

Universidade de Trás-os-Montes e Alto-Douro
Escola de Ciências e Tecnologia - Departamento de Engenharias



Mestrado em Engenharia Informática

Robótica de Assistência a Idosos

Apoio à Toma de Medicamentos

Leonel Agostinho Costa Crisóstomo

Dissertação realizada no âmbito do Mestrado em Engenharia Informática

Orientador: Professor Doutor Vítor Filipe

Vila Real, 2018

Robótica de Assistência a Idosos

Apoio à Toma de Medicamentos

Dissertação realizada no âmbito do Mestrado em Engenharia Informática

Leonel Agostinho Costa Crisóstomo

Orientador: Professor Doutor Vítor Filipe

Composição do Júri:

Presidente:

Doutor Hugo Alexandre Paredes Guedes da Silva

Vogais:

Doutor Nuno Miguel Fonseca Ferreira

Doutor Vítor Manuel de Jesus Filipe

Vila Real, 2018

Agradecimentos

A elaboração desta dissertação não teria sido possível sem colaboração e apoio de diversas pessoas e por esse motivo gostava de deixar aqui expresso o meu reconhecimento e gratidão a todas essas pessoas que de forma direta ou indireta, contribuíram para que este projeto se tornasse possível.

Em primeiro lugar, ao meu orientador **Professor Doutor Vítor Manuel de Jesus Filipe**, gostava de demonstra o meu apreço pela orientação prestada, motivação, partilha de conhecimentos e disponibilidade durante toda a execução deste projeto.

Passo a agradecer a todos os professores da licenciatura e mestrado, com os quais me cruzei, pelos conhecimentos transmitidos, que se tornaram conhecimentos importantes para o desenvolvimento desta dissertação.

Aos meus pais, por tudo o que me proporcionaram desde a educação, até à possibilidade da realização do mestrado e desta dissertação. As minhas palavras de gratidão pelo incentivo, paciência e apoio em todas as etapas da minha vida, não esquecendo o apoio financeiro.

À minha irmã, pelo apoio, críticas e ajuda na revisão dos textos.

À minha namorada, Carla Costa, pela presença constante em todos os momentos de desenvolvimento deste projeto. Obrigado pela paciência, compreensão, apoio, confiança e ajuda demonstrada no decorrer de todo este projeto.

A todos os meus amigos pelo apoio, motivação e por proporcionarem momentos que permitiram descontraír de todo o stress.

Resumo

O envelhecimento da população mundial e a correspondente diminuição de população jovem ativa nos países desenvolvidos está a provocar uma grande pressão sobre os recursos humanos especializados na prestação de cuidados necessários às pessoas idosas.

Frequentemente, os idosos possuem um regime de medicação vasto e complexo. Aliado a isto, os constantes avanços na tecnologia e neste caso específico, na robótica, fazem com que os robôs surjam como uma alternativa para preencher a lacuna existente entre a necessidade e o fornecimento de cuidados de saúde a idosos.

Embora existam diversas soluções tecnológicas que visam o auxílio à toma de medicação, todas elas são limitadas na interação com o utilizador, nomeadamente incapazes de irem ao encontro do idoso e de lhe entregar o medicamento e a água no horário devido.

Nesta dissertação é proposto um sistema robótico que visa o auxílio a idosos, ajudando-os no cumprimento do regime de medicação a que estejam sujeitos. O robô usa o seu sistema de locomoção para se movimentar até ao idoso e através de visão por computador deteta a embalagem do medicamento e identifica a pessoa que o deve tomar no horário correto. Para a realização da tarefa foi desenvolvida uma aplicação suportada por uma base de dados com informações sobre os idosos da habitação, os medicamentos que têm prescritos e os respetivos horários de toma.

O trabalho experimental foi efetuado com o robô NAO, da empresa SoftBank Robotics, recorrendo a ferramentas de desenvolvimento como o MySQL, Python e OpenCV. A identificação facial dos idosos e o reconhecimento dos medicamentos é efetuado através algoritmos de visão por computador que processam as imagens adquiridas pela câmara do robô.

Com o objetivo de avaliar o desempenho dos algoritmos de visão por computador, foram efetuados testes de reconhecimento de objetos, deteção facial e reconhecimento facial utilizando bases de dados públicas. Os testes possibilitaram obter métricas qualitativas do desempenho dos algoritmos utilizados.

Para demonstração do conceito foi feito um teste num cenário simples que recria o ambiente de uma habitação com idosos que são assistidos pelo robô na toma de medicamentos.

**Palavras-chave – Robótica de assistência a idosos; Visão por computador;
Reconhecimento de objetos; Reconhecimento facial;**

Abstract

The ageing of the world's population and the corresponding decrease in the active young population in developed countries, are putting a great strain on human resources specialized in the care of the elderly population.

Often, the elderly have a vast and complex medication regimen. Alongside this, the constant advances in technology and, in this specific case, in robotics, make the robots appear as an alternative to fill the gap between the need and supply of health care for the elderly population.

Although there are several technological solutions that provide help in taking their medications, they are all limited in the interaction with the user. In this case, they are unable to walk to the elderly and deliver the medication and water in time.

In this dissertation it is proposed a robotic system that provides assistance to the elderly, helping them to accomplish the medication regimen to which they are subject. The robot uses its locomotion system to move to the elderly and through computer vision, detects the packaging of the medicine and identifies the person who should take it at the correct time. For the accomplishment of the task an application was developed supported by a database with information about the elderly of the dwelling, the medicines that they have been prescribed and the respective time of intake.

The experimental work was done with the robot NAO, of the company SoftBank Robotics, using development tools like MySQL, Python and OpenCV. Facial recognition of the elderly and the identification of the right medications are performed through computer vision algorithms that process the images acquired by the robot's camera.

In order to evaluate the performance of the computer vision algorithms, object recognition, facial detection and facial recognition, tests were performed using public databases. The tests enable the acquisition of qualitative metrics of the performance accomplished by the algorithms used.

For demonstration of the concept a test was done in a simple scenario that recreates the environment of a dwelling with elderly people who are assisted by the robot in the intake of the medication.

Keywords – Assistive robotic for the elderly; Computer vision; Object recognition; Facial recognition;

Índice geral

Agradecimentos.....	I
Resumo.....	III
Abstract.....	V
Lista de figuras.....	XI
Lista de tabelas.....	XV
Abreviaturas, siglas e acrónimos	XVII
Abreviaturas	XVII
Siglas.....	XVII
Acrónimos	XVIII
1. Introdução.....	1
1.1. Motivação e objetivos	1
1.2. Estrutura da Dissertação	4
2. Revisão da literatura.....	5
2.1. Robôs.....	5
2.1.1. Robótica de assistência	7
2.1.2. Robótica social interativa	9
2.1.3. Robótica social de assistência.....	9
2.1.4. Robótica de assistência a idosos.....	11
2.2. Interação Humano-Robô (HRI)	12
2.3. Reconhecimento facial.....	13
2.3.1. Detecção de face	15
2.3.2. Extração de caraterísticas	18
2.3.3. Classificação	18
2.3.4. Reconhecimento facial através de redes neuronais	20
2.4. Reconhecimento de objetos.....	20
2.4.1. Extração de caraterísticas	22
2.4.2. Classificação	23
3. Trabalhos relacionados.....	25

3.1. Robôs companhia.....	25
3.1.1. Paro	25
3.1.2. iCat	27
3.2. Robôs de serviço.....	29
3.2.1. Care-o-bot.....	29
3.2.2. Accompany	31
3.3. Robôs ou aplicações de suporte à toma de medicamentos.....	32
3.3.1. PILLO	33
3.3.2. Lembrete de remédios - Medisafe	36
4. Materiais e métodos	39
4.1. Robô NAO	39
4.1.1. Hardware	40
4.1.2. Software	47
4.2. Algoritmos de visão por computador	50
4.2.1. Local Binary Patterns Histogram (LBPH)	50
4.2.2. SIFT.....	52
4.3. Ferramentas de desenvolvimento.....	55
4.3.1. Python	55
4.3.2. MySql	55
4.3.3. Python SDK	56
4.3.4. OpenCV	56
4.3.5. Schedule	56
4.3.6. Numpy	57
4.3.7. PILLOW	57
5. Implementação	59
5.1. Arquitetura do sistema desenvolvido.....	59
5.2. Base de dados	63
5.3. Reconhecimento de medicamentos	64
5.3.1. Aquisição da imagem.....	65
5.3.2. Extração das características	66
5.3.3. Correspondência dos pontos chave	67
5.4. Reconhecimento da pessoa.....	69
5.4.1. Aquisição da imagem.....	70

5.4.2. Detecção facial.....	70
5.4.3. Treino	71
5.4.4. Reconhecimento	71
6. Resultados e testes	73
6.1. Métricas	73
6.2. Testes de avaliação dos algoritmos de visão por computador.....	76
6.2.1. Reconhecimento de objetos	76
6.2.2. Detecção facial.....	79
6.2.3. Reconhecimento facial	82
6.3. Teste em cenário real.....	84
7. Conclusões.....	89
7.1. Limitações e trabalho futuro	90
Referências bibliográficas.....	91

Lista de figuras

Figura 1 - Revolta dos robôs contra os humanos na obra R.U.R. ¹	5
Figura 2 - Unimate, primeiro robô industrial. ²	6
Figura 3 - Robô da Vinci, robô cirurgião. ³	7
Figura 4 - Robôs na indústria automível. ⁴	7
Figura 5 - (a) Robô da Vinci, robô cirurgião. ⁵ (b) "cão-robô" para ajuda a pessoas com deficiências visuais. ⁶ (c) Robô Welwalk, robô de reabilitação. ⁷ (d) Robô Autom, robô para ajuda na perda de peso. ⁸	8
Figura 6 - Esquema explicativo da robótica social de assistência como interceção da robótica social interativa e de assistência.	10
Figura 7 - Exemplo de exoesqueleto. ⁹	11
Figura 9 - Robô Paro, um robô de companhia. ¹⁰	12
Figura 10 - Etapas genéricas do reconhecimento facial. Adaptado de (Zhao & Rosenfeld, 2003).	14
Figura 11 - Diferentes abordagens ao reconhecimento facial.	15
Figura 12 - Figura ilustrativa do funcionamento do algoritmo knn (Alaliyat, 2008).	19
Figura 13 - Etapas genéricas no reconhecimento de objetos.	21
Figura 14 - Exemplo de extração de características importantes da imagem usando o algoritmo SURF.	22
Figura 15 - Exemplo de correspondência de características entre uma imagem de treino e uma imagem do ambiente incluindo o objeto usando o algoritmo knn. ¹²	23
Figura 16 - Idosos interagindo com o robô Paro. ¹³	26
Figura 17 - iCat, o robô com aparência de um gato, desenvolvido pela Philips Research. ¹⁴ ...	27
Figura 18 - Exemplo de expressões faciais representadas pelo robô iCat (Saerbeck, 2009).	28
Figura 19 - Evolução das versões do robô Care-o-bot, desenvolvido pelo Fraunhofer institute. (A evolução das versões está organizada da esquerda para a direita e de cima para baixo). ¹⁵	30
Figura 20 - Protótipo do robô utilizado no projeto Accompany (Pérez et al., 2014).	31
Figura 21 - Caixa organizadora de medicamentos em que cada coluna corresponde aos dias da semana e cada linha corresponde respetivamente a medicamentos para tomar de manhã, meio-dia, noite e ainda um suplente.	33
Figura 22 - Robô Pillo, um robô para cuidados de saúde. ¹⁶	34
Figura 23 - Lembrete do horário da toma do medicamento gerado pelo robô pillo. ¹⁷	35
Figura 24 - Notificação recebida pelo utilizador quando chega ao horário da toma do medicamento.	36
Figura 25 - Funcionalidade de completar o nome do medicamento no processo de inserção dos medicamentos a tomar.	37
Figura 26 - Robô NAO desenvolvido pela SoftBank Robotics. ¹⁸	39
Figura 27 - Posição dos quatro microfones na cabeça do robô. Adaptado de ("Loudspeakers — Aldebaran 2.1.4.13 documentation," n.d.).	41
Figura 28 - Posição dos dois altifalantes na cabeça do robô. Adaptado de ("Microphones — Aldebaran 2.1.4.13 documentation," n.d.).	41
Figura 29 - Imagem com o nome de todas as articulações e respetiva localização. ¹⁹	43

Figura 30 - Imagem explicativa dos movimentos possíveis e a respetiva associação ao eixo de rotação. ²⁰	43
Figura 31 – Imagem demonstrativa do ângulo de visão das duas câmaras do robô NAO. ²¹ (a) – Ângulo de visão das duas câmaras na vertical. (b) – Ângulo de visão das duas câmaras na horizontal.....	45
Figura 32 - Esquema representativo do funcionamento do NaoQi e as interações entre os diversos componentes.	48
Figura 33 - Layout principal do programa Choregraphe.....	50
Figura 34 – Exemplo de extração do código LBP. Adaptado de (Huang et al., 2011).	51
Figura 35 – Exemplo das etapas de criação do histograma concatenado. ²³ (a) Imagem original. (b) Imagem com as caraterísticas realçadas. (c) Divisão da imagem com as caraterísticas realçadas em blocos. (d) Criação de histogramas de cada bloco. (e) Concatenação de todos os histogramas parciais num só histograma.....	51
Figura 36 - Etapas do algoritmo SIFT.....	52
Figura 37 – Processo de deteção de extremos comparando o pixel identificado com “X” com os 8 pixéis vizinhos e com os 9 pixéis vizinhos de escala superior e inferior. Adaptado de (David G Lowe, 2004).	53
Figura 38 – Histograma de orientação de um bloco 4x4. ²³	54
Figura 39 - Demonstração da criação de um vetor descritor através da concatenação do histograma de orientação. ²³	54
Figura 40 – Exemplo de funcionamento do python SDK no robô NAO.	56
Figura 41 - Exemplo de funcionamento da biblioteca Schedule.....	57
Figura 42 - Diagrama do sistema implementado para apoio à toma de medicação.	59
Figura 43 - Esquema da arquitetura do sistema desenvolvido e as respetivas interações entre os três componentes do mesmo.	60
Figura 44 - Função desenvolvida para a leitura de todos os horários de toma de medicamento e respetivo agendamento dos mesmos utilizando a biblioteca Schedule.	60
Figura 45 - Diagrama de atividades demonstrativo de todos os passos do sistema no momento da toma de medicamento.....	62
Figura 46 - Diagrama da base de dados desenvolvida.	64
Figura 47 - Diagrama de atividades de todos os passos no reconhecimento de medicamentos.	65
Figura 48 - Exemplo de extração de caraterísticas utilizando o SIFT.	66
Figura 49 – Exemplo do resultado da comparação entre as caraterísticas de uma imagem de treino e uma imagem de teste.	67
Figura 50 - Exemplo do resultado do algoritmo de reconhecimento de objetos no caso do medicamento ser reconhecido.	68
Figura 51 - Imagem do robô NAO a carregar o medicamento e a água com o cesto preso no pescoço.	68
Figura 52 - Diagrama de atividades de todos os passos no reconhecimento de um idoso.....	70
Figura 53 - Aplicação desenvolvida para facilitar na contagem dos dados obtidos nos testes efetuados.....	76
Figura 54 - Exemplos de imagens de treino utilizadas no decorrer deste teste.....	77
Figura 55 - Exemplos de imagens de teste utilizadas no decorrer deste teste.....	77
Figura 56 - Imagem do cenário de teste em ambiente real.....	85

Figura 57 - Medicamentos utilizados no teste em cenário real. (a) Mirtazapina Psidep. (b) Claritine. (c) Bissoltussin.	85
Figura 58 - Representação gráfica das várias etapas do teste em cenário real.	87

Lista de tabelas

Tabela 1 - Caraterísticas detalhadas dos componentes do robô NAO.	42
Tabela 2 - Informações de todas as articulações e respectivos valores mínimos e máximos aceites.	44
Tabela 3 - Apresentação de algumas das caraterísticas das câmaras do NAO e respetiva parametrização.	46
Tabela 4 - Parametrização da resolução das câmaras e respetivas resoluções associadas.	46
Tabela 5 - Identificadores associados a cada uma das câmaras do NAO.	46
Tabela 6 - Tabela das linguagens de programação aceites pelo robô NAO. Baseado em ("SDKs – Aldebaran 2.5.10.8 documentation",n.d).....	49
Tabela 7 - Tabelas da base de dados.	63
Tabela 8 - Matriz de confusão. Adaptado de (Bradley, 1997).	74
Tabela 9 - Matriz de confusão com os resultados obtidos no teste de reconhecimento de objetos efetuado.....	77
Tabela 10 - Avaliação de desempenho do algoritmo de reconhecimento de objetos.	78
Tabela 11 - Matriz de confusão com os resultados obtidos no teste de deteção facial efetuado.	80
Tabela 12 – Avaliação de desempenho do algoritmo de deteção facial.....	82
Tabela 13 – Resultados obtidos no teste de reconhecimento facial.	83
Tabela 14 - Avaliação de desempenho do algoritmo de deteção facial.	84
Tabela 15 - Tabela demonstrativa do procedimento experimental.	86

Abreviaturas, siglas e acrónimos

Abreviaturas

Abreviatura	Significado(s)
et al	e outros autores.
etc.	etcetera, outros

Siglas

Sigla	Significado(s)
HRI	Interação humano-robô
HCI	Interação Humano-Computador
KNN	k-nearest neighbors
HMM	Hidden Markov Model
CNN	Convolutional neural network
LFA	local feature analysis
LPB	Local binary patterns
LBPH	Local Binary Patterns Histograms
SGBD	Sistema de Gestão de Base de Dados
SDk	Software Development Kit
Png	Portable Network Graphics
VP	Verdadeiro positivo
VN	Verdadeiro negativo

FP	Falso positivo
FN	Falso negativo

Acrónimos

Acrónimo	Significado(s)
AR	Robótica de assistência
SIR	Robótica social interativa
SAR	robótica social de assistência
SIFT	Scale-invariant feature transform
SURF	Speeded up robust features
ORB	Oriented FAST and rotated BRIEF
ICA	Independent component analysis
AIST	The National Institute of Advanced Industrial Science and Technology
ASCII	American Standard Code for Information Interchange

1. Introdução

O aumento da esperança média de vida em conjunção com a queda da natalidade gerou um dos problemas centrais no século XXI, o envelhecimento da população. Este facto está a fazer com que as sociedades mais desenvolvidas se transformem em sociedades envelhecidas (Cabral, Ferreira, Silva, Jerónimo, & Marques, 2013). Até 2050, a população em idade ativa na Europa cairá para 364 milhões, uma redução de 25% em comparação com o nível de 1995. Por outro lado, a população com idade de 65 anos ou superior aumentará de forma constante, de 101 milhões em 1995, para quase 173 milhões em 2050 (United Nations, 2007). De forma semelhante, em Portugal as projeções indicam que no ano de 2060 existirão 3 idosos para apenas 1 jovem e uma esperança média de vida de 81 anos. Em 2011, 19% da população tinha 65 ou mais anos, segundo os resultados do censo efetuado em 2011 (Cabral et al., 2013).

