população no seu geral, que necessite de orientação em locais desconhecidos, sendo necessário unicamente um equipamento como um *SmartPhone*, que cada vez é mais comum entre nós. A maior parte desses sistemas foi desenvolvido para ser utilizado em ambientes abertos (*outdoor*), fazendo recurso a instrumentos como o GPS ou a cálculos por triangulação das células GSM, para efetuar a localização.

Os meios *indoor* têm contudo demonstrado um dos grandes desafios para os investigadores. Devido à impossibilidade da utilização de instrumentos como o GPS, ou determinação por triangulação através das células GSM, são utilizadas tecnologias baseadas em sistemas de sensores, câmaras, dispositivos RFID e utilização da infraestrutura de *Access Points* da rede *WLAN*, para determinação da localização e auxílio na navegação.

A solução estudada e desenvolvida nesta dissertação, utiliza para a localização e navegação interior unicamente pontos de referência passivos e pertencentes à própria natureza do meio, dispensando a utilização de equipamentos e utensílios exteriores, exceto uma câmara e um Sistema de Informação Geográfica, ao contrário de algumas propostas referidas no estado de arte como o projeto Dhristi (Ran *et al.*, 2004) que utiliza *tags* de ultrassom ou o NAVIO (Retscher e Thienelt, 2004) e Feng e Liu, 2012, que propõe filosofias de navegação *indoor* baseadas nos sinais das redes *WiFi* e ainda os trabalhos desenvolvidos por Willis e Helal, 2005, com *tags* RFID. No entanto, a solução apresentada nesta dissertação não deverá ser entendida como melhor ou substituta das referidas, mas como complementar às já existentes técnicas mistas de navegação.

O modelo adotado para o sistema de navegação é inspirado no modelo cognitivo humano e da forma como a informação obtida pelos sensores óticos é processada e relacionada, estabelecendo associações que permitem determinar a posição espacial com

recurso a um mapa mental pré-aprendido da área de navegação. Durante a nossa locomoção, através de um reflexo aparentemente involuntário é desencadeado um processo contínuo que visa retirar atributos e pontos de referência que possam ser elucidativos quanto à possível localização. O conjunto de atributos extraído permite através de associações lógicas relacionar a informação e compará-la com um mapa mental, já existente de uma determinada área ou região. Esse mesmo mapa mental possibilita a navegação entre dois pontos numa área geográfica considerando unicamente a triagem de “determinados pontos-chave,” armazenados em memória.

No seguimento dos mesmos princípios, foi desenvolvido um sistema de localização e navegação que com base num mapa e relacionamentos lógicos entre os objetos extraídos das imagens capturadas e as informações estimadas de movimento realize uma previsão quanto ao posicionamento espacial do indivíduo. É de realçar que qualquer sistema de navegação, o humano inclusive, é fortemente dependente da sua percepção ao movimento realizado, principalmente quando o ambiente é muito semelhante e a extração de atributos só por si é insuficiente. Deste modo a odometria desempenha um papel fundamental não podendo ser desprezada, ainda mais em sistemas cuja capacidade de absorção de informação é muito restrita quando comparada com a realizada pelo cérebro humano, pelo que no sistema desenvolvido a estimação de posição é diretamente influenciada pelas componentes de movimento estimadas, na falta de sensor adequado, permitindo a localização através da utilização de pontos de referência semelhantes.

Contudo, sendo a componente de odometria estimada e não real, pode resultar em erros de localização quando o indivíduo não segue o movimento esperado ou quando ocorrem falsas deteções, originando em perda de localização, só colmatada quando o utilizador encontra novamente uma série de objetos de referência característicos de uma determinada área ou região. Futuramente poderá ser de equacionar a utilização de um sensor de aceleração e uma bússola digital como componentes auxiliares para determinar com precisão os movimentos do indivíduo, bem como criar métodos de análise contínua que permitam criar um modelo simples da arquitetura do meio capturado, através de conjuntos de linhas e cantos que permitam distinguir corredores de salas e restantes divisões.

A extração de atributos das imagens capturadas, por sua vez, envolve diversos métodos, visando o reconhecimento de vários objetos tais como portas, escadas,

retângulos, cacifos, sinalética vertical, e informações textuais. O foco na diversidade de objetos a detetar permite uma maior redundância, aumentando a probabilidade de detecção, no meio, de pelo menos um dos objetos e conseqüentemente aumentar a eficiência de o individuo conseguir se localizar. Fator diferenciador de alguns dos sistemas enunciados no estado de arte como são exemplo Tjan *et al.*, 2005 ou Cougland e Manduchi, 2009 que se focam unicamente na detecção de *tags* e Foong e Razali, 2011 que baseia a localização somente na análise dos textos extraídos.

As técnicas utilizadas na construção dos algoritmos, na sua maioria, são inspiradas pelos mesmos princípios como são exemplo os métodos desenvolvidos para o reconhecimento de portas e escadas, que utilizam a extração de linhas dos contornos dos objetos na imagem com recurso à transformada de *Hough* (Hough , 1962), diferindo unicamente no pós-tratamento da informação obtida. A detecção de portas e escadas demonstrou-se eficaz, no entanto é influenciável pelo ruído e deficiências da iluminação bem como pela existência de estruturas e padrões que pela sua semelhança podem ser confundíveis, resultando em falsas detecções. Devido às restrições impostas à orientação das linhas que constituem os contornos pode existir a ocorrência de falhas de detecção em ambas as estruturas. No caso prático das portas só é possível a detecção a partir de imagens frontais das mesmas, contudo o modelo de detecção desenvolvido é composto por dois algoritmos que permitem o reconhecimento de portas a partir de imagens parciais e independentemente das mesmas se encontrarem abertas ou fechadas. Futuramente será previsto a criação de métodos adicionais que permitam flexibilizar os ângulos que limitam o reconhecimento de portas e escadas sem comprometer a fiabilidade.

Por sua vez, a detecção de objetos retangulares é baseada numa análise geométrica com base nos ângulos formados entre os segmentos de reta de um conjunto de vértices extraído após análise aos contornos dos objetos, sendo que alguns dos princípios de análise foram retirados dos algoritmos desenvolvidos para as portas e escadas. Através das relações de altura e comprimento é possível diferenciar o tipo de retângulo detetado e conseqüentemente retirar ilações de reposicionamento com recurso ao SIG. Os cacifos, por sua vez, são reconhecidos através de uma comparação entre as áreas e as relações das arestas dos retângulos extraídos. A detecção de retângulos tem o intuito de servir como informação adicional, visto este tipo de formas geométricas ser muito popular em qualquer meio interior demonstrou-se de pouca utilidade para a estimação

direta de posicionamento, no entanto são de extrema utilidade quando usados como complemento a outras informações adquiridas, ou associados como realizado para o reconhecimento de cacifos. Tal como no reconhecimento de portas e escadas a eficiência de detecção demonstrou-se influenciável pelo ruído e deficiências de iluminação.

Para o reconhecimento de portas, escadas e retângulos foi utilizado um detetor de contornos *Canny* (Canny, 1986) de valores de *threshold* fixos determinados empiricamente através dos ensaios realizados, será de equacionar para efeitos futuros a utilização de um detetor de contornos com determinação automática dos valores de *threshold*, como o proposto por Lv *et al.*, 2010, com a finalidade de tornar os algoritmos de reconhecimento adaptáveis às oscilações de iluminação sentidas durante a navegação.

Para o reconhecimento de outros objetos de estrutura mais complexa como são exemplos os extintores e sinalizadores, foram utilizadas bibliotecas de descritores SURF (Bay *et al.*, 2008) que caracterizam cada um dos objetos através da extração dos pontos-chave que os definem. A utilização do descritor SURF demonstrou-se bastante eficiente no reconhecimento de objetos, apesar de terem sido verificados tempos de execução comprometedores nos ensaios realizados, obtidos muito provavelmente na ocorrência de imagens com áreas de interesse muito grandes. Como desenvolvimento futuro, será importante a utilização de mapas de saliência, através de bibliotecas computacionais como a *Nick's Machine Perception ToolBox* (Butko *et al.*, 2008), já utilizada com bons resultados nos trabalhos que implicam navegação em tempo real, como por exemplo o desenvolvido por Saleiro *et al.*, 2012.

Por fim as informações de texto são retiradas com o auxílio de uma aplicação externa, Tesseract (Tesseract-OCR, 2007). A utilização de informações de texto como referência demonstrou-se de extrema utilidade, visto geralmente essas mesmas informações permitirem determinar concretamente a posição do indivíduo. Tal como para o reconhecimento de objetos com recurso ao SURF, é proposto como trabalho futuro a utilização de mapas de saliência, para a segmentação das áreas de interesse para aplicação do OCR, com o objetivo de diminuir consideravelmente o tempo de execução.