O crescimento exponencial da população idosa e correspondente diminuição de população jovem ativa nos países desenvolvidos, está a conduzir à falta de mão-de-obra especializada para prestar os cuidados necessários aos idosos. Os avanços registados no campo da robótica fazem com que os robôs surjam como uma alternativa para preencher a lacuna existente entre a necessidade e o fornecimento de cuidados de saúde a uma população cada vez mais envelhecida (Broadbent, Stafford, & MacDonald, 2009).

O conceito da robótica aplicada na saúde surgiu há algumas décadas, onde era usado principalmente para atividades físicas, na reabilitação, bem como na assistência pessoal para tarefas e atividades diárias. (Butter et al., 2008). O mais recente desenvolvimento na robótica são robôs com comportamento social, intitulados de robôs sociais. Estes robôs são projetados com a habilidade de interagir como humanos ou como animais de estimação. Qualquer robô desenvolvido com uma entidade social e capaz de comunicar com os utilizadores pertence a esta categoria de robôs sociais. (Dautenhahn & Billard, 1999)

1.1. Motivação e objetivos

Cerca de 60% dos idosos vivem sozinhos ou com outros idosos (Reis et al., 2016). A maioria dos idosos desejam permanecer nas suas casas o máximo de tempo possível (Tinker & Lansley, 2005), evitando o trauma psicológico e a mudança dispendiosa para

um lar de idosos ou hospital longe dos seus amigos e familiares (Scopelliti & Vittoria, 2005).

De forma a robôs assistentes serem usados pelos idosos, é necessário que os robôs sejam aceites pelos idosos (Heerink, Kroese, Evers, & Wielinga, 2006). Esta aceitação é dependente de diversos fatores relacionados com o utilizador e com o robô. Devido aos diversos problemas dos idosos, como por exemplo problemas de mobilidade, memória e visão, a opinião em relação a robôs assistentes é muito positiva (Broadbent et al., 2009). No geral, as pessoas idosas mostram-se interessadas na utilização de robôs assistentes para ajudar a manter a sua independência (Broadbent et al., 2009) especialmente quando o idoso tem a perceção da necessidade para a utilização do robô assistente (Tinker & Lansley, 2005).

Com o envelhecimento das pessoas, os riscos para deficiências cognitivas e falhas de memória aumentam e habilidades de atenção diminuem (Deary et al., 2009) (Simen, Bordner, Martin, Moy, & Barry, 2011). Esses declínios cognitivos relacionados com a idade entre os idosos diminuem a sua capacidade de realizar determinadas atividades diárias de forma independente, como comer e vestir (Lafortune, 2007).

Frequentemente, os idosos possuem vários problemas de saúde e sofrem de diversas doenças. Consequentemente, a ingestão múltipla de medicamentos é comum para os idosos podendo diminuir os sintomas, melhorar a qualidade da vida e ocasionalmente curar a doença. Infelizmente, mais de 50% dos idosos têm dificuldades em cumprir os regimes de medicação devido a muitos fatores, como a gravidade da doença, comprometimento cognitivo e o regime de medicação complexo (J. Lin, Sklar, Oh, & Li, 2008) (Campbell et al., 2012). O não cumprimento do regime de medicação pode resultar na progressão da doença (Jimmy & Jose, 2011) e subseqüentes complicações evitáveis, desta forma, para ajudar no cumprimento do regime de medicação podem ser usados robôs assistentes.

Existem diversas propostas tecnológicas para assistir o cumprimento dos horários da medicação por parte dos idosos, incluindo aplicações para dispositivos móveis, caixas inteligentes para medicamentos e dispensadores de comprimidos que expõem o comprimido no horário devido.

Este trabalho pretende dar um contributo ao desafio causado pelo crescimento exponencial da população idosa e consequente insuficiência de pessoal especializado para lhe prestar os cuidados necessários, propondo um sistema robotizado de suporte à toma de medicamentos por parte de pessoas idosas. Os robôs possuem a capacidade de simular um assistente humano, fornecendo o medicamento e a água, ajudando o idoso nessa tarefa. Para este fim, o robô deve ser capaz de reconhecer o idoso e aceder ao seu regime de medicação para que no momento da toma da medicação de algum dos idosos da habitação, se possa dirigir ao repositório dos medicamentos, identificar o medicamento pretendido e consequentemente dirigir-se ao idoso para lhe entregar o medicamento e a água para o ingerir, auxiliando-o nesse processo para prevenir situações de não cumprimento do regime de medicação. Para isto, é necessário que o sistema robótico esteja incorporado com um sistema de visão que dê ao robô a percepção do ambiente e de um sistema de locomoção de forma a permitir a sua movimentação no ambiente em que se encontra. Estes sistemas são fundamentais para o cumprimento da tarefa de apoio à toma de medicamentos.

Neste contexto, são estabelecidos os seguintes objetivos principais do trabalho:

- Aprofundar o conhecimento acerca de algoritmos de visão por computador para o reconhecimento facial e reconhecimento de objetos;
- Implementação no robô NAO de algoritmos de reconhecimento facial e de objetos;
- Integração no robô de uma aplicação que faça a gestão da toma de medicamentos por partes dos idosos de uma habitação.

O principal foco desta dissertação incide sobre o reconhecimento facial e de objetos através do processamento das imagens adquiridas pelo robô, não sendo desenvolvida a parte da cinemática relativa à movimentação do robô nem à manipulação de objetos. Todos os aspetos relativos ao movimento do robô são feitos de modo determinístico e com distâncias pré-determinadas, não se considerando possíveis variações no cenário experimental apresentado e descrito no capítulo 6.3.

1.2. Estrutura da Dissertação

Esta dissertação encontra-se estruturada em 7 capítulos. Neste capítulo é feita uma introdução ao tema em estudo, a motivação e são definidos os objetivos que se pretendem atingir com esta dissertação.

No capítulo 2, revisão da literatura, são apresentados conceitos de robótica, incluindo o conceito de robótica de assistência para idosos, o tema do reconhecimento facial e reconhecimento de objetos através de visão por computador que constituem as bases necessárias para o desenvolvimento do trabalho.

No capítulo 3, são descritos alguns trabalhos relacionados com o tema abordado na tese. Neste capítulo são apresentados vários robôs e aplicações, que possuem como objetivo a assistência a idosos. Estão incluídos nestes trabalhos robôs para providenciar vários tipos de ajuda, incluindo ajuda no cumprimento de medicação.

No capítulo 4, são apresentados e descritos todos os materiais utilizados, desde o robô utilizado, os algoritmos de visão por computador e as ferramentas de desenvolvimento.

No capítulo 5, são apresentados todos os passos de implementação do sistema proposto, desde a idealização até às várias etapas desenvolvidas. Durante este capítulo, a arquitetura do sistema desenvolvido, o processo de reconhecimento de medicamentos e idosos serão apresentados e explicados.

No capítulo 6, são efetuados testes às várias etapas do sistema, como a detecção facial, o reconhecimento facial e o reconhecimento de medicamentos. Neste capítulo são descritas as métricas usadas, metodologia dos testes e são apresentados os respetivos resultados.

Finalizando, no capítulo 7, é realizada uma conclusão geral do projeto desenvolvido bem como as principais limitações.

2. Revisão da literatura

2.1. Robôs

Uma definição de robô afirma que este é um “mecanismo automático, por vezes com a aparência de um ser humano, capaz de fazer movimentos e executar certos trabalhos em substituição do homem” (“Definição ou significado de robô no Dicionário Infopédia da Língua Portuguesa,” n.d.).

O termo *robô* teve origem na palavra checa *robota*, que significa, servo, trabalho forçado (“Definition of robot - Merriam-Webster’s Student Dictionary,” n.d.). Este termo teve origem em 1921, na obra *R.U.R.*, do dramaturgo checo Karel Capek. A obra intitulada de *R.U.R.*, aparece em forma de protesto contra o rápido crescimento da tecnologia, onde é descrita a evolução de robôs com grandes capacidades e a eventual revolta contra os humanos (Capek K, 1923 apud Hockstein, Gourin, Faust, & Terris, 2007).



Figura 1 - Revolta dos robôs contra os humanos na obra *R.U.R.*¹

Apesar disto, a popularização dos robôs é atribuída a Isaac Asimov devido às suas três leis (Hockstein et al., 2007), formuladas pela primeira vez em 1940, responsáveis

¹ Disponível em: <http://www.computerhistory.org/revolution/artificial-intelligence/robotics/13/324/1262>

por governar o comportamento dos robôs nos seus livros de ficção científica (Clarke, 1994).

Após diversos anos de existência de robôs na ficção científica, estes tornaram-se realidade em 1958 através da General Motors que utilizou o primeiro robô industrial, o Unimate (figura 2). Este robô foi utilizado pela primeira vez em 1961 e após esse marco a utilização da robótica na indústria deflagrou (“Unimate - The First Industrial Robot,” n.d.) e os robôs começaram-se a ser utilizados para efetuar trabalhos mecânicos na indústria numa tentativa de minimizar o erro e a lesão dos trabalhadores enquanto é aumentada a eficiência da produção (Faust, 2007).



Figura 2 - Unimate, primeiro robô industrial.²

Robôs são dispositivos bastante heterógenos, por conseguinte existe uma variedade de classificações para diferentes tipos de robôs para os ajudar a descrever (Hockstein et al., 2007). Robôs podem ser caracterizados como braços automatizados, robôs de serviço, robôs de investigação, robôs médicos ou humanoides (“Clasificación de robots | Wiki de Robótica,” n.d.). Devido a esta heterogeneidade, as características dos robôs variam conforme a classificação do mesmo. Por exemplo, as características dos robôs industriais divergem das dos robôs cirúrgicos. Robôs cirúrgicos (Figura 3) são projetados para serem lentos, delicados e precisos, enquanto os industriais (Figura 4) são projetados para serem rápidos, fortes e reincidentes.

² Disponível em: <https://www.robotics.org/joseph-engelberger/unimate.cfm>



Figura 3 - Robô da Vinci, robô cirurgião.³



Figura 4 - Robôs na indústria automóvel.⁴

Com o envelhecimento das sociedades desenvolvidas (Cabral et al., 2013), existe um aumento da necessidade do uso de robótica de assistência, sendo necessário outra abordagem a esta categoria da robótica permitindo a inclusão de robôs que operam através da interação social. (Feil-Seifer & Matarić, 2005) apresentaram uma definição de robótica social de assistência distinguindo as três categorias apresentadas em seguida.

2.1.1. Robótica de assistência

Inicialmente, robótica de assistência (AR), fazia referência a sistemas robóticos que forneciam suporte a pessoas com necessidades físicas. Com a evolução da robótica, esta abordagem tornou-se insuficiente devido à existência de robôs que fornecem suporte a

³ Disponível em: https://www.francetvinfo.fr/replay-radio/nouveau-monde/robot-chirurgien-ce-n-est-plus-de-la-science-fiction_1771307.html

⁴ Disponível em: <https://singularityhub.com/2012/05/04/better-faster-and-cheaper-these-robots-are-invading-car-manufacturing-plants/>

determinada necessidade humana sem o uso de contacto humano, tais como os robôs de companhia (Feil-seifer & Matari, 2005).

Desta forma, embora não exista uma definição exata para este termo, um robô de assistência é visto como um robô que auxilia utilizadores humanos em diversas atividades (Tapus, Mataric, & Scassellati, 2007) (Feil-seifer & Matari, 2005).

Existem diversos tipos de robôs de assistência como os robôs cirúrgicos (figura 5(a)) (Howe & Matsuoka, 1999), robôs que ajudam os deficientes visuais a manter a sua independência (figura 5(b)) (Tejima, 2001), robôs que ajudam na terapia de reabilitação (figura 5(c)) (Krebs, Dipietro, Volpe, & Hogan, 2003) e robôs que motivam para a prática de exercício físico e a perda de peso (figura 5(d)) (Kidd & Breazeal, 2005).



(a)



(b)



(c)



(d)

Figura 5 - (a) Robô da Vinci, robô cirurgião.⁵ (b) "cão-robô" para ajuda a pessoas com deficiências visuais.⁶ (c) Robô Welwalk, robô de reabilitação.⁷ (d) Robô Autom, robô para ajuda na perda de peso.⁸

⁵ Disponível em: <https://www.revistaplaneta.com.br/confie-em-seu-novo-cirurgiao/>

⁶ Disponível em: <https://www.dailymail.co.uk/sciencetech/article-2058754/Japanese-inventors-lead-way-robo-dog-guides-blind.html>

⁷ Disponível em: <https://mixpressweb.wordpress.com/2017/05/03/robo-da-toyota-ajudara-na-reabilitacao-de-pessoas-com-paralisia-de-membros-inferiores/>

⁸ Disponível em: <https://www.medgadget.com/2012/11/autom-the-robotic-weight-coach-interview-with-intuitive-automatas-founder-dr-cory-kidd.html>

2.1.2. Robótica social interativa

O termo robótica social interativa (SIR) foi inicialmente utilizado por (Fong, Nourbakhsh, & Dautenhahn, 2003). Quando se trata de um agente social isto implica a existência de relações interativas. Um agente capaz de interagir e ter um comportamento comunicativo é considerado social. Assim sendo, um robô social é um robô que participa em interações sociais de forma a completar algum objetivo ou motivação (Dautenhahn, 2007).

O termo robótica social interativa foi utilizado para descrever robôs onde a interação social desempenha um papel fundamental. Os robôs descritos como sociais interativos apresentam características como: expressar e/ou entender emoções, estabelecer/manter relações sociais e comunicar com um diálogo de alto nível (Fong et al., 2003). Este paradigma foi possível através da junção da ciência da computação com a interação Humano-Robô (HRI).

2.1.3. Robótica social de assistência

A robótica social de assistência (SAR) surge de forma a expandir a robótica de assistência de modo a incluir robôs que operam através da interação social. SAR é definida pela interseção entre a AR e SIR (figura 6). O seu objetivo, compartilhado com AR, é fornecer assistência a humanos. No entanto, o método usado para providenciar essa ajuda na SAR é através da interação social (Feil-Seifer & Matarić, 2005).

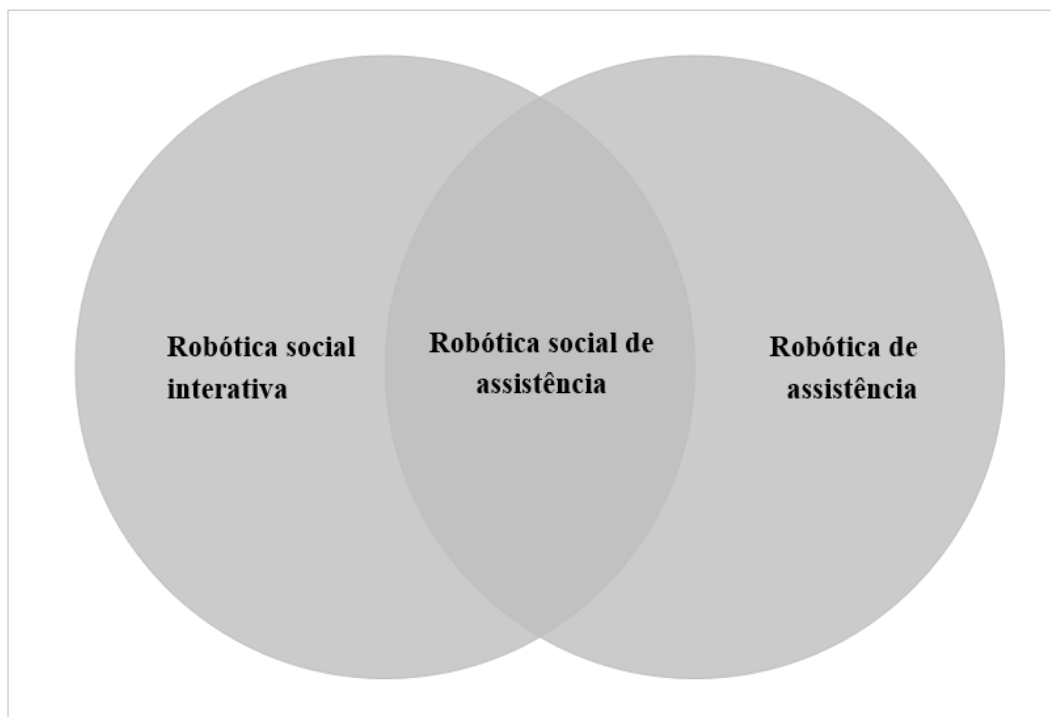


Figura 6 - Esquema explicativo da robótica social de assistência como interseção da robótica social interativa e de assistência.

A característica comum entre a SAR e a SIR é a existência de interação social em ambas, no entanto os propósitos são diferentes. No SIR, com o uso do robô, o objetivo é o desenvolvimento de interações humanas, por outro lado, no SAR, o objetivo é criar uma interação com o humano visando o fornecimento de assistência (Feil-Seifer & Matarić, 2005).

A SAR possui potencial para melhorar a qualidade de vida de diversos utilizadores, entre eles, idosos, pessoas com deficiências cognitivas e de desenvolvimento e pessoas com transtornos sociais. Embora estes dispositivos sejam relevantes para a sociedade, são ainda um desafio pois são altamente específicos, devido às necessidades especiais de cada utilizador (Tapus et al., 2007).

A SAR foi idealizada para ser usada em diversos locais, como hospitais, lar-de-idosos e residências particulares (Shah, 2017). Nesta, os indivíduos com variados tipos de necessidades especiais são vistos como utilizadores finais destes sistemas robotizados, no entanto, os sistemas SAR devem agir num meio real, onde familiares e médicos podem e devem ser incluídos. Por conseguinte, devem ser consideradas

questões éticas para todos aqueles indivíduos que sejam atingidos por este (Feil-Seifer & Mataric, 2011).