Num âmbito geral para a diminuição de tempo de processamento, para além das enunciadas alterações futuras, também é importante referir que o tempo de processamento está igualmente ligado às dimensões das imagens, que nos ensaios



realizados foi de 640x480. Como tal será de esperar melhores resultados de processamento se for utilizado em testes futuros imagens de dimensões inferiores, como por exemplo 320x240.

A utilização de um sintetizador de voz para a comunicação de ordens de movimento, objetos detetados e localização atual é ferramenta indispensável quando os recetores são pessoas com deficiências visuais. No entanto, como verificado nos ensaios os tempos envolvidos em todo o processo de comunicação atrasam todos os restantes processos de reconhecimento e navegação, resultando em algum desconforto por parte do utilizador. De forma a minimizar este problema e tornar todo o sistema mais fluído, podemos no futuro optar por paralelizar o código, nomeadamente o módulo de comunicação, com o da análise de imagem, com recurso ao OpenMP (OpenMP, 2012). Ou usar outro tipo de interface com o utilizador, como uma pulseira que por “pequenos choques” sinalize as direções e sentidos a tomar.

Os objetivos propostos para a realização desta dissertação foram atingidos com sucesso, sendo a utilização do sistema desenvolvido praticável em ambientes interiores. Contudo, sistemas como o aqui retratado ainda estão em tenra idade e apresentam limitações que os inviabilizam de prescindir dos utensílios normais de um cego, no entanto são um precioso auxílio na navegação diária de invisuais, principalmente quando estes se encontram perante ambientes desconhecidos. O seu amadurecimento é uma questão de tempo e esforço, sendo certo que num futuro recente ganhem a robustez necessária para serem um instrumento imprescindível e fiável na mobilidade de deficientes visuais.

Como conclusão final, a presente dissertação foca a criação de um sistema de navegação *indoor* utilizando pontos de referência visuais e um SIG para a contextualização espacial e tem como principais contribuições:

- a) Permitir a localização através de um sistema de informação geográfica construído em qualquer aplicação para o efeito, muitas das quais livres de encargos, ou mesmo utilizar os desenhos de arquitetura, desenvolvidos por aplicações como o AutoCad (que é a ferramenta mundialmente aceite como “*standard*” para o desenho técnico de plantas em edifícios e ordenamento paisagístico);
- b) Permitir a localização simplesmente através da combinação do SIG com referências visuais, obtidas com recurso a uma câmara de baixo custo, não sendo

necessários outros equipamentos auxiliares (incluindo equipamentos de sonar, lazer, etc. ou etiquetas ativas);

- c) Estabelecer rotas de navegação e determinação de coordenadas de movimento para o auxílio à navegação.

As publicações realizadas e em preparação no âmbito da dissertação foram:

- a) Serrão, M., Rodrigues, J.M.F., Rodrigues, J.I. and du Buf, J.M.H (2012) Indoor localization and navigation for blind persons using visual landmarks and a GIS, *Procedia Computer Science*, Vol. 14, pp. 65 – 73. doi: 10.1016/j.procs.2012.10.008.
- b) Serrão, M., Shahrabadi, S., Moreno, M., José, J., Rodrigues, J.I., Rodrigues, J.M.F., du Buf, J.M.H. (2013) Vision and GIS system for indoor localization and navigation for blind persons, accepted for *Int. J. Universal Access in the Information Society*.
- c) Serrão, M., Rodrigues, J.M.F., and du Buf, J.M.H (2013) Vision based indoor localization and navigation for blind persons, in preparation for *Int. J. of Image Processing (IJIP)*.

# Bibliografia

- Arras, K., Tomaris, N., Jensen, B., Siegwart, R. (2001). Precision and reliability for applications: Multisensor on-the-fly localization. *Robotics and Autonomous Systems*, 34, 2-3.
- Balakrishnan, G., Sainarayanan, G., Nagarajan, R., Yaccob, S. (2005). On stereo processing procedure applied towards blind navigation aid – SVETA. *Proc. 8th Int. Symposium on Signal Processing and Its Applications*, (2), 567-570.
- Bay, H., Ess, A., Tuytelaars, T., Gool, L. V. (2008). SURF: Speeded up robust features. *Computer Vision and Image Understanding*, 110(3), 346-359.
- Bennett (2010). Jampal, Acedido em 2 de Outubro de 2012 <http://www.jampal.sourceforge.net>.
- Bolstad, P. (2012). Geodesy, datums, map projections, and coordinate systems, In: *GIS Fundamentals*, 4th Edition, 71-122. Minnesota: Eider Press.
- Bradski, G.,Kaehler, A. (2008a). Contours, In: *Learning OpenCV*, 234-244. Sebastopol: O'Reilly Media.
- Bradski, G., Kaehler, A. (2008b). Contours, In: *Learning OpenCV*, 245-247 Sebastopol: O'Reilly Media.
- Bradski, G., Kaehler, A. (2008c). Contours, In: *Learning OpenCV*, 258-260 Sebastopol: O'Reilly Media.
- Butko, T., Zhang, L., Cottrell, G., Movellan, J. (2008). Visual salience model for robot cameras. *Proc. IEEE Int. Conf. on Robotics and Automation*, 2398-2403.
- Calonder, M., Lepetit, V., Strecha, C., Fua, P. (2010). Brief: Binary robust independent elementary features. *Proc. European Conf. on Computer Vision*, 778-792.
- Canny, J. (1986). A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 8 (6), 679 – 698.

- Chandrasekhar, V., Takacs, G., Chen, D. M., Tsai, S. S., Reznik, Y., Grzeszczuk, R., Girod, B. (2012). Compressed Histogram of Gradients: A Low-Bitrate Descriptor. *Int. J. of Computer Vision*, 96(3), 384-399.
- Coughlan, J., Manduchi, R. (2009). Functional Assessment of a camera phone-based wayfinding system operated by blind and visually impaired users. *Int. Journ. Artif. Intell. Tools*, 18(3), 379-397.
- Crandall, W., Bentzen, B. L., Myers, L., Brabyn, J. (2001). New orientation and accessibility option for persons with visual impairment: transportation applications for remote infrared audible signage. *Clinical and Experimental Optometry*, 84(3), 120-131.
- Darken, R., Allard, T., Achille, L. (1998). Spatial orientation and wayfinding in large-scale virtual spaces. *Presence*, 7(2), 101–107.
- de By, R. A., Knippers, R. A., Sun, Y., Ellis, C. M., Kraak, M., Weir, M. J. C., Georgiadou, Y., Radwan, M. M., Van Westen, C. J., Kainz, W., Sides, E. J. (2001a). A gentle introduction to GIS, In: *Principles of Geographic Information Systems*, 52-54. Netherlands: ITC.
- de By, R. A., Knippers, R. A., Sun, Y., Ellis, C. M., Kraak, M., Weir, M. J. C., Georgiadou, Y., Radwan, M. M., Van Westen, C. J., Kainz, W., Sides, E. J. (2001b). Data processing systems, In: *Principles of Geographic Information Systems*, 158-164. Netherlands: ITC.
- Desimone, R., Duncan, J. (1995). Neural mechanisms of selective visual attention. *Annual Review of Neuroscience*, 18, 193–222.
- Douglas, D., Peucker, T. (1973). Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *The Canadian Cartographer*, 10(2), 112–122.
- du Buf, J.M.H., Barroso, J., Rodrigues, J.M.F., Paredes, H., Farrajota, M., Fernandes, H., José, J., Teixeira, V., Saleiro, M. (2011) The SmartVision navigation prototype for blind users. *JDCTA: Int. J. of Digital Content Technology and its Applications*, 5(5), 351 - 361. doi : 10.4156/jdcta.vol5.issue5.39.

- E-Adept (2009). Electronic Assistance for Disabled and Elderly Pedestrians and Travellers. Astando Inc. Acedido a 22 de Novembro de 2012. <http://www.eadept.se>.
- Evans, C. (2009). Notes on the opensurf library. Tech. Rep. CSTR-09-001, University of Bristol. Acedido em Janeiro de 2013. <http://www.chrisevansdev.com>.
- Feng, J., Liu, Y. (2012), Wifi-based Indoor Navigation with Mobile GIS and Speech Recognition. *Int. J. of Computer Science Issues*, 9(6), 256-263.
- Foong O., Razali N. (2011). Signage Recognition Framework for Visually Impaired People. *Proc. Int. Conf. on Computer Communication and Management*, 488-492.
- Fox, D., Burgard, W., Thrun, S. (1999). Markov Localization for Mobile Robots in Dynamic Environments. *Journal of Artificial Intelligence Research*, 11, 391-427.
- Gargi,U., Kasturi, R., Strayer, S.H. (2002). Performance characterization of video-shot-change detection methods. *IEEE Trans. on Circuits and Systems for Video Technology*, 10, 1–13.
- Golledge, R. G. (2002). The open door of GIS. *Handbook of environmental psychology*, 244-255.
- Goodchild, M. F., (2012). Geographic Information Systems. W.S. Bainbridge (Ed), *Leadership in Science and Technology*. (2) 238–245. Thousand Oaks, CA: SAGE.
- GDAL (2012). Geospatial Data Abstraction Library. Acedido a 22 de Maio de 2012. <http://www.gdal.org/>.
- Hamming, R. W. (1950). Error Detecting and Error Correcting Codes, *The Bell Systems Technical J.*, 29(2), 147-160.
- Harris, C., Stephens, M. (1988). A combined corner and edge detector. *Alvey Vision Conf.*, 147–151.
- Hough, P., 1962. A method and means for recognizing complex patterns. U.S.Patent No.3 069 654.
- Huang, C.-R., Lee, H.-P., Chen, C.-S. (2008). Shot Change Detection via Local Keypoint Matching. *IEEE Trans. on Multimedia*, 10( 6), 1097-1108.