2.1.4. Robótica de assistência a idosos

A robótica de assistência a idosos pode ser dividida em dois grupos. Os dois grupos são inseridos separadamente nas categorias de robótica de assistência e robótica social de assistência. O primeiro grupo, pertence ao grupo da AR e corresponde aos robôs de reabilitação (Broekens, Heerink, & Rosendal, 2009). Os robôs deste grupo enquadram-se na categoria de AR (descritos na seção 2.1.1) e são responsáveis por fornecer ajuda a um utilizador humano sem uma interação social, como por exemplo um exoesqueleto (figura 7).

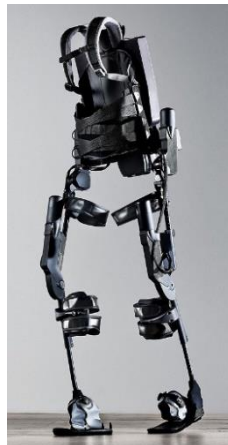


Figura 7 - Exemplo de exoesqueleto.⁹

O segundo grupo, pode ser separado em dois subgrupos, robôs de serviço e robôs de companhia que podem ser incluídos nas categorias de SAR, no caso do primeiro subgrupo e SIR no segundo subgrupo. Os robôs de serviço fornecem aos idosos a ajuda necessária nas tarefas básicas do dia-a-dia, como comer, tomar banho e vestir. Por outro lado, o segundo subgrupo é conhecido como os robôs de companhia (figura 8) e possui o objetivo de assistir na saúde e no bem-estar psicológico dos idosos (Broekens et al., 2009) (Kachouie, Sedighadeli, Khosla, & Chu, 2014).

⁹ Disponível em: <http://blogs.discovermagazine.com/crux/2011/11/15/exoskeletons-will-be-the-eyeglasses-of-the-21st-century/#.W88oZ2hKiUl>



Figura 8 - Robô Paro, um robô de companhia.¹⁰

Quando os robôs são desenvolvidos para aplicações específicas como a assistência a idosos, vários fatores pessoais e do robô devem ser considerados para a sua aceitação. Entre os vários fatores pessoais encontram-se, a idade, as necessidades, experiência com tecnologia e/ou robôs e as capacidades cognitivas e educacionais (Broadbent et al., 2009).

Além destes fatores pessoais, destacam-se os fatores do robô como a aparência, expressões faciais e tamanho (Broadbent et al., 2009).

A criação de sistemas robotizados, do segundo grupo, para assistência a idosos (robôs de serviço e robôs de companhia) implica a interação com o utilizador humano. Desta forma, torna-se imprescindível o estudo da interação Humano-Robô (HRI), permitindo assim projetar sistemas robóticos capazes de realizar tarefas interativas em ambientes partilhados com humanos, tornando-os capazes de interagir de forma eficaz e segura com humanos (Feil-Seifer & Mataric, 2009).

2.2. Interação Humano-Robô (HRI)

A HRI é um subconjunto do campo da interação Humano-Computador (HCI) (Yanco & Drury, 2002) e consiste no estudo das interações entre os humanos e os robôs. A HRI é composto na interseção de variadas áreas como a robótica, engenharia, ciência da computação, psicologia, ética entre outras (Dautenhahn, 2007). Com a evolução da robótica, a HRI ganhou importância de forma a projetar sistemas robóticos eficazes para uso em ambientes partilhados com humanos (Feil-Seifer & Mataric, 2009).

¹⁰ Disponível em: <http://journalofdementiacare.com/robots-in-dementia-care/>

A HRI pode ser categorizada com três direções, não mutuamente exclusivas: HRI centrada no robô, centrada no humano ou centrada na cognição do robô. No caso da HRI centrada no robô, realça a visão do robô como uma entidade autónoma, ou seja, o robô tenta cumprir algumas das suas necessidades (como identificado pelo desenvolvedor do robô). Neste caso, a HRI acontece para cumprir as suas necessidades sociais mesmo que não envolva nenhuma tarefa específica. A HRI centrada no homem, foca-se principalmente na forma como o robô pode cumprir uma tarefa de forma aceitável e confortável para os seres humanos. A HRI centrada na cognição do robô, realça a visão do robô como um sistema inteligente, ou seja, o robô toma decisões e resolve problemas para efetuar as ações que necessita (Dautenhahn, 2007)

Com a evolução da robótica, os robôs são cada vez mais utilizados em ambientes que estão adequados para as capacidades dos humanos. O ambiente onde o robô está presente pode sofrer alterações sem qualquer interferência do robô, por esse motivo, os robôs necessitam ter a perceção do ambiente que os rodeia.

De modo a que os robôs consigam efetuar as suas tarefas com eficiência é necessário que estes possuam não só habilidades mecânicas mas também boa interação social com os humanos (Tzafestas, 2016). O reconhecimento facial é um dos pontos importantes na HRI para que um robô social reconheça pessoas (Pan, Ge, He, & Chen, 2009). Esta importância é aumentada quando o sistema robotizado é utilizado com pessoas com necessidades especiais, em conformidade com as necessidades de específicas deste grupo de utilizadores.

2.3. Reconhecimento facial

Um sistema de reconhecimento facial é um sistema biométrico (Li & Jain, 2011) que pretende identificar faces presentes numa imagem ou vídeo, automaticamente (Parmar & Mehta, 2013). Este pode operar de duas formas distintas: 1) verificação de face (autenticação) ou 2) identificação facial (reconhecimento). A verificação de face consiste na correspondência de um para um, ou seja, é comparada a imagem adquirida com o *template* da face que está a ser verificada. Na identificação facial, é utilizado a correspondência de um para muitos, em que a imagem adquirida é comparada com os

templates de todas as faces de forma a determinar a identidade da face adquirida (Li & Jain, 2011).

Um sistema de reconhecimento facial normalmente consiste em três etapas (Figura 9). Estas etapas, ordenadas de forma sequencial, são: detecção da face, extração de características e classificação. No entanto, as duas primeiras etapas podem ser executadas de forma simultânea (Zhao & Rosenfeld, 2003).

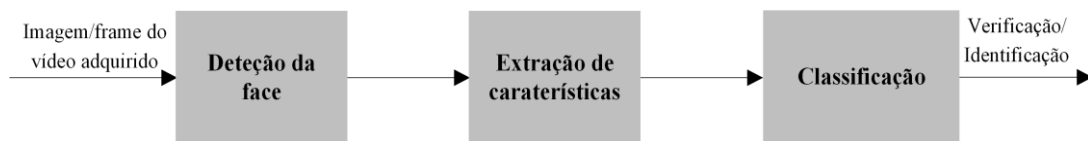


Figura 9 - Etapas genéricas do reconhecimento facial. Adaptado de (Zhao & Rosenfeld, 2003).

Sucintamente, no sistema de reconhecimento facial, a entrada consiste numa imagem ou um *frame* de um vídeo. Após findarem as várias etapas, a saída do sistema de reconhecimento facial, consiste na identificação ou na verificação da(s) face(s) presente(s) a imagem de entrada.

Embora estas etapas sejam genéricas existem outras abordagens que permitem a resolução ao problema de reconhecimento facial que não adotam este modelo de três etapas, como por exemplo o uso de *deep learning* (Zhao & Rosenfeld, 2003) (Lawrence, Giles, Tsoi, & Back, 1997) (Nixon & Aguado, 2012).

Os diversos algoritmos para o reconhecimento facial, podem ser divididos nas seguintes abordagens (Zhao & Rosenfeld, 2003):

- Holísticos
- Baseada em características
- Híbridas

Os métodos holísticos levam em consideração a região total da face para obter características (Zhao & Rosenfeld, 2003). Os métodos baseados em características extraem características faciais como os olhos, boca, nariz e outras marcas de referência.

Após a extração destas características, pretende-se relacioná-las e obter uma representação simplificada da face, como por exemplo calcular relações geométricas entre os pontos de referência (Zhao & Rosenfeld, 2003) (Parmar & Mehta, 2013). Os métodos híbridos, assim como o sistema de percepção humano, recorrem às outras duas abordagens de modo complementar, na tentativa de obter uma maior percentagem de sucesso (Zhao & Rosenfeld, 2003) (Parmar & Mehta, 2013).

Na figura 10 são apresentadas as diferentes abordagens no reconhecimento facial e alguns dos métodos pertencentes a cada abordagem (Zhao & Rosenfeld, 2003).

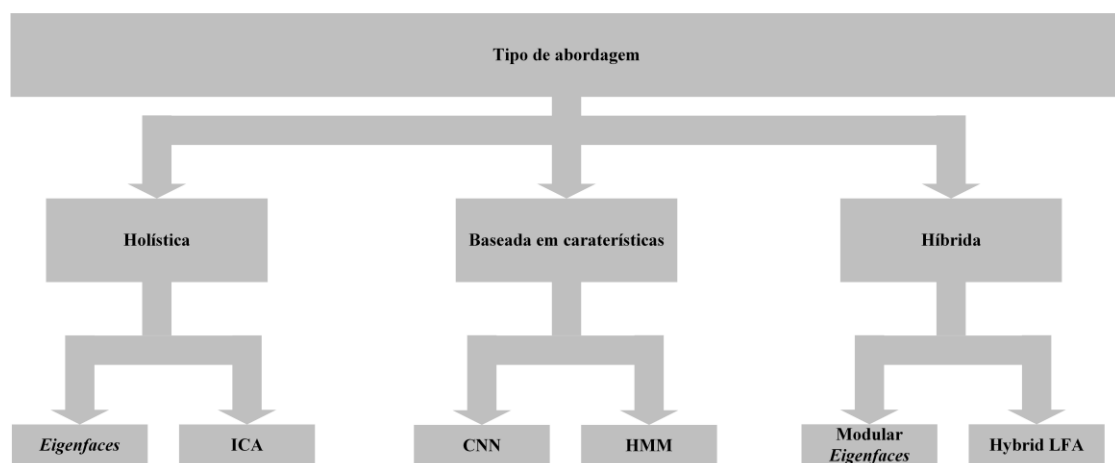


Figura 10 - Diferentes abordagens ao reconhecimento facial.

2.3.1. Deteção de face

Nesta etapa do reconhecimento facial, o principal objetivo é determinar se na imagem existem faces e proceder à segmentação, separando a região da(s) face(s) do plano de fundo (Li & Jain, 2011).

Na execução deste processo existem alguns problemas comuns como descrito pelos autores (Yang, Kriegman, & Ahuja, 2002). Entre os problemas são:

- **Pose** - A imagem da face varia conforme a posição relativa da câmara ou da própria posição da face. Com a variação da pose algumas das características podem ficar ocultas.
- **Expressão facial** - Expressões faciais fazem diferir a aparência da face.

- **Alteração nos componentes estruturais** - A presença de componentes estruturais como bigode, barba ou óculos que podem variar em tamanho, forma e cor.
- **Obstrução** - Numa imagem ou vídeo uma face pode estar totalmente ou parcialmente oclusa por objetos ou outras faces.
- **Orientação da imagem** - As imagens da face podem aparecer com diferentes rotações sobre o eixo ótico da câmara
- **Condição da imagem** - Fatores como iluminação e características da câmara afetam a aparência da face.

Existem diversas taxonomias dos métodos de deteção facial, no entanto, não existe nenhuma que seja globalmente aceite. Uma classificação proposta por (Yang et al., 2002) divide os algoritmos de deteção facial em quatro categorias, métodos baseados em conhecimento, características invariáveis, *templates* e aparência.

Métodos baseados em conhecimento

Estes métodos propõem detetar faces a partir da utilização de regras previamente definidas a partir do conhecimento humano. No geral, todas as faces têm características comuns, como dois olhos, nariz e boca e estes estão dispostos de forma específica na face. Um problema destes métodos remete-se ao facto da elevada dificuldade de transmitir o conhecimento para regras, pois caso as regras sejam muito detalhadas, isto pode gerar problemas em detetar faces que não cumpram todas as regras. Por outro lado, regras muito gerais podem gerar falsos positivos (Yang et al., 2002).

Métodos baseados em características invariantes

Estes métodos procuram encontrar características invariantes como a cor da pele e a textura da face de forma a detetar a sua presença na imagem (Yang et al., 2002).

A cor pode ser utilizada para a deteção da região facial devido à distribuição dos componentes de crominância de um rosto humano estar localizada numa estreita faixa do espaço de cores (Terrillon, David, & Akamatsu, 1998).

Estes métodos possuem, no entanto, uma desvantagem, essas características podem ser corrompidas devido à iluminação, ruído ou oclusões (Yang et al., 2002).

Métodos baseado em templates

Os métodos baseados em *templates* visam procurar uma face através da correspondência da imagem com formas geométricas associadas a elementos faciais como os olhos, boca, nariz e a forma da face. Estes elementos podem ser detetados na maioria das faces e possuem uma forma aproximada a certas formas geométricas. Através disto, os métodos procuram encontrar a melhor correspondência entre a imagem de teste e um *template* (Yang et al., 2002).

No início, estes métodos mostraram-se inadequados para a deteção facial, devido a dificuldades como a variação da escala, pose e forma. Posteriormente, foram propostos métodos mais complexos para combater estes problemas, entre os quais estão incluídos os templates deformáveis (Yuille, Cohen, & Hallinan, 1989) (Yang et al., 2002).

Métodos baseados na aparência

Ao contrário dos métodos baseados em *templates*, os métodos baseados na aparência não necessitam de conhecimento prévio sobre a característica a ser detetada, baseiam-se em técnicas de análise estatística e *machine learning*. Estes métodos utilizam um conjunto de imagens de teste que permitem encontrar características relevantes das imagens faciais e não faciais. Posteriormente, as características encontradas nas imagens de treino permitem detetar a presença ou não de uma face na imagem de teste.

Um dos algoritmos mais utilizados para detetar faces numa imagem é o Viola-Jones, que consiste num método baseado na aparência (Viola & Jones, 2001).

2.3.2. Extração de caraterísticas

A extração de caraterísticas consiste em extrair da imagem informações que descrevam de forma incontestável a face. Embora não exista uma definição concreta, caraterísticas podem ser vistas como segmentos de uma imagem com relativo significado.

Esta etapa é importante para as técnicas de reconhecimento facial e permite extrair informação útil para caraterizar uma classe, neste caso, distinguir faces de diferentes pessoas (Kumar & Bhatia, 2014) (Li & Jain, 2011).

Ao utilizar um método de reconhecimento facial nem todos os pixéis são essenciais, os pixéis redundantes podem ser descartados, sem prejudicar a finalidade do método. Através deste processo apenas os pixéis com relevância são considerados, baixando o custo computacional, tornando possível a utilização de sistemas de reconhecimento facial em tempo real (Silva, 2015).

2.3.3. Classificação

O processo de classificação consiste em descobrir qual a imagem de treino que se assemelha mais à imagem de teste (Hassaballah, Ali, & Alshazly, 2016). Através deste processo, a descrição da imagem mais semelhante é usada para a classificação da imagem de teste (Nixon & Aguado, 2012).

Nesta etapa são usadas as caraterísticas extraídas no processo anterior como entrada num algoritmo classificador, de forma a concluir se as caraterísticas extraídas nas imagens de treino e na imagem de teste são suficientemente idênticas de forma a poder concluir que existe uma correspondência entre as duas (Silva, 2015).

Um dos algoritmos mais conhecidos que segue esta abordagem é o o *K-nearest neighbour* (KNN) (Cover & Hart, 1967).

K-nearest neighbour (KNN)

O KNN é um dos algoritmos mais simples para a classificação no reconhecimento facial. Este algoritmo consiste em classificar as faces baseando-se nas imagens de treino mais próximas do vetor de características da imagem a detetar. O valor de “ k ” tem influência no número de vizinhos que são considerados, por exemplo, caso o “ k ” seja igual a 1 o algoritmo classifica o vizinho mais próximo como sendo a face correta (Kramer, 2013). Na figura 11, a imagem a classificar é representada pelo círculo verde. Os quadrados azuis e os retângulos vermelhos representam as imagens de treino de duas pessoas diferentes. A variação do valor de “ k ” está representada pelos círculos com linha preta a tracejado e sólida. No caso do círculo de linha solida, “ k ” possui o valor de 3, o que significa que serão consideradas as 3 amostras de treino mais próximas da imagem a classificar. No caso do círculo de linha tracejada, “ k ” possui o valor de 5, o que significa que serão consideradas as 5 amostras de treino mais próximas da imagem a classificar (Alaliyat, 2008).

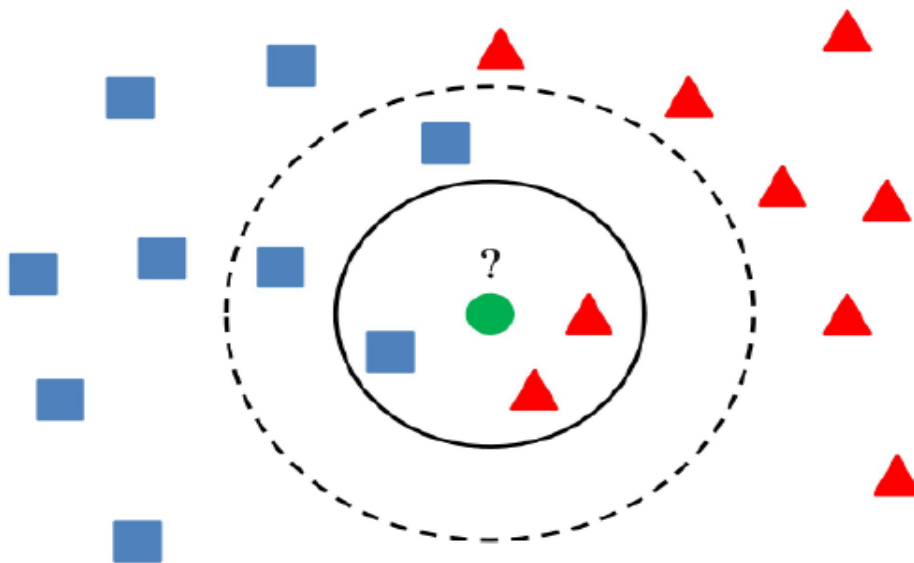


Figura 11 - Figura ilustrativa do funcionamento do algoritmo knn (Alaliyat, 2008).