- Itti, L., Koch, Ch., Niebur, E. (1998). A model of saliency-based visual attention for rapid scene analysis. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 20(11), 1254-1259.
- José, J., Moreno, M., Pinilla-Dutoit, J., Rodrigues, J.M.F., du Buf, J.M.H. (2012). Blavigator: a navigation aid for blind persons. *In Proc. European Conf. on Visual Perception*, 41, pp. 99.
- Kawewong, A., Tangruamsub, S., Hasegawa, O. (2010). Position-invariant Robust Features for Long-term Recognition of Dynamic Outdoor Scenes. *IEICE Trans. on Information and Systems*, 9, 2587–2601.
- Kim, C.G., Song, B.S. (2007). Design of a wearable walking-guide system for the blind. *Proc. of the 1st Int. Convention on Rehabilitation engineering and assistive technology*, 118-122.
- Koch, Ch., Ullman, S. (1985). Shifts in selective visual attention: towards the underlying neural circuitry. *Human Neurobiology*, 4, 219-227.
- Kortenkamp, D., Weymouth, T. (1994). Topological mapping for mobile robots using a combination of sonar and vision sensing. *Proc. of the AAAI*, 979–984.
- Krulikovská, L., Polec, J. (2012). An Efficient Method of Shot Cut Detection. *Int. J. of Computer and Information Engineering*, 63-79.
- Le Meur, O., Le Callet, P., Barba, D., Thoreau, D., François, E. (2004). From low level perception to high level perception, a coherent approach for visual attention modeling. *Proc. of SPIE Human Vision and Electronic Imaging*, 5292, 284-295.
- Lowe, D. G. (2004). Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60 (2), 91–110.
- Lv, M., Su, H., Li, Y. (2010). An Adaptative Canny Detector with New Differential Operator. *Proc. 6th Int. Conf. on Wireless Communications Networking and Mobile Computing*, 1-4.
- Martins, J. A., Rodrigues, J., du Buf, J. M. H. (2009). Focus of attention and region segregation by low-level geometry. *Proc. Int. Conf. Comp. Vis. Theo. App.*, (2), 267–272.

- Meger, D., Forssen, P., Lai, K., Helmer, S., McCann, S., Southey, T., Baumann, M., Little, J. J., Lowe, D. G. (2008). Curious George: An attentive semantic robot. *Rob. Aut. Sys.*, 56 (6), 503–511.
- Montello, D. R. (2005). Navigation, In P. Shah & A. Miyake (Eds.). *The Cambridge handbook of visuospatial thinking*, 257–294. Cambridge: Cambridge University Press.
- Montello, D.R., Sas, C. (2006). Human Factors of Wayfinding in Navigation. *International Encyclopedia of Ergonomics and Human Factors*, 2003-2008.
- Morioka, H., Yi, S., Hasegawa, O. (2011). Vision-Based Mobile Robot's SLAM and Navigation in Crowded Enviroments. *Proc. Int. Conf. on Intelligent Robots and Systems*, 3988-4005.
- OpenCV (2003). Open Computer Vision. Acedido a 12 de Junho de 2012. <https://code.ros.org/trac/opencv/browser/trunk/opencv/samples/c/squares.c?rev=27>.
- OpenCV (2012). Open Computer Vision. Acedido a 10 de Fevereiro de 2013. <http://opencv.willowgarage.com/>.
- OSGEO (2013). *Open Source Geospatial Foundation*. Acedido a 13 de Janeiro de 2013. <http://www.osgeo.org/>.
- Otsu, N. (1975). A Threshold Selection Method from Gray Level Histograms, *IEEE Trans. in Systems Man and Cybernetics*, (9), 62-66.
- Priya, G.G., Domnic, S. (2012). Transition Detection Using Hilbert Transform and Texture Features. *American J. of Signal Processing*, 35-40.
- QGIS (2011). Quantum Geographic Information System. Acedido em 15 de Dezembro de 2011. <http://www.qgis.org/>.
- Ramer, U. (1972). An iterative procedure for the polygonal approximation of plane curves. *Computer Graphics and Image Processing*, 1(3), 244–256.
- Ran, L., Helal, S., Moore, S. (2004). Drishti: an integrated indoor/outdoor blind navigation system and service. *Proc. of the 2nd IEEE Annual Conf. on Pervasive Computing and Communications*, 22-30.