2.3.4. Reconhecimento facial através de redes neuronais

Uma solução para o problema de reconhecimento facial que não utiliza o modelo genérico de três etapas anteriormente apresentado é através da utilização de redes neuronais artificiais (Zhao & Rosenfeld, 2003).

O desenvolvimento de redes neuronais na computação surgiu, com a tentativa de simulação do sistema nervoso biológico, através de vários elementos de computação simples (neurônios) num ambiente altamente conectado, esperando que a resposta em problemas complexos surja como o resultado da auto-organização ou aprendizagem (Sarle, 1994).

As redes neuronais são uma ferramenta bastante utilizada para reconhecimento e classificação de padrões e foram adaptadas para serem usadas no reconhecimento facial (Hjelmås & Low, 2001). O uso de redes neuronais no reconhecimento facial foi inicialmente demonstrado pelo autor Kohonen, através de uma rede neuronal de duas camadas responsável pelo reconhecimento de faces alinhadas e normalizadas (T. Kohonen, 1989 apud Fleming & Cottrell, 1990).

Após o aparecimento das redes neuronais no reconhecimento facial, foram desenvolvidos métodos que utilizam redes neuronais apenas para o processo de classificação (Kung & Taur, 1995), métodos para a detecção facial (Juell & Marsh, 1996) (Burel & Carel, 1994) (Vaillant, Monrocoq, & Cun, 1994) e métodos que as utilizam para ambos os processos (S.-H. Lin, Kung, & Lin, 1997).

2.4. Reconhecimento de objetos

O reconhecimento de objetos é uma tarefa fundamental em visão por computador e é amplamente utilizada para fins de inspeção, registro e manipulação (D G Lowe, 1999). Esta tarefa consiste em reconhecer automaticamente a identidade ou a categoria de um objeto (SHUKLA, Desai, & Publication, 2015) (Opelt, Pinz, Fussenegger, & Auer, 2006). Este processo pode ser baseado em diversos aspectos da imagem como por

exemplo a forma (Malik, Belongie, Leung, & Shi, 2001) ou a cor (Koubaroulis & Kittler, 2002).

O reconhecimento de objetos pode ser feito para um objeto específico ou para uma categoria genérica (SHUKLA et al., 2015) (Opelt et al., 2006). O reconhecimento da classe de um objeto requer uma forte capacidade de generalização dos métodos de classificação, de forma a ser capaz de obter sucesso com variações de iluminação, oclusões e ângulos de visão (Grauman & Darrell, 2005).

Um bom desempenho nos algoritmos de reconhecimento de objetos passa por usar características invariantes às transformações comuns. Os algoritmos como o *Scale-invariant feature transform* (SIFT) são bastante robustos para o reconhecimento de objetos específicos, sob diferentes pontos de vista ou condições de iluminação (David G Lowe, 2004). No entanto, para identificar uma categoria genérica, essas abordagens apresentam desempenhos inferiores. A análise da forma (Belongie, Malik, & Puzicha, 2000) de um objeto tem a vantagem de ter uma melhor capacidade de generalização em comparação com análises de características (como SIFT). Contudo, a análise da forma tem muitas limitações, especialmente quando o objeto a ser classificado não possui um plano de fundo limpo ou está parcialmente oculto (Ribeiro, 2008).

Para que seja possível um sistema de visão por computador reconhecer um objeto, é necessário que este conheça previamente as características do mesmo, obtidas na fase de treino (Shapiro & Stockman, 2001).

O processo de reconhecimento de objetos está normalmente dividido em 2 etapas: extração de características e classificação (figura 12).

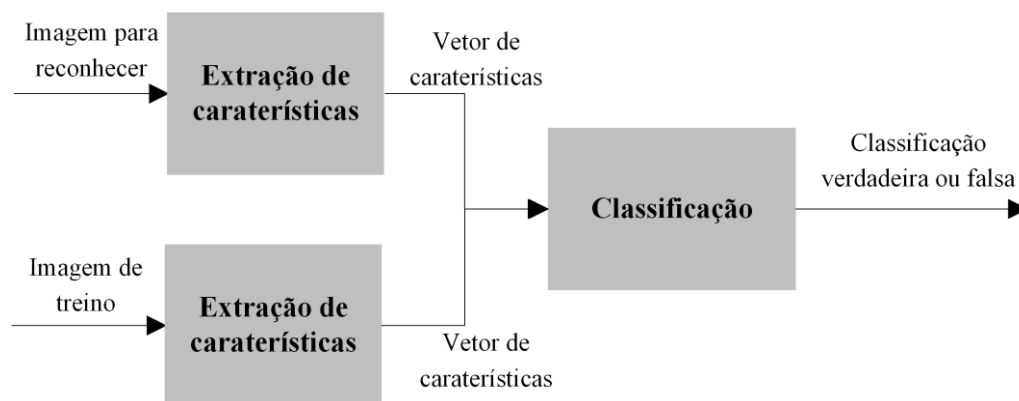


Figura 12 - Etapas genéricas no reconhecimento de objetos.

De forma sucinta, o processo de reconhecimento de objetos possui duas fases, uma fase de treino e uma fase de reconhecimento. Na fase de treino, o sistema extrai características das imagens de treino e insere uma descrição para o objeto ou classe de objetos. Na fase de reconhecimento, são extraídas as características da imagem de teste e são comparados com as características da fase de treino de modo a verificar se existem correspondências entre ambas as características.

2.4.1. Extração de características

De forma a que o sistema conheça a aparência de um objeto é necessário extrair características da imagem do objeto. Embora não exista uma definição concreta, as características podem ser vistas como segmentos de uma imagem com algum significado.

A extração de características tem por objetivo extrair determinados atributos das imagens de modo a que essas características descrevam de forma incontestável a classe de objetos ou o objeto em questão (Forsyth & Ponce, 2012) (figura 13)

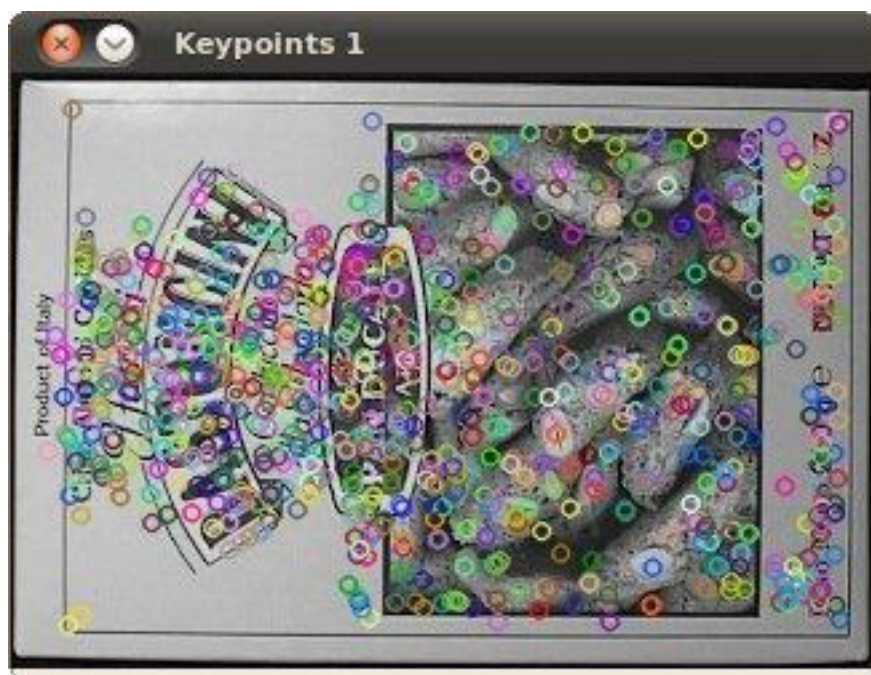


Figura 13 - Exemplo de extração de características importantes da imagem usando o algoritmo SURF.¹¹

Para o reconhecimento de objetos, geralmente, são usadas características invariáveis à mudança de iluminação ou ruídos.

¹¹ Disponível em: https://docs.opencv.org/2.4/doc/tutorials/features2d/feature_detection/feature_detection.html

Existem diversos algoritmos para extrair características, como por exemplo o SIFT (D.G. Lowe, 1999), o SURF (Bay, Ess, Tuytelaars, & Van Gool, 2008) ou o ORB (Rublee, Rabaud, Konolige, & Bradski, 2011). O resultado destes algoritmos consiste num vetor com as características que definem o objeto.

2.4.2. Classificação

O processo de classificação consiste na comparação entre as características da imagem de treino e da imagem de teste (figura 14), com o objetivo de verificar se o objeto da imagem de treino se encontra na imagem de teste e qual a sua localização.

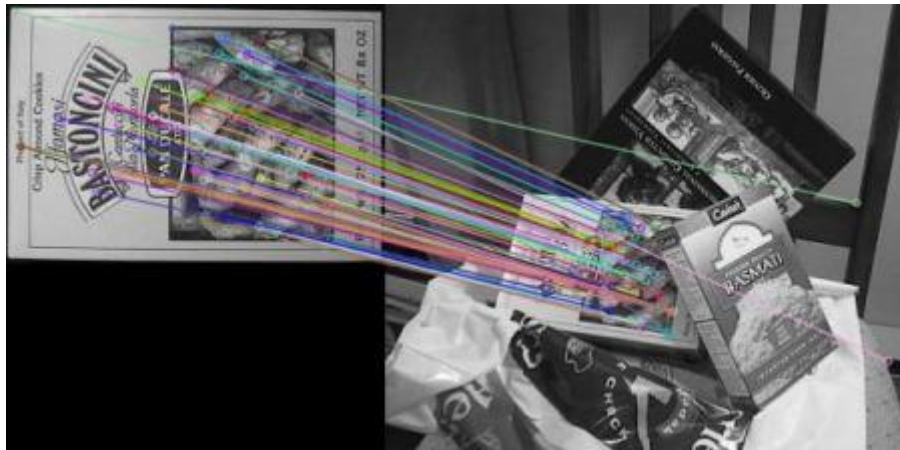


Figura 14 - Exemplo de correspondência de características entre uma imagem de treino e uma imagem do ambiente incluindo o objeto usando o algoritmo knn.¹²

Nesta etapa são usadas as características extraídas no processo anterior como entrada num algoritmo classificador de forma a concluir se as características extraídas nas imagens de treino e na imagem de teste são suficientemente idênticas de forma a poder concluir que existe uma correspondência entre as duas (Silva, 2015).

Um dos algoritmos usados para a classificação de objetos é o *K-nearest neighbour* (KNN) (Cover & Hart, 1967).

¹² Disponível em: https://docs.opencv.org/3.4/dc/dc3/tutorial_py_matcher.html

3. Trabalhos relacionados

Neste capítulo, serão apresentadas diversas soluções que visam o auxílio na toma de medicamentos, bem como sistemas robóticos de assistência a idosos que auxiliam de variadas formas e em diversas tarefas. Foi, também, efetuada uma pesquisa de forma a apurar as características e estudos realizados relacionados com as soluções apresentadas.

Este capítulo é dividido em três subcapítulos: Robôs de companhia, onde o principal objetivo é a assistência na saúde do idoso e do seu bem-estar psicológico; Robôs de serviço, em que o objetivo principal é o auxílio nas tarefas básicas do dia-a-dia do idoso; Robôs ou aplicações de suporte à toma de medicamentos, que visam o auxílio e o não esquecimento da toma de medicação.

3.1. Robôs companhia

Neste subcapítulo, serão apresentados alguns robôs do subgrupo robôs de companhia, enumerando as suas principais características e alguns estudos existentes relacionados com esses mesmos robôs.

Como já visto anteriormente, os robôs deste subgrupo caracterizam-se pelo objetivo de assistir o idoso na saúde e no bem-estar psicológico (Broekens et al., 2009) (Kachouie et al., 2014). Fatores como a sua aparência e uso das expressões faciais são fatores que ajudam na sua aceitação, permitindo cumprir o seu objetivo (Broadbent et al., 2009).

3.1.1. Paro

Paro é um robô parecido com uma foca bebé (figura 15), foi desenvolvido em 2003 pelo Instituto Nacional de Ciência e Tecnologia Industrial Avançada do Japão AIST (“PARO Therapeutic Robot,” n.d.) e mais tarde comercializado pela empresa *Intelligent System Co.*



Figura 15 - Idosos interagindo com o robô Paro.¹³

Este robô tem uma altura de aproximadamente 55cm e pesa 2,7Kg (Shibata, 2012). Foi desenvolvido com o objetivo de ser utilizado com idosos, particularmente com aqueles que possuem demência, incluindo-se assim na subcategoria de robôs de companhia. O Paro foi galardoado com o título de robô mais terapêutico do mundo, atribuído pelo *Guinness World Records* (“PARO Therapeutic Robot,” n.d.). O robô Paro possui uma imitação de voz de uma foca bebê real, sensor de luz, sensor de toque, sensor de áudio, sensor de temperatura e sensor de posição (Shibata, 2012).

Segundo o website oficial do robô Paro, a utilização deste robô pelos idosos resulta em:

- Redução do stress dos idosos e dos seus cuidadores;
- Estimulação da interação entre idosos e entre idosos e cuidadores;
- Efeitos psicológicos nos idosos, melhorando o relaxamento e a motivação.

O aumento das interações entre idosos relacionado com a utilização do robô Paro tem sido alvo de diversos estudos, alguns dos quais são sucintamente descritos em seguida.

¹³ Disponível em: <http://www.parorobots.com/photogallery.asp>

(Wada & Shibata, 2007) realizaram um estudo num lar de idosos em *Tsukuba*, no Japão. No estudo contaram com a participação de 12 idosos de idades compreendidas entre os 67 e os 89 anos que interagiram livremente com o robô durante 9 horas diárias. As conclusões retiradas com este estudo foram que as interações entre os 12 idosos aumentaram e originou melhorias em 2 meses.

(Kidd, Taggart, & Turkle, 2006) realizaram o estudo em dois lares de idosos, durante 4 meses, com um total de 23 idosos. Neste estudo foi comparado o nível de interação entre os idosos na presença do Paro ligado, desligado ou sem a presença do Paro. Os autores concluíram que a interação entre os idosos aumentou com a presença do Paro ligado.

3.1.2. iCat

iCat é um robô com a aparência de um gato, foi desenvolvido pela Philips Research (figura 16) para ser utilizado em cima de uma secretária (Saerbeck, 2009).



Figura 16 - iCat, o robô com aparência de um gato, desenvolvido pela Philips Research.¹⁴

¹⁴ Disponível em: <http://www.bartneck.de/2009/08/12/photos-of-philips-icat-robot/>

Este robô possui uma face mecânica, aproximadamente 40 cm de altura e não possui qualquer tipo de mecanismo de locomoção. O iCat é capaz de expressar emoções e expressões faciais (Figura 17) graças aos 13 graus de liberdade que possui para movimentar várias partes da sua cabeça. Para além disto, o iCat possui uma câmara, dois microfones, altifalantes, leds RGB nos ouvidos e patas, quatro sensores de toque nos ouvidos e patas e sensor de distância infravermelho. Com estas características é possível o robô comunicar através de som, luz e movimento.



Figura 17 - Exemplo de expressões faciais representadas pelo robô iCat (Saerbeck, 2009).

Este robô, através das suas expressões faciais, é capaz de estimular a interação social das pessoas, com o objetivo de fazer a ponte entre o mundo dos dispositivos digitais e dos seus utilizadores. A principal finalidade do desenvolvimento do iCat é permitir o estudo de aspetos de interação social entre utilizadores e robôs com expressões faciais (Saerbeck, 2009).

(Heerink et al., 2006) usaram o iCat para estudar como os utilizadores avaliavam a interação com o robô. Para isso, foram projetados dois cenários de interação diferentes, onde um deles o robô era socialmente mais interativo. Este estudo foi realizado na Holanda com 40 idosos. A avaliação foi realizada com questionários no final da experiência e mostrou que não houve diferença em termos de aceitação do robô nas duas formas de interação social, no entanto, os utilizadores mostraram-se mais confortáveis com a versão mais interativa.

Outro estudo dos mesmos autores visa avaliar e comparar o efeito do iCat e de uma personagem humana mostrada num ecrã em dois modos de condições expressivas diferentes. O agente da tela é capaz de expressar diferentes expressões faciais assim como o robô iCat, no entanto, as interações sociais não consistiam na mesma tarefa. Este estudo foi realizado na Holanda, com um total de 40 participantes com idades compreendidas entre 65 e 96 anos. O estudo concluiu que há um padrão claro de expressividade mais conversacional, uma frequência maior de comportamentos não-verbais de participantes que conversavam com um agente em uma condição mais expressiva. No entanto, o efeito é muito mais forte para o agente robótico (Heerink, Krose, Evers, & Wielinga, 2007).

3.2. Robôs de serviço

Os robôs de serviço têm o propósito de fornecer a ajuda necessária para as tarefas básicas do dia-a-dia. Estes robôs necessitam de algumas características específicas, como braços controláveis para diversas tarefas, sistema de locomoção bastante desenvolvido devido há existência de sítios com espaços reduzidos e sensores, devido ao uso dos robôs em ambiente real.

Neste subcapítulo serão debatidos dois robôs, apresentando as suas principais características e alguns estudos existentes relacionados com esses mesmos robôs.

3.2.1. Care-o-bot

Care-o-bot é um robô doméstico, desenvolvido pelo *Fraunhofer Institute*, e que está atualmente na versão 4 (figura 18). O Care-o-bot foi desenvolvido para diversos ambientes, desde ambiente doméstico até ambiente médico. No caso do ambiente doméstico, este pode assistir pessoas fornecendo serviços físicos, como por exemplo, carregar objetos (Hans, Graf, & Schraft, 2002) ou fornecer água (Graf, Reiser, Hägele, Mauz, & Klein, 2009). Além disso, serve como controlador central sobre os dispositivos de uma casa, o que permite que a informação seja enviada a familiares e serviços de saúde (Saerbeck, 2009).

O Care-o-bot 4, apresentado em 2015 (Lûdtke, 2016), é um robô com 158 cm de altura e 140Kg. A locomoção acontece através das quatro rodas na base do robô. Com este sistema de locomoção o robô fica possibilitado de efetuar movimentos complexos, mesmo com espaço reduzido. Este robô possui câmara estéreo 3D, três scanners a laser e dois braços mecânicos que lhe permitem entregar objetos aos utilizadores. Além disso, contem um ecrã tátil de 15” que permite uma interação básica através de menu (“Care-O-bot 4 - Technical Data,” n.d.). Outra característica importante de caráter social é o robô possuir expressões faciais apresentadas no ecrã (Kittmann et al., 2015).

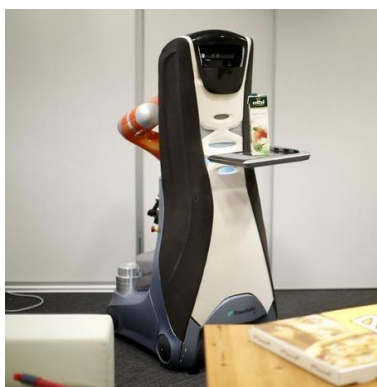


Figura 18 - Evolução das versões do robô Care-o-bot, desenvolvido pelo Fraunhofer institute. (A evolução das versões está organizada da esquerda para a direita e de cima para baixo).¹⁵

Ainda não existem estudos realizados com o Care-o-bot e segundo os autores (Graf et al., 2009) a previsão da futura utilização do Care-o-bot 3 em lares de idosos é muito

¹⁵ Disponível em: <https://www.care-o-bot.de/en/care-o-bot-3/history/care-o-bot-i.html>
<https://www.care-o-bot.de/en/care-o-bot-3/history/care-o-bot-ii.html>
<https://www.indiatimes.com/boyz-toyz/machines/careobot-service-robot-73511.html>
https://phoenixdesign.com/en/company/press/products/roboer_red-dot-best-of-the-best/

difícil de fazer pois o setor de saúde Alemão é o setor onde a tecnologia e a automação se difundem mais lentamente.

3.2.2. **Accompany**

Accompany é um projeto que foi apoiado pela União Europeia de Outubro de 2011 a Setembro de 2014 e teve contributos da Universidade de Hertfordshire, Universidade de Twente e o centro especializado em tecnologia e serviços para a manutenção da autonomia de idosos em casa (Gallego Pérez, Lohse, & Evers, 2014).

O projeto teve como objetivo fornecer aos idosos um parceiro robótico que lhes fornecesse serviços de maneira motivadora e socialmente aceitável. O sistema Accompany proporcionou assistência física, cognitiva e social nas tarefas diárias.

O robô desenvolvido para este projeto foi baseado na aparência do robô Care-o-bot 3 (figura 19). O software do robô é baseado no Arduino e possui uma autonomia de aproximadamente 20 horas. (Gallego Pérez et al., 2014)



Figura 19 - Protótipo do robô utilizado no projeto Accompany (Pérez et al., 2014).

Os investigadores deste projeto fizeram um estudo durante 3 semanas colocando o robô em casa de um utilizador. A atividade deste estudo consiste num exercício de

respiração proposto pelo robô três vezes por dia em horário aleatório. Durante a atividade o robô monitoriza, através de um sensor, os batimentos cardíacos do utilizador. Além disto, o utilizador poderá iniciar o exercício de respiração comunicando ao robô através de um tablet. Os dados para análise deste estudo, foram adquiridos através de sensores, entrevistas e questionários. Estes dados serviram para a obtenção de informações sobre três fatores: a evolução da percepção do utilizador em relação ao robô, o impacto do robô nas rotinas diárias do utilizador e as mudanças no estado psicológico do utilizador devido ao robô e ao exercício. Os resultados obtidos mostraram uma diminuição da percepção do utilizador em relação ao robô e as rotinas não se alteraram (embora numa das entrevistas o utilizador tenha reportado que teve mais cuidado a limpar o chão). Em relação ao estado psicológico, o utilizador mostrou elevado grau de depressão e baixo grau de felicidade, no entanto, os autores apontam estes fatores ao seu estado emocional na época em que se realizou o estudo. Os autores concluíram que o mais importante não foram propriamente os resultados obtidos nesta pesquisa, mas sim o conhecimento adquirido sobre os desafios reais e as advertências metodológicas. Com este estudo os autores aprenderam que é necessário testes técnicos intensivos e um esforço extra para explicar aos utilizadores o que é necessário estes fazerem (Gallego Pérez et al., 2014)

3.3. Robôs ou aplicações de suporte à toma de medicamentos

Existem diversas soluções que visam combater a preocupação com o não cumprimento da toma de medicação.

Apesar destas inúmeras soluções, nem todas as soluções são tecnológicas, como é o exemplo da caixa organizadora de medicamentos semanal (figura 20). No entanto, existem diversas soluções tecnológicas como aplicações para smartphone, robôs dispensadores de medicamentos, etc.



Figura 20 - Caixa organizadora de medicamentos em que cada coluna corresponde aos dias da semana e cada linha corresponde respectivamente a medicamentos para tomar de manhã, meio-dia, noite e ainda um suplente.

Neste capítulo, serão apresentadas duas soluções tecnológicas para auxiliar o cumprimento do regime de medicação, uma aplicação para smartphone e um robô. Para ambas as soluções serão apresentadas as suas características bem como as suas vantagens e desvantagens.

3.3.1. PILLO

O robô Pillo (figura 21) , foi desenvolvido pela Pillo Health e possui como principal objetivo o auxílio na saúde do utilizador (Kite-Powell, n.d.).



Figura 21 - Robô Pillo, um robô para cuidados de saúde.¹⁶

O Pillo está equipado com um processador ARM, um display touchscreen de 7", câmara HD, vários microfones, altifalante, Wi-Fi, Bluetooth e compartimento para medicamentos ("Meet Pillo | Your Personal Home Health Companion," n.d.).

Algumas das funcionalidades existentes no Pillo são: reproduzir rádio, mostrar o plano de saúde diário do utilizador, efetuar videochamadas e indicar as calorias de determinado alimento. No entanto as funcionalidades de lembrete e a dispensa do medicamento são as funcionalidades principais a destacar.

O Pillo permite o controlo na toma de medicação, lembrando o utilizador da toma dos medicamentos às horas devidas (figura 22), bem como a dispensa para um copo do medicamento correto, de entre os armazenados num compartimento existente no robô. No mesmo seguimento, o robô consegue alertar o utilizador, dizendo que os medicamentos estão a terminar ("Conheça Pillo, o robô que promete cuidar da sua saúde," n.d.).

¹⁶ Disponível em: <https://www.pillohealth.com/>



Figura 22 - Lembrete do horário da toma do medicamento gerado pelo robô pillo.¹⁷

Este robô é complementado com a aplicação móvel, apenas existente para iOS neste momento. Esta aplicação permite a familiares receber uma notificação aquando do esquecimento de algum dos medicamentos, para além de possibilitar o estabelecimento de uma videochamada entre o robô e a aplicação. É através da aplicação que pode ser configurado o regime de medicação dos utilizadores.

Concluindo, com a utilização deste robô o utilizador é ajudado a cumprir o seu regime de medicação. A utilização deste robô possui como vantagens o facto de este emitir um lembrete do horário da toma do medicamento e a sua dispensa no horário correto. Ainda assim, caso o utilizador não efetue a toma do medicamento este robô consegue informar um familiar para o alertar sobre a situação. No entanto, existem ainda alguns problemas que poderão surgir e estes estão relacionados com a inexistência de um sistema de locomoção. Entre estes problemas encontra-se o facto de o utilizador não estar na mesma divisória do robô ou o facto de este possuir mobilidade reduzida e ter de se deslocar até ao robô e posteriormente até à água para efetuar a toma do medicamento.

¹⁷ Disponível em: <https://www.pillohealth.com/>

3.3.2. Lembrete de remédios - Medisafe

Lembrete de remédios - Medisafe é uma aplicação desenvolvida pela empresa Medisafe para as plataformas Android e iOS. O seu principal objetivo foca-se em lembrar o utilizador dos horários a que deve tomar a sua medicação.

Para o cumprimento do seu principal objetivo, esta aplicação informa o utilizador através de uma notificação, esta notificação é gerada no horário da toma de um medicamento (figura 23). A gestão do stock de medicação é outra funcionalidade desta aplicação, esta funcionalidade permite ao utilizador estipular quando gostaria de receber notificação para reabastecimento de medicamentos consoante a quantidade de medicamentos disponível.

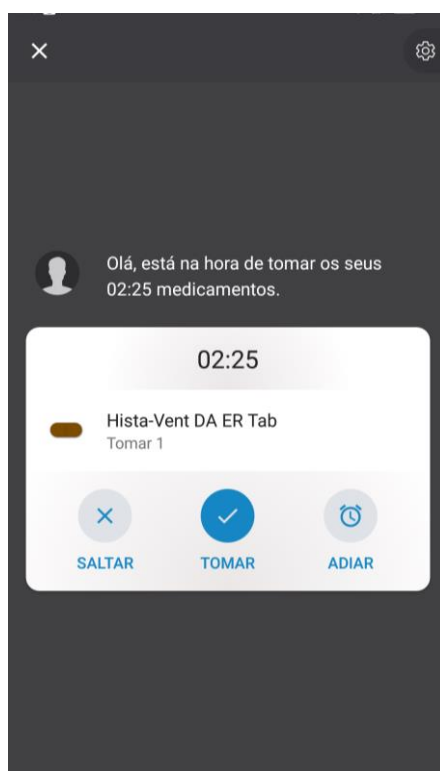


Figura 23 - Notificação recebida pelo utilizador quando chega ao horário da toma do medicamento.

Após uma pesquisa na loja de aplicações, é possível verificar que existem inúmeras aplicações com o mesmo objetivo, porém esta aplicação destaca-se no meio de 46 outras aplicações analisadas pelo especialista em tecnologia de consumo, David Pogue (Pogue, 2017).

Uma das vantagens visíveis nesta aplicação, e que a difere das outras concorrentes, é a fácil utilização e a sua personalização (Pogue, 2017). A própria aplicação tem a função de completar o nome do medicamento, sem ser necessário o utilizador escrever o nome completo, o que facilita a inserção de medicamentos cujos nomes são mais difíceis de escrever e perceber (figura 24). De realçar também o facto do utilizador poder personalizar a forma e a cor do medicamento, permitindo assim, o melhor reconhecimento do medicamento a tomar. A confirmação da toma da medicação é feita pelo próprio utilizador, que ao receber a notificação à hora da toma basta um toque para confirmar que tomou o medicamento indicado.

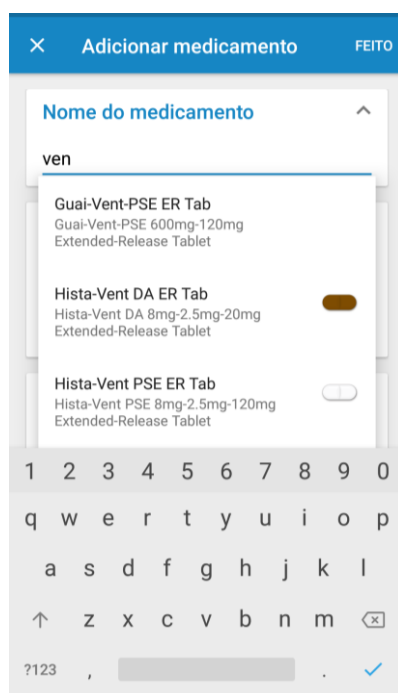


Figura 24 - Funcionalidade de completar o nome do medicamento no processo de inserção dos medicamentos a tomar.

Estes tipos de aplicações realmente facilitam a tarefa da toma de medicação, porém, uma das desvantagens encontradas é o facto de o utilizador ter que procurar qual o medicamento certo a tomar, bem como dirigir-se até ao local onde se encontra a água para fazer a sua ingestão. Outro problema deste tipo de soluções é o facto do utilizador não ter acesso ao seu smartphone a todo o momento, o que pode fazer com que falhe a toma da medicação no horário correto.

4. Materiais e métodos

No desenvolvimento deste capítulo, são apresentados e descritos todos os materiais utilizados, desde o robô utilizado, os algoritmos de visão por computador e as ferramentas de desenvolvimento.

4.1. Robô NAO

A parte experimental da dissertação foi efetuada com o robô humanoide NAO, desenvolvido pela *SoftBank Robotics*.

O robô NAO (figura 25) que vem sendo desenvolvido desde 2006, encontra-se atualmente na sua quinta versão. Segundo a *SoftBank Robotics*, NAO, é um robô interativo para companhia. Já foram vendidos cerca de 10000 robôs NAO (“Who is NAO?,” n.d.), com particular destaque no sucesso na área da investigação e educação (“About us,” n.d.).

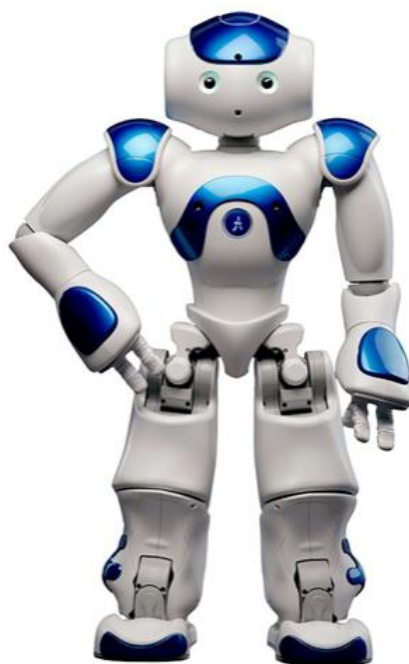


Figura 25 - Robô NAO desenvolvido pela *SoftBank Robotics*.¹⁸

¹⁸ Disponível em: <https://www.juguetronica.com/robot-nao>

4.1.1. Hardware

O robô NAO mede aproximadamente 58 centímetros e pesa 5,4 kg, (“NAO - Construction,” n.d.) este tem uma estrutura corporal semelhante a um humano e está equipado com vários componentes para a visão e percepção do ambiente. A sua última versão é controlada por um CPU ATOM de 1,6 GHz, localizado na cabeça do robô (“Motherboard,” n.d.), que executa um sistema operativo próprio, OpenNAO, baseado numa distribuição Linux (Gentoo) (“NAOqi OS - Getting started,” n.d.).

De modo a que o robô possua um comportamento o mais semelhante possível a um humano, ele inclui várias características que lhe permitem movimento, percepção e comunicação.

Movimento: Possui 25 motores que controlam 26 articulações de forma a permitir ao robô efetuar movimentos humanos básicos (“Joints — Aldebaran 2.1.4.13 documentation,” n.d.).

Percepção: Equipado com vários sensores de toque, situados na cabeça, mãos e pés, este robô possui a capacidade de reagir ao toque. Possui ainda sensores de ultrassons que permitem ao robô ter a percepção do ambiente e orientar-se no mesmo (“NAO - Technical overview,” n.d.).

Comunicação: A comunicação verbal entre o robô e o humano é efetuado devido aos quatro microfones na cabeça do robô (figura 26) (“Microphones — Aldebaran 2.1.4.13 documentation,” n.d.) e dois altifalantes (figura 27) (“Loudspeakers — Aldebaran 2.1.4.13 documentation,” n.d.)

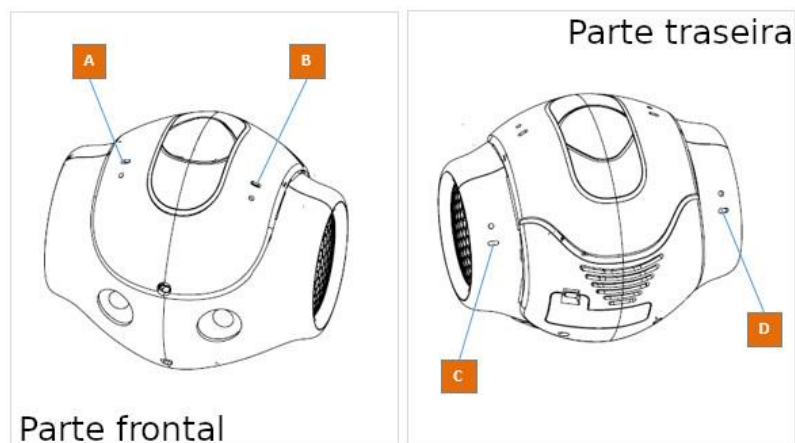


Figura 26 - Posição dos quatro microfones na cabeça do robô. Adaptado de (“Loudspeakers — Aldebaran 2.1.4.13 documentation,” n.d.).

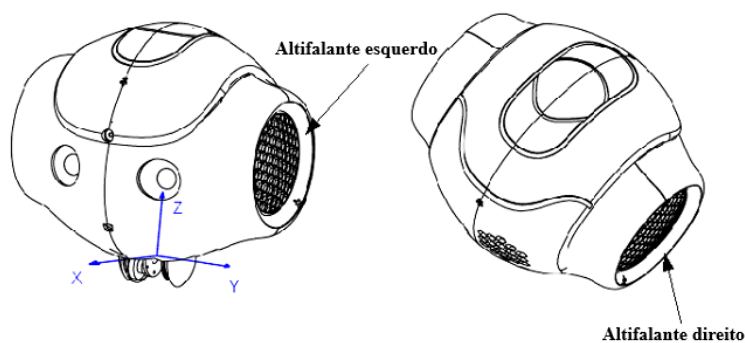


Figura 27 - Posição dos dois altifalantes na cabeça do robô. Adaptado de (“Microphones — Aldebaran 2.1.4.13 documentation,” n.d.).

De forma a suportar todas as atividades desempenhadas pelo NAO, o robô é equipado com uma bateria 48.6 Wh que permite entre 60 a 90 minutos de utilização (“NAO - Battery — Aldebaran 2.1.4.13 documentation,” n.d.).

Relativamente à conectividade, o robô possui Wi-Fi, porta Ethernet e uma porta USB usada para ligação de Kinect, sensor Asus 3D ou Arduino (“Connectivity — Aldebaran 2.1.4.13 documentation,” n.d.).

Após serem salientadas sucintamente as principais características deste robô, na tabela 1 são apresentadas mais detalhadamente as características dos vários componentes do robô.

Tabela 1 - Características detalhadas dos componentes do robô NAO.