- Retscher G., Thienelt M. (2004). A Navigation Guidance Service for Pedestrians. *J. of Global Positioning Systems*, (3), 208-217.
- Ridler, T. W., Calvard, S. (1978). Picture thresholding using an iterative selection method. *IEEE Trans. on Systems Man. and Cybernetics*, 8(8), 630–632.
- Rodrigues, J., du Buf, J. M. H. (2009). A cortical framework for invariant object categorization and recognition. *Cogn. Proc*, 10 (3), 243–261.
- Rosten, E., Drummond, T. (2006). Machine learning for highspeed corner detection. *Proc. European Conf. on Computer Vision*, 1, 430-443.
- Rublee, E., Rabaud, E., Konolige, K., Bradski, G. (2011). ORB: An Efficient Alternative to SIFT or SURF. *Proc. Int. Conf. on Computer Vision*, 2564-2571.
- Russ, J. C. (2011a). Human Vision, In: *The Image Processing Handbook, 6th Edition*, 88-90. Boca Raton: CRC Press/ Taylor & Francis Gorup.
- Russ, J. C. (2011b). Image Enhancement in the Spatial Domain, In: *The Image Processing Handbook, 6th Edition*, 269-336. Boca Raton: CRC Press/ Taylor & Francis Gorup.
- Saleiro, M., Rodrigues, J. M. F., du Buf, J.M.H. (2012). Minimalistic Vision-based Cognitive SLAM. *Proc. 4th Int. Conf. on Agents and Artificial Intelligence, Special Session Intelligent Robotics*, 1, 614-623.
- Seitz, J., Vaupel, T., Jahn, J. Meyer, S., Boronat, J.G. Thielecke, J. (2010). A Hidden Markov Model for urban navigation based on fingerprinting and pedestrian dead reckoning. *Proc. 13th Conf. on Information Fusion*, 1-8.
- Simmons, R., Koenig, S. (1995). Probabilistic robot navigation in partially observable environments. *Proc. of the Int. Join Conf. on Artificial Intelligence*, 1080–1087.
- Suzuki, S., Abe, K. (1985). Topological Structural Analysis of Digitized Binary Images by Border Following. *Computer Vision, Graphics, and Image Processing*, 30(1), 32-46.
- Tesseract-OCR (2007). Google Code. Acedido a 8 de Março de 2012. <http://code.google.com/p/tesseract-ocr/>.



- Tjan, B., Beckmann, P., Roy, R., Giudice, N., Legge, G. (2005). Digital Sign System for Indoor Wayfinding for the Visually Impaired. *Proc. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, 3(20), 30-30.
- Treisman, A. M., Gelade, G. (1980). A feature Integration Theory of Attention. *Cognitive Psychology*, 12, 97-136.
- Ulrich, I., Borenstein, J. (2000). Local obstacle avoidance with look-ahead verification. *Proc. IEEE Int. Conf. on Robotics and Automation*, 3, 2505–2511.
- Vaughan, R., Stoy, K., Sukhatme, G. S., Mataric, M. (2002). Lost: Localization-space trails for robot teams. *IEEE Trans. on Robotics and Automation*, 18(5), 796–812.
- Willis, S., Helal, S. (2005). RFID Information Grid for Blind Navigation and Wayfinding. *Proc. 9th IEEE Int. Symposium on Wearable Computers*, 34-37.
- Zheng, W., Yuan, J., Wang, H., Lin, F., Zhang, B. (2005). A novel shot boundary detection framework. *Visual Communications and Image Processing*, 5960, 410-420.