Corpo	
Altura (mm)	574
Profundidade (mm)	311
Largura (mm)	275
Peso (Kg)	5.4
Material	ABS-PC/PA-66/XCF-30
Placa-mãe	
Processador	ATOM Z530 1.6 GHz
RAM	1 GB
Memória	2 GB memória flash e 8 GB Micro SDHC
Bateria	
Energia	48.6 Wh
Duração de carregamento	Aproximadamente 3 horas
Autonomia	Aproximadamente 60 minutos de uso ativo ou 90 minutos de uso normal
Conectividade	
Rede	IEEE 802.11 a/b/g/n e 1×RJ45 - 10/100/1000
Dispositivos externos	1 x porta USB

Articulações

De modo a que o robô possua um comportamento o mais aproximado possível a um humano é necessário que este efetue movimentos básicos. O robô NAO, possui 25 motores que controlam 26 possíveis movimentos das articulações. Estes 26 possíveis movimentos são identificados por nomes específicos que estão apresentados na figura 28.

Uma articulação é responsável pela ligação de duas partes do corpo do robô, aquando do movimento, a parte mais próxima do tronco é considerada fixa e a outra parte gira em torno do eixo da articulação.

Existem três movimentos possíveis nas articulações. A diferença entre estes foca-se no eixo em torno do qual acontece a rotação. O movimento *Roll* acontece em torno do eixo X, o movimento *Pitch* acontece em torno do eixo Y e o movimento *Yaw* em torno do eixo Z como ilustrado na figura 29 (“Joints — Aldebaran 2.1.4.13 documentation,” n.d.).

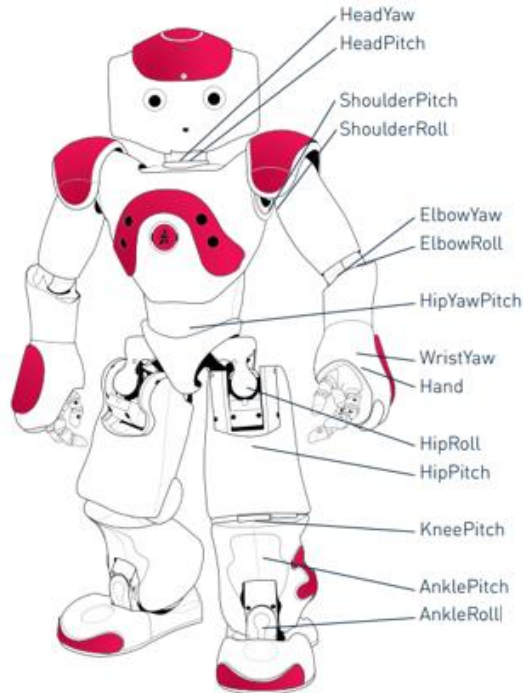


Figura 28 - Imagem com o nome de todas as articulações e respetiva localização.¹⁹

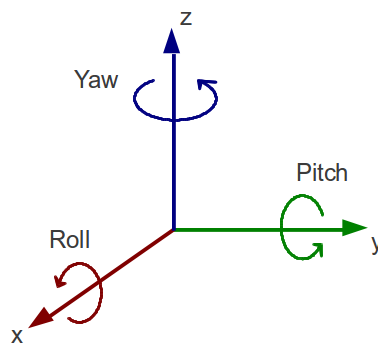


Figura 29 - Imagem explicativa dos movimentos possíveis e a respetiva associação ao eixo de rotação.²⁰

Todos os possíveis movimentos das articulações podem ser controlados entre um intervalo de determinados valores. Na tabela 2 estão descritas todas as possibilidades de movimentação de cada articulação e os respetivos valores permitidos, medidos em graus.

¹⁹ Disponível em: <https://www.robotlab.com/support/all?hsFormKey=5bc13c89364aa1125a6af57473cd7f25>

²⁰ Disponível em: http://doc.aldebaran.com/2-1/family/robots/joints_robot.html

Tabela 2 - Informações de todas as articulações e respectivos valores mínimos e máximos aceites.

Nome da articulação	Intervalo de valores possíveis (°)
Cabeça	
HeadYaw	-119.5 até 119.5
HeadPitch	-38.5 até 29.5
Braço	
RshoulderPitch	-119.5 até 119.5
RshoulderRoll	-76 até 18
RElbowYaw	-119.5 até 119.5
RElbowRoll	2 até 88.5
RWristYaw	-114.5 até 104.5
Rhand	Abrir ou fechar
LshoulderPitch	-119.5 até 119.5
LshoulderRoll	-18 até 76
LElbowYaw	-119.5 até 119.5
LElbowRoll	-88.5 até -2
LWristYaw	-104.5 até 104.5
LHand	Abrir ou fechar
pelve	
LhipYawPitch	-65.62 até 42.44
RhipYawPitch*	-65.62 até 42.44
Perna	
LhipRoll	-21.74 até 45.29
LHipPitch	-88.0 até 27.73
LKneePitch	-5.29 até 121.04
LAnklePitch	-68.15 até 52.86
LAnkleRoll	-22.79 até 44.06
RhipRoll	-45.29 até 21.74
RHipPitch	-88.0 até 27.73
RKneePitch	-5.90 até 121.47
RAnklePitch	-67.97 até 53.40
RAnkleRoll	-44.06 até 28.80

* Fisicamente, *RhipYawPitch* e *LhipYawPitch* são apenas um motor e não podem ser independentemente controlados. *LhipYawPitch* possui sempre prioridade caso existam conflitos entre ordens para essa articulação e *RhipYawPitch*.

Câmaras

O robô NAO possui duas câmaras idênticas localizadas na parte frontal da cabeça do robô. As duas câmaras providenciam aquisição de vídeo até à resolução máxima de 1280x960 à taxa de 30 fps. Cada uma das câmaras possui uma limitação no ângulo de visão de 60,97° na horizontal e 47,64° na vertical, como mostrado na figura 30 (“NAO - Video camera — Aldebaran 2.1.4.13 documentation,” n.d.).

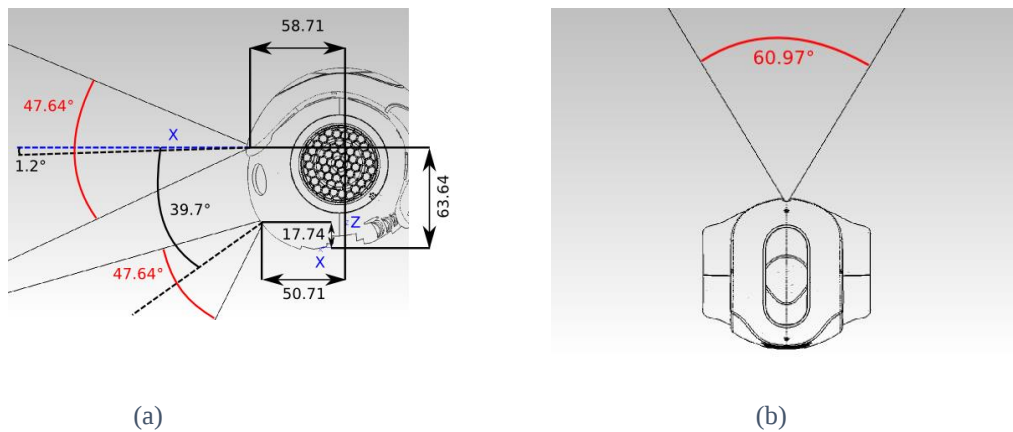


Figura 30 – Imagem demonstrativa do ângulo de visão das duas câmaras do robô NAO.²¹ (a) – Ângulo de visão das duas câmaras na vertical. (b) – Ângulo de visão das duas câmaras na horizontal.

Ambas as câmaras possuem vários parâmetros que podem ser personalizados, como por exemplo, o brilho, contraste ou a resolução. Na tabela 3 são apresentadas algumas das características, com a parametrização permitida, os intervalos de valores possíveis, assim como os valores por defeito.

²¹ Disponível em: http://doc.aldebaran.com/2-1/family/robots/video_robot.html

Tabela 3 - Apresentação de algumas das características das câmaras do NAO e respetiva parametrização.

Caraterística	Valor mínimo	Valor máximo	Valor por defeito
Brilho	0	255	55
Contraste	16	64	32
Saturação	0	255	128
Tonalidade	-180	180	0
Resolução (*)	kQVGA	k4VGA	kQVGA
FPS	1	30	5
Selecionar câmara (**)	0	1	0

(*)

Tabela 4 - Parametrização da resolução das câmaras e respetivas resoluções associadas.

Nome ID parâmetro	Valor id	Resolução
AL::kQQQQVGA	8	40 x 30 px
AL::kQQQVGA	7	80 x 60 px
AL::kQQVGA	0	160 x 120 px
AL::kQVGA	1	320 x 240 px
AL::kVGA	2	640 x 480 px
AL::k4VGA	3	1280 x 960 px

(**)

Tabela 5 - Identificadores associados a cada uma das câmaras do NAO.

Câmara	ID câmara
Superior	1
Inferior	0

4.1.2. Software

OpenNao

O sistema operativo do robô NAO, intitulado de OpenNao, é baseado na distribuição linux Gentoo. Este foi desenvolvido especificamente para atender às necessidades do robô (“OpenNAO - NAO OS — NAO Software 1.14.5 documentation,” n.d.).

O sistema operativo fornece diversas aplicações e bibliotecas de forma a permitir que o robô seja programável (“OpenNAO - NAO OS — NAO Software 1.14.5 documentation,” n.d.).

NaoQi

NaoQi é o middleware responsável por controlar o hardware do robô. A utilização deste middleware permite a programação do robô e possibilita combater algumas das necessidades comuns na robótica como o paralelismo, sincronização, eventos e recursos (“NAOqi Framework — NAO Software 1.14.5 documentation,” n.d.).

O NaoQi permite a comunicação homogénea entre vários módulos (movimento, áudio, ect), programação homogénea e partilha homogéneo de informações (“NAOqi Framework — NAO Software 1.14.5 documentation,” n.d.).

Este framework é multiplataforma, multilinguagem e permite introspeção.

O NaoQi é responsável por gerir vários proxies. Um proxy é um objeto que representa um módulo e se comporta como este, ou seja, proxies intercedem na interação com os módulos de periferia, como por exemplo a câmara ou a memória. Isto é possível pois estes estão ligados ao broker, que é executado no robô e permite que o robô receba comandos através da rede, o que possibilita o controlo do robô através de um computador remoto (“NAOqi Framework — NAO Software 1.14.5 documentation,” n.d.). Esta interação está demonstrada na figura 31.

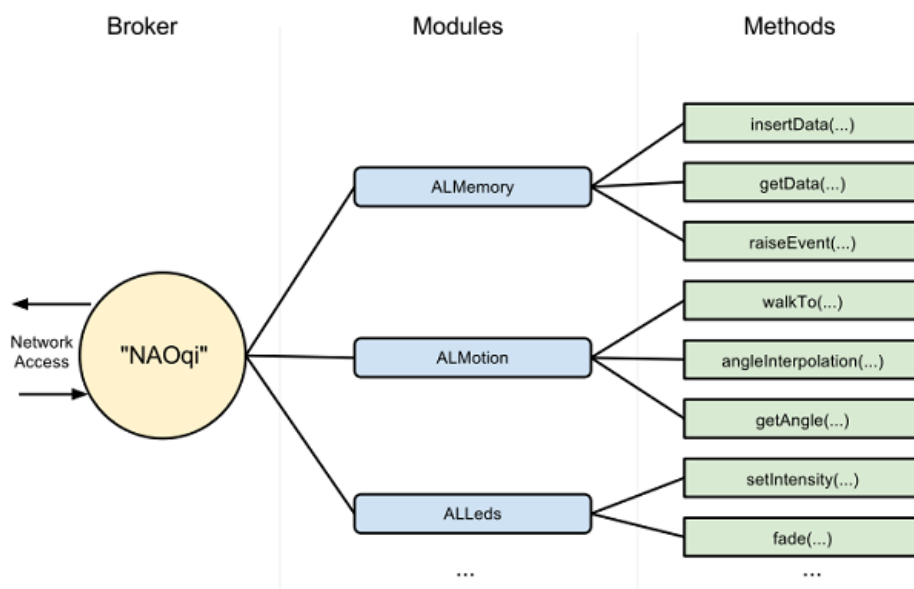


Figura 31 - Esquema representativo do funcionamento do NaoQi e as interações entre os diversos componentes.

De acordo com o diagrama da figura 31, o NaoQi permite que o robô seja controlado utilizando os métodos disponíveis nos vários módulos. Existem duas formas de executar métodos de um módulo, localmente ou remotamente. Os módulos executados localmente são executados diretamente no robô, não havendo necessidade de existir outro sistema computacional. Por outro lado, os módulos executados remotamente são executados num sistema computacional e comunicam através da rede as ações a executar. Alguns dos métodos são exclusivos para o uso remoto (“NAOqi Framework — NAO Software 1.14.5 documentation,” n.d.).

A programação do robô pode ser feita através do software próprio, o Choregraphe, ou através dos SDK, disponibilizados para as linguagens Python, C++, Java e javascript (“Programming — Aldebaran 2.1.4.13 documentation,” n.d.). Portanto, existem diversas linguagens que podem ser utilizadas tanto no Choregraphe como através dos SDK fornecidos, todas estas linguagens suportadas pelo NAO estão listadas na tabela 6.

²² Disponível em: <http://doc.aldebaran.com/1-14/dev/naoqi/index.html>

Tabela 6 - Tabela das linguagens de programação aceites pelo robô NAO. Baseado em (“SDKs – Aldebaran 2.5.10.8 documentation”, n.d.).

Linguagens de programação	Executar aplicações		Choregraphe	
	Computador	Robô	Criar	Editar
Python	☑	☑	☑	☑
C++	☑	☑	☒	☒
Java	☑	☒	☒	☒
JavaScript	☑	☑	☑	☒

☑ Suportado

☒ Não suportado

Choregraphe

O Choregraphe é um software multiplataforma que permite a criação de animações e comportamentos, monitorizar, controlar o robô NAO e testar os comportamentos desenvolvidos num robô real. Este software permite através de programação por blocos desenvolver alguns comportamentos ou ações sem que seja necessário a implementação de código. Algumas destes comportamentos ou ações podem ser, por exemplo, o robô enviar e-mails ou dançar. Embora a programação seja feita por blocos, é possível editar ou criar o próprio bloco de código. No entanto, comportamentos desenvolvidos no Choregraphe possuem um tempo de execução mais elevado do que quando são desenvolvidos em C++ ou Python (“Choregraphe overview — NAO Software 1.14.5 documentation,” n.d.).

Este software possui uma interface simples como pode ser visto na figura 32, a área de trabalho está situada no centro da janela, do lado esquerdo contém os ficheiros do projeto e mais a baixo os vários blocos que se poderão arrastar para a área de trabalho. Enquanto do lado direito é possível controlar a posição do robô (sentado, levantado, etc), monitorizar a câmara do NAO em tempo real e executar aplicações existentes no robô.

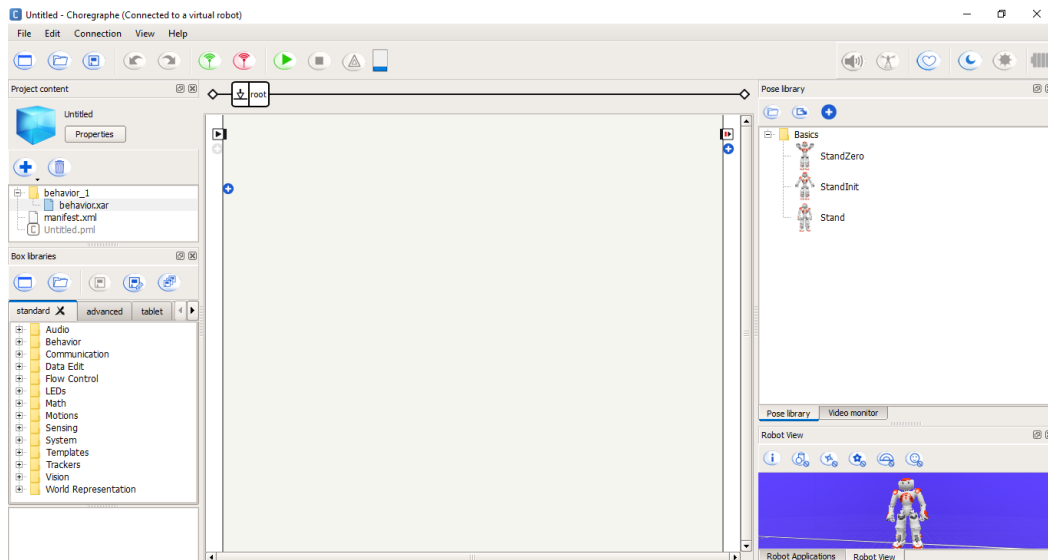


Figura 32 - Layout principal do programa Choregraphe.

4.2. Algoritmos de visão por computador

4.2.1. Local Binary Patterns Histogram (LBPH)

O algoritmo LBP foi inicialmente proposto com a finalidade de analisar a textura através da comparação dos pixels centrais com os pixels vizinhos (Ojala, Pietikäinen, & Harwood, 1996). O LBP pelo facto de ser simples mas poderoso para descrever estruturas locais, tem sido explorado em amplas áreas, (Huang, Shan, Ardabilian, Wang, & Chen, 2011) como por exemplo, a análise de movimento de objetos (Heikkila & Pietikainen, 2006), reconhecimento facial (Ahonen, Hadid, & Pietikainen, 2006) e detecção facial (Hadid, Pietikainen, & Ahonen, 2004).

Segundo (Ahonen et al., 2006), as imagens faciais podem ser consideradas micro padrões como áreas planas, pontos, contornos e bordas que podem ser descritos pelo LBP. Por esse motivo LBP pode ser utilizado não só para a detecção da face, mas também para identificar ou autenticar a face.

O LBP começa por dividir a imagem em blocos de 3x3 para se analisar a vizinhança à volta do pixel central de cada bloco. Como se mostra na figura 33, o valor do pixel central é comparado com o valor dos restantes oito vizinhos, utilizando como *threshold* o seu valor de intensidade. Como resultado dessa comparação, é atribuído o valor de “1” aos pixels vizinhos que possuem um valor superior ou igual ao valor de *threshold* e

aos restantes é atribuído o valor “0” (Ojala et al., 1996). Após a atribuição dos valores a todos os pixéis vizinhos os valores são concatenados, iniciando pelo pixel do canto superior esquerdo no sentido dos ponteiros do relógio, obtendo assim o código LBP do pixel central (Huang et al., 2011).

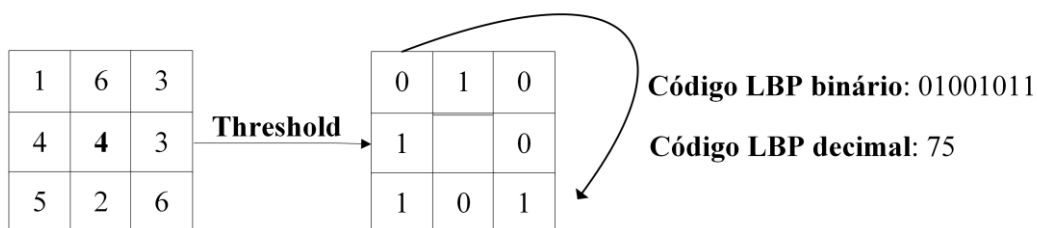


Figura 33 – Exemplo de extração do código LBP. Adaptado de (Huang et al., 2011).

Após o processo anterior ser efetuado em todos os pixéis, é obtida uma nova imagem que realça as características faciais (Figura 34(b)).

Em seguida aplica-se o LBPH que divide a imagem resultante do LBP em blocos (figura 34(c)) e em cada um calcula o histograma da ocorrência dos códigos LBP decimais. Após a criação de todos os histogramas parciais (figura 34(d)), eles são concatenados num histograma que representa as principais características da imagem inicial (figura 34(e)).

Na figura 34 estão representados os passos para a criação do histograma concatenado.

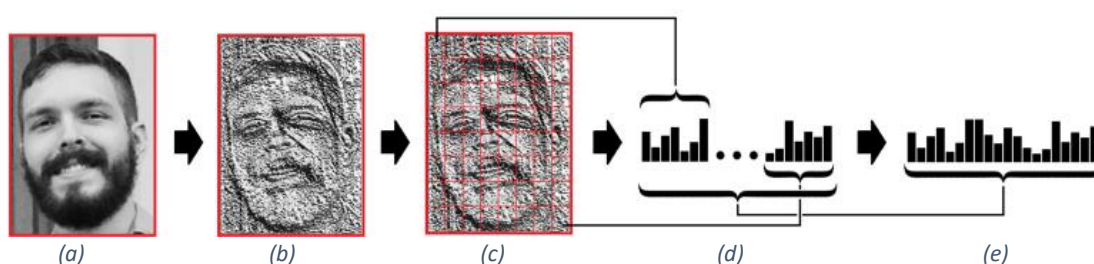


Figura 34 – Exemplo das etapas de criação do histograma concatenado.²³ (a) Imagem original. (b) Imagem com as características realçadas. (c) Divisão da imagem com as características realçadas em blocos. (d) Criação de histogramas de cada bloco. (e) Concatenação de todos os histogramas parciais num só histograma.

A detecção facial usando LBP é uma extensão do algoritmo apresentado por Viola e Jones (Viola & Jones, 2001) e usa os códigos LBP obtidos em vez de características *Haar*,

²³ Disponível em: <https://towardsdatascience.com/face-recognition-how-lbph-works-90ec258c3d6b>

obtendo uma solução computacionalmente mais leve e conseqüentemente um processamento mais rápido.

Para efetuar o reconhecimento facial através do LBPH, inicialmente calculam-se os histogramas das imagens de treino, obtendo-se um histograma padrão para cada classe.

Na fase de teste, também se calcula o LBPH de cada imagem em que se pretende fazer o reconhecimento facial. Em seguida é calculada a diferença entre o histograma da imagem de treino e o da imagem de teste que serve de indicador da confiança e pode ser calculada através da fórmula:

$$confiança = \sqrt{\sum_{i=1}^n (histograma1_i - histograma2_i)^2}$$

O classificador de distância mínima é usado para atribuir a classe à imagem de teste. A aceitação do resultado da classificação está condicionada por um valor de *threshold* que se estabelece para aumentar a confiança no processo de reconhecimento.

4.2.2. SIFT

O Scale-invariant feature transform (SIFT) é um algoritmo de visão por computador publicado por (D.G. Lowe, 1999) utilizado para detetar e descrever caraterísticas locais em imagens. As caraterísticas extraídas por este algoritmo são invariáveis à escala e rotação e são utilizados para fornecer uma classificação robusta mesmo na presença de distorções, mudança da perspectiva da imagem, ruído e mudanças de iluminação (David G Lowe, 2004).

Segundo (David G Lowe, 2004), o SIFT consiste principalmente em quatro etapas (figura 35). Estas etapas, ordenadas de forma sequencial, são: deteção de extremos, localização de pontos-chave, atribuição da orientação e descrição dos pontos-chave.

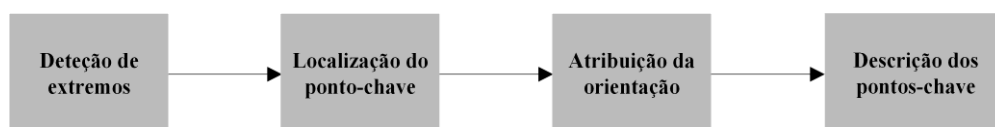


Figura 35 - Etapas do algoritmo SIFT.

Deteção de extremos: Esta etapa consiste em procurar pontos que não possuam variações com mudanças de escala da imagem e rotações do objeto, possibilitando a detecção desses pontos independentemente da distância da câmara ao objeto. Tal objetivo é alcançado procurando características estáveis em diferentes escalas, utilizando a diferença de funções Gaussianas. Esta diferença permite eliminar ruídos indesejados e destacar características. Após esse processo, é efetuada a detecção de extremos (máximos ou mínimos) comparando-os com os oito pixéis vizinhos da imagem e com os nove pixéis vizinhos da imagem de escala superior e inferior (ver figura 36). O pixel analisado é selecionado apenas se o seu valor for superior ou inferior a todos os outros pixéis vizinhos.

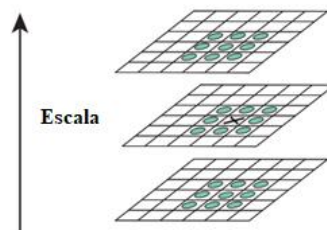


Figura 36 – Processo de detecção de extremos comparando o pixel identificado com “X” com os 8 pixéis vizinhos e com os 9 pixéis vizinhos de escala superior e inferior. Adaptado de (David G Lowe, 2004).

Localização do ponto-chave: Todos os extremos detetados na etapa anterior são possíveis pontos-chave. Por isso, nesta etapa o objetivo é calcular a localização exata, escala e proporção de curvaturas de todos os extremos permitindo rejeitar pontos com baixo contraste (sensível ao ruído) ou pontos mal localizados ao longo de uma aresta.

Atribuição da orientação: Nesta etapa é atribuída uma ou mais direções a cada ponto-chave por forma a que cada um possa ser representado pela sua orientação, proporcionando invariância às rotações da imagem. Neste processo são considerados pixéis vizinhos e é construído um histograma dividido em 36 orientações (cobrindo os 360°). A cada pixel é associado a magnitude do gradiente como “peso” de forma a que pixéis com maior magnitude tenham maior importância. O máximo do histograma, assim como as orientações superiores a 80% são consideradas para calcular a orientação do ponto chave.

Descrição dos pontos-chave: Nesta etapa, é calculado um descritor para a região da imagem que seja invariante às alterações na iluminação ou perspectiva.

De forma a calcular o descritor do ponto chave são considerados os pixels vizinhos num bloco de 16x16, dividido em sub-blocos de 4x4. Para cada sub-bloco é criado um histograma de orientação com 8 posições. Por isso, no total existem 128 posições de histogramas (figura 37).

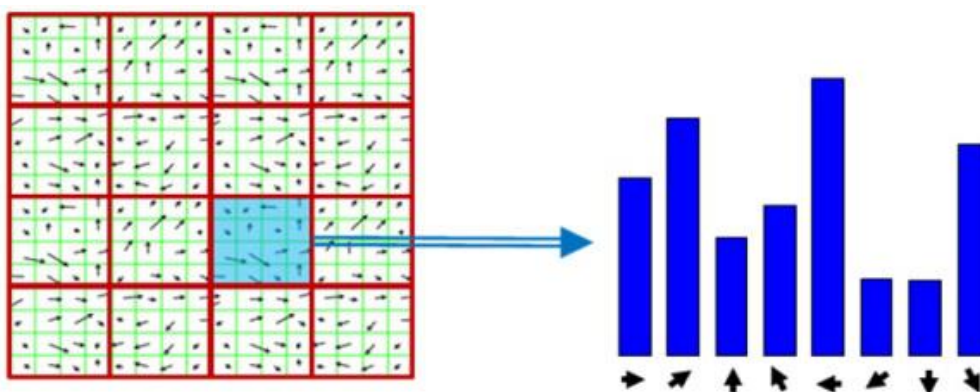


Figura 37 – Histograma de orientação de um bloco 4x4.²³

Após esse processo os histogramas são concatenados num só e com os valores do histograma é criado um vetor de 128 posições que representa o descritor do ponto chave (figura 38).

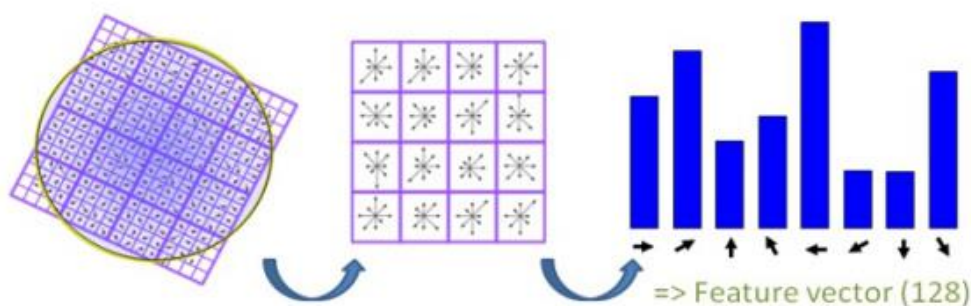


Figura 38 - Demonstração da criação de um vetor descritor através da concatenação do histograma de orientação.²³

²⁴ Disponível em: <https://gilscvblog.com/2013/08/18/a-short-introduction-to-descriptors/>

4.3. Ferramentas de desenvolvimento

4.3.1. Python

O Python é uma linguagem programação de alto nível orientada a objetos e foi criada por Guido Van Rossum no ano de 1991 (“Computer Programming for Everybody | Python.org,” n.d.). Esta linguagem de programação apareceu com a ideologia de “uso geral”, ou seja, desenvolvida para ser fácil aprender e usar (“General Python FAQ — Python 2.7.15 documentation,” n.d.). Esta linguagem de programação é amplamente utilizada nas áreas de programação numérica, inteligência artificial, processamento de imagens, biologia e outros.

O Python é grátis, opensource e compatível com sistemas operativos Windows, Linux e MacOS (“General Python FAQ — Python 2.7.15 documentation,” n.d.).

4.3.2. MySql

O MySQL é um sistema de gestão de base de dados (SGBD) que utiliza a linguagem padrão SQL (Niederauer, 2008). Este SGBD surgiu em 1995 através dos programadores David Axmark, Allan Larsson e Michael Widenius (Buytaert, 2010).

Devido aos bons resultados, entre os quais a rapidez de acesso, o MySQL foi bem aceite, sofrendo várias melhorias e novas funcionalidades ao longo dos tempos e é largamente utilizado em aplicações para a internet como o website da google e da NASA (Niederauer, 2008).

Algumas das vantagens/características do MySQL são:

- Grátis e opensource
- Portabilidade: o MySQL pode ser utilizado em diferentes sistemas operativos, plataformas e compiladores. Além disso, possui API para utilização em várias linguagens, como Java, Python, PHP, etc.
- Limite de conexões: MySQL permite um número ilimitado de utilizadores simultâneos.
- Velocidade: Alta velocidade de execução de comandos.
- Segurança: Fácil e eficiente controlo de privilégios de utilizadores.

4.3.3. Python SDK

Através do Python SDK é possível desenvolver scripts em Python que podem ser executados remotamente ou no próprio robô (“Python SDK — NAO Software 1.14.5 documentation,” n.d.).

Após a instalação do Python e do SDK do robô, é possível utilizar todos os métodos dos vários módulos disponibilizados pelo NaoQi. Para isso, é necessário importar todos os módulos existentes no NaoQi, sendo criado um proxy para o módulo pretendido. Por fim, é possível executar todos os métodos do módulo à qual a proxy se refere. Na figura 39 é apresentado um exemplo dos passos necessários para o robô dizer a frase “Olá mundo”.

```
from naoqi import ALProxy
tts = ALProxy("ALTextToSpeech", "IP do robô", 9559)
tts.say("Olá mundo")
```

Figura 39 – Exemplo de funcionamento do python SDK no robô NAO.

4.3.4. OpenCV

O OpenCV é uma biblioteca opensource com a implementação de inúmeros algoritmos de visão por computador e machine learning. Esta biblioteca possui mais de 2500 algoritmos otimizados desde algoritmos clássicos até algoritmos de última geração (“About - OpenCV library,” n.d.).

O OpenCV possui interfaces para C++, Python, Java e Matlab e é multiplataforma, suportando Windows, Linux, Android e Mac OS (“About - OpenCV library,” n.d.).

4.3.5. Schedule

Esta biblioteca permite executar funções Python periodicamente em intervalos pré-determinados através de uma sintaxe simples e amigável (“schedule — schedule 0.4.0 documentation,” n.d.). Na figura 40 é apresentado um excerto de código demonstrativo da sintaxe da biblioteca.

```

import schedule
import time

def job():
    print("I'm working...")

schedule.every(10).minutes.do(job)
schedule.every().hour.do(job)
schedule.every().day.at("10:30").do(job)
schedule.every().monday.do(job)
schedule.every().wednesday.at("13:15").do(job)

while True:
    schedule.run_pending()
    time.sleep(1)

```

Figura 40 - Exemplo de funcionamento da biblioteca Schedule.

4.3.6. Numpy

O NumPy é uma biblioteca usada na área da computação numérica (“NumPy — NumPy,” n.d.). Esta biblioteca permite uma sintaxe semelhante ao Matlab com uma eficiência mais elevada e a expressividade característica da linguagem (“Numpy - PyScience-Brasil,” n.d.).

Esta biblioteca possui um poderoso objeto para arranjos multidimensionais, objeto para cálculo de matrizes, ferramentas de álgebra linear, transformadas de Fourier básicas e capacidades de gerar números aleatórios (“Numpy - PyScience-Brasil,” n.d.).

4.3.7. PILLOW

Pillow é um fork amigável da PIL (Python Imaging library) (“Pillow — Pillow (PIL Fork) 5.2.0 documentation,” n.d.)

Esta biblioteca permite adicionar recursos ao interpretador Python que podem ser utilizados no processamento de imagens, oferecendo suporte extensivo a formatos de ficheiros, representação interna eficiente e recursos de processamento de imagem poderosos (“Overview — Pillow (PIL Fork) 5.2.0 documentation,” n.d.).

5. Implementação

Este capítulo descreve a integração dos algoritmos de visão por computador no robô NAO de forma a atingir os objetivos anteriormente apresentados.

Embora tenha sido utilizado um robô específico, a abordagem desenvolvida não é exclusiva para este robô, podendo ser adaptada a qualquer robô que disponha de um sistema de locomoção e visão por computador.

O principal foco desta dissertação remete-se ao reconhecimento facial e de objetos, não sendo desenvolvida a parte da cinemática. Toda a parte relativa ao movimento é feita de modo imutável não considerando a variação do cenário. Desta forma, o produto desenvolvido está preparado para um cenário específico descrito mais à frente.

O desenvolvimento foi totalmente feito em Python, recorrendo ao Python SDK para controlar o robô. O reconhecimento de faces e objetos foi efetuado recorrendo à biblioteca OpenCV.

5.1. Arquitetura do sistema desenvolvido

Na figura 41 é apresentado um diagrama simplificado do sistema implementado para apoio à toma de medicação em pessoas idosas.

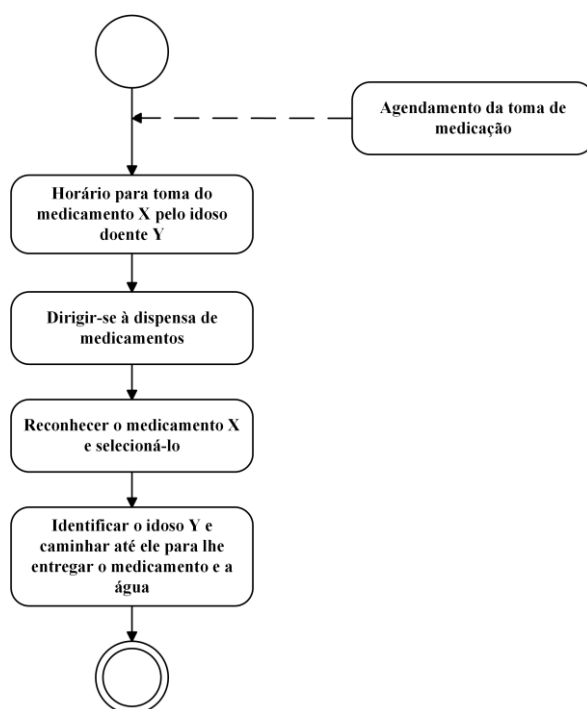


Figura 41 - Diagrama do sistema implementado para apoio à toma de medicação.

De forma a suportar todas as informações dos medicamentos, idosos e horários da toma dos respectivos medicamentos foi criada uma base de dados em MySQL.

Uma aplicação, desenvolvida em Python, é responsável por controlar todas as ações do robô. Esta aplicação faz a ligação à base de dados, controla a aquisição e o processamento das imagens e faz a verificação dos horários para a toma dos medicamentos de cada idoso. Estas interações entre os diferentes componentes do sistema desenvolvido estão representadas graficamente na figura 42.

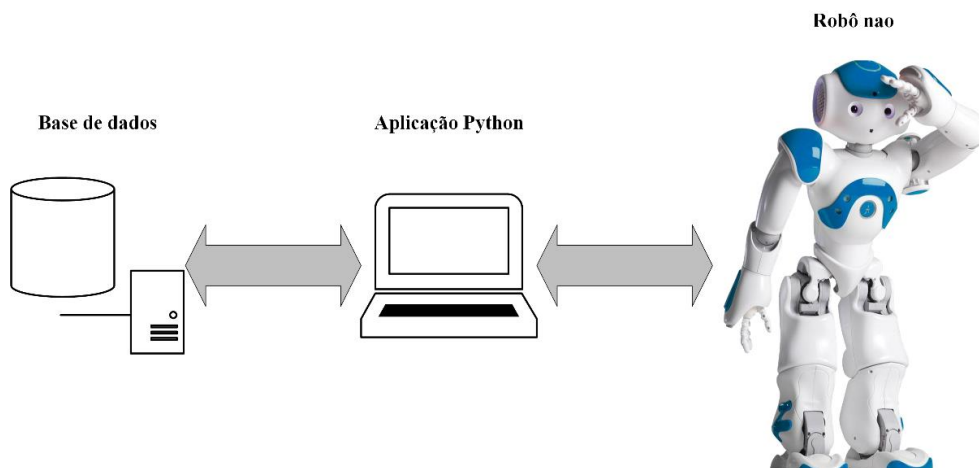


Figura 42 - Esquema da arquitetura do sistema desenvolvido e as respectivas interações entre os três componentes do mesmo.

O sistema pode-se dividir em duas fases: fase de planeamento e a fase de execução. Na fase de planeamento são adquiridos todos os dados presentes na base de dados e são agendadas as ações a realizar nos horários da toma de medicamentos. Neste processo é utilizado a biblioteca *schedule*. Na figura 43 é apresentado um excerto de código que mostra como este processo é feito.

```
def carregarHorarios():
    cursor = bd.bd.cursor()
    cursor.execute("SELECT * FROM horario")

    rows = cursor.fetchall()
    for row in rows:
        idMedicamento= int(str(row).split(",")[0].replace(",",""))
        idIdoso= int(str(row).split(",")[1])
        horas= str(row).split('\\')[1].split('.')[0]
        minutos= str(row).split('\\')[1].split('.')[1]
        schedule.every().day.at(horas+":"+minutos).do(lambda: tomarMedicamento(idMedicamento,idIdoso))
```

Figura 43 - Função desenvolvida para a leitura de todos os horários de toma de medicamento e respetivo agendamento dos mesmos utilizando a biblioteca Schedule.

Após todos os horários presentes na base de dados estarem planejados para serem executados entramos na segunda fase do sistema. A segunda fase deste sistema consiste na execução de todo o processo na toma da medicação aquando do horário planeado na fase anterior. Nesta fase é incluída a parte do movimento do robô, reconhecimento dos medicamentos e reconhecimento facial.

Este processo da toma da medicação é efetuado de forma sequencial desde o reconhecimento do medicamento até à entrega do mesmo ao idoso correto. Na figura 44 estão representadas através de um diagrama todas as etapas e as condições para o correto cumprimento da tarefa.

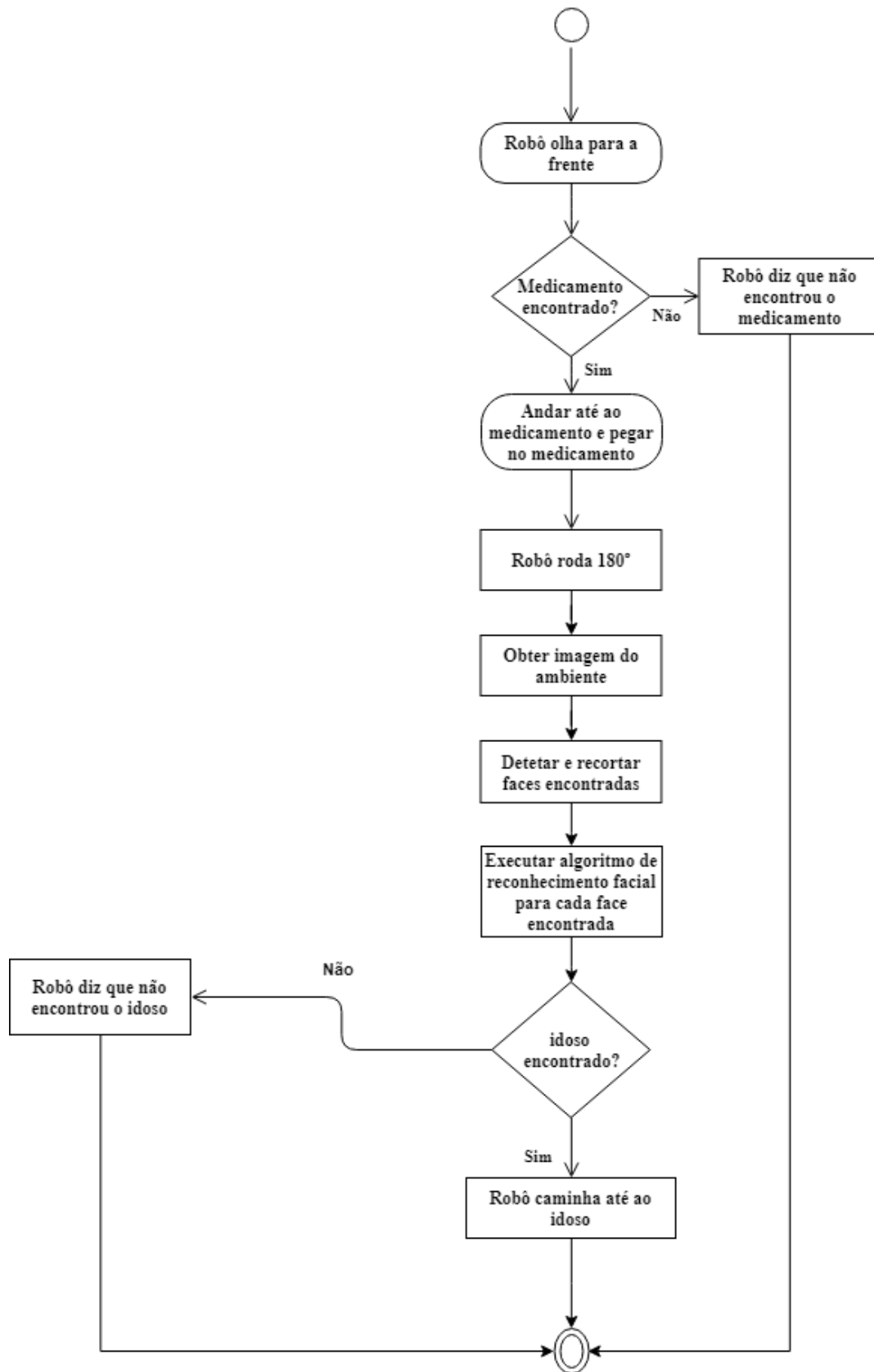


Figura 44 - Diagrama de atividades demonstrativo de todos os passos do sistema no momento da toma de medicamento.

5.2. Base de dados

Para o desenvolvimento da base de dados que suporta o sistema desenvolvido foi usada a tecnologia MySQL. A base de dados foi armazenada em *localhost* e consiste em três tabelas denominadas de Idoso, Medicamento e Horário, responsáveis por armazenar respetivamente dados dos utilizadores, medicamentos e todos os horários do regime de medicação de todos os idosos. A tabela 7 apresenta a listagem das tabelas criadas e o seu propósito.

Tabela 7 - Tabelas da base de dados.

Tabela	Propósito
Idoso	Responsável por guardar a informação dos idosos para os quais o sistema está preparado com nome e um identificador.
Medicamento	Responsável por guardar a informação dos medicamentos para os quais o sistema está preparado com nome e um identificador.
Horário	Responsável por guardar todo o regime de medicação de todos os idosos incluídos no sistema. Esta tabela guarda o horário da toma do medicamento, o identificador do idoso e o respetivo identificador do medicamento.

As imagens de treino são guardadas localmente numa diretoria dividida em duas subdiretorias, idosos e medicamentos. Dentro de cada subdiretoria foi criado um conjunto de diretorias cujo nome é o identificador de cada idoso/medicamento, contendo as imagens de treino de cada um dos idosos/medicamentos registados no sistema.

De forma a demonstrar melhor as relações entre as tabelas da base de dados na figura 45 é apresentado o diagrama da base de dados desenvolvida.

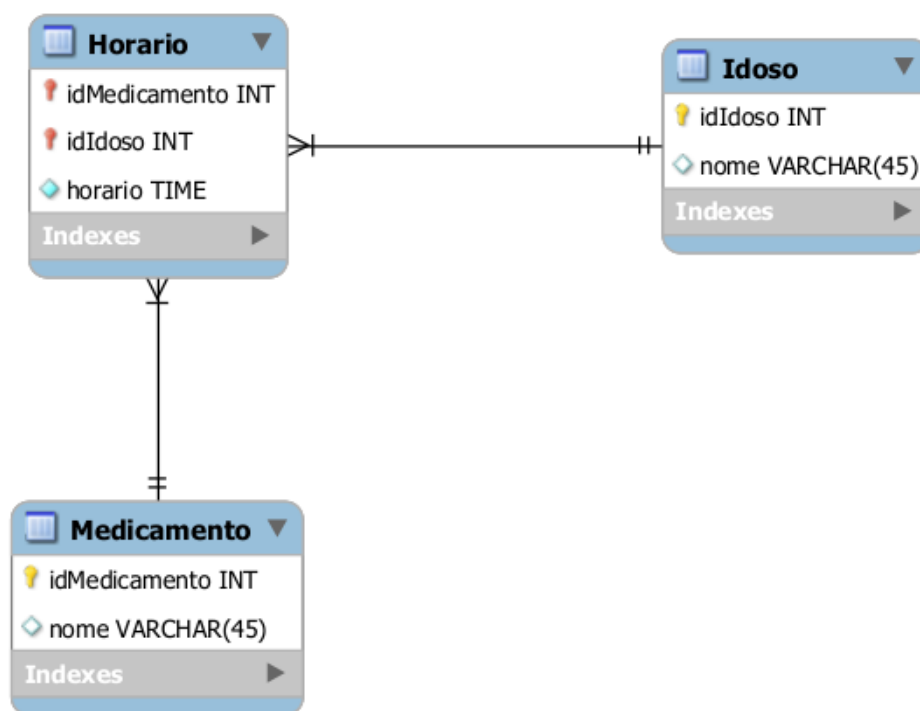


Figura 45 - Diagrama da base de dados desenvolvida.

5.3. Reconhecimento de medicamentos

Com o intuito do robô ajudar no cumprimento do regime de medicação é necessário que o robô reconheça objetos, neste caso em particular, é necessário que o robô consiga identificar na prateleira dos medicamentos aquele que se pretende entregar ao idoso.

Para o reconhecimento dos medicamentos foi utilizado o algoritmo SIFT para a extração de características e o algoritmo KNN para a classificação. Sucintamente, este processo consiste em comparar imagem do ambiente (obtida pelo robô) com as imagens de treino de modo a verificar se o medicamento pretendido se encontra na prateleira. Na figura 46 é apresentado um diagrama de atividades que demonstra as etapas efetuadas para o reconhecimento do medicamento. Este processo é disparado quando é atingido o horário de um certo idoso tomar um determinado medicamento. Após o processo ser iniciado, o robô direciona o campo de visão das câmaras para uma prateleira medicamentos (local onde se encontram disposto todas as embalagens dos medicamentos disponíveis) e obtém uma imagem através da câmara superior. Posteriormente são carregados os pontos de interesse (*feature points*) obtidos do treino com as imagens da respetiva embalagem do medicamento pretendido e são detetados os

pontos de interesse da imagem obtida pelo robô (imagem de teste). Finalmente, os pontos de interesse obtidos são usados no classificador KNN para identificar a localização do medicamento que se pretende entregar ao idoso.

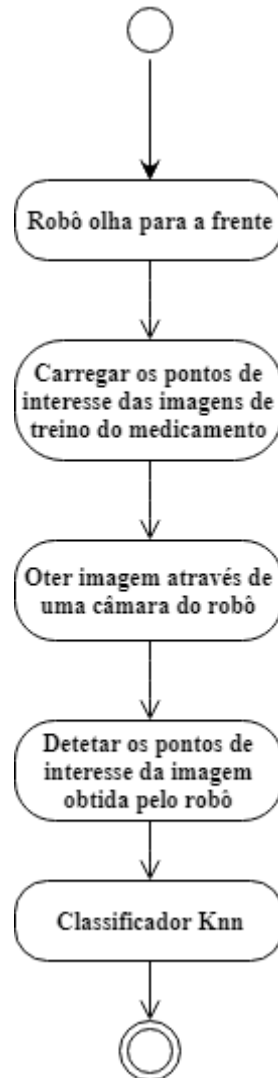


Figura 46 - Diagrama de atividades de todos os passos no reconhecimento de medicamentos.

5.3.1. Aquisição da imagem

A aquisição da imagem possui uma elevada importância no reconhecimento de medicamentos pois é através da obtenção da imagem que se tentará encontrar o medicamento pretendido. A imagem é adquirida através da câmara superior do robô. Para isto é utilizado o método *getImageRemote*, este método permite adquirir uma

imagem através de uma das câmaras do NAO com uma resolução e espaço de cor parametrizáveis como apresentado anteriormente.

Após a obtenção da imagem através do método referido, o output gerado é um vetor com informações da imagem, como altura, largura, matriz de caracteres ASCII que contém os dados da imagem adquirida, etc.

No entanto, é necessário guardar a imagem no computador para que possa ser usada posteriormente no algoritmo de reconhecimento de medicamentos. Para esse processo é utilizada a biblioteca PILLOW, que permite através dos dados do vetor guardar a imagem localmente no computador.

5.3.2. Extração das caraterísticas

A extração das caraterísticas da imagem é feita através do algoritmo SIFT que já possui uma implementação na biblioteca OpenCV. Este algoritmo permite obter diretamente os pontos chave (feature points) e calcula os descritores associados a cada ponto chave. Cada descritor é composto por um vetor de 128 posições que representam a orientação dos pontos chave. Na figura 47 é apresentado um exemplo de caraterísticas extraídas utilizando o algoritmo SIFT.



Figura 47 - Exemplo de extração de caraterísticas utilizando o SIFT.

5.3.3. Correspondência dos pontos chave

A comparação das imagens para posterior reconhecimento do medicamento faz uso dos descritores dos pontos chave, obtidos nas imagens de treino, e os descritores extraídos da imagem de teste. A correspondência é efetuada através do algoritmo KNN já implementado no OpenCV, considerando os 2 vizinhos mais próximos. Este valor permite que sejam retornados os dois pontos chave, melhores vizinhos do ponto chave da imagem de treino.

Após a obtenção dos melhores vizinhos, são considerados *good match* apenas os pontos chave cuja distância do melhor vizinho seja 75% menor do que a distância do segundo melhor vizinho. Esta verificação é efetuada de forma a eliminar possíveis falsos positivos. Na figura 48 é apresentado um as caraterísticas de uma imagem de teste e uma imagem de exemplo do resultado da correspondência entre treino.

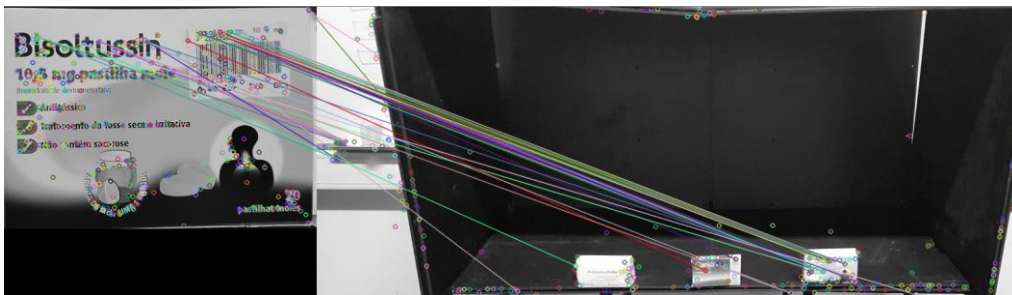


Figura 48 – Exemplo do resultado da comparação entre as caraterísticas de uma imagem de treino e uma imagem de teste.

A decisão da presença ou não na imagem do medicamento pretendido é tomada com base no número de *matches* encontrados entre cada um dos padrões de treino e a imagem de teste. O número de *matches* mínimo para se fazer a deteção da embalagem foi estabelecido empiricamente com o valor 30. Caso a caixa do medicamento esteja na imagem são desenhados contornos na zona onde a caixa se encontra como se mostra na figura 49.

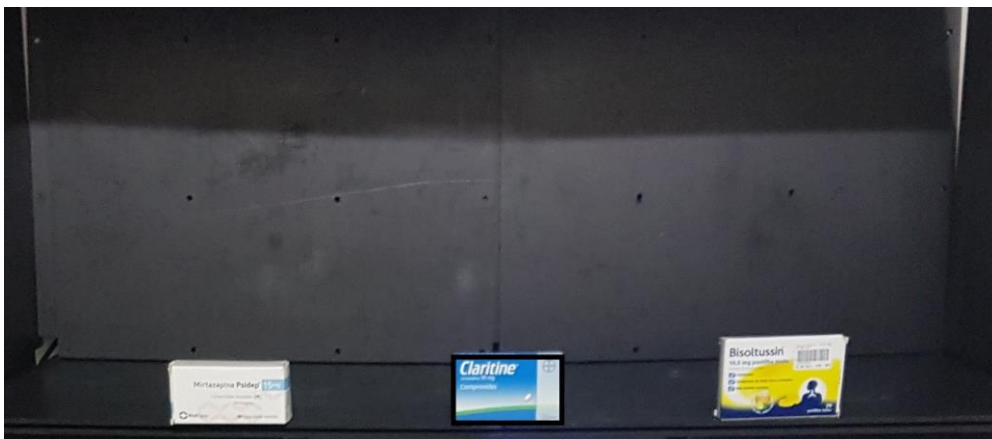


Figura 49 - Exemplo do resultado do algoritmo de reconhecimento de objetos no caso do medicamento ser reconhecido.

Após reconhecer o medicamento pretendido e detetar a sua posição na prateleira, o robô “empurra” a embalagem para dentro de um cesto que transporta preso no pescoço (figura 50) e no qual já se encontra uma garrafa de água para entregar ao idoso.

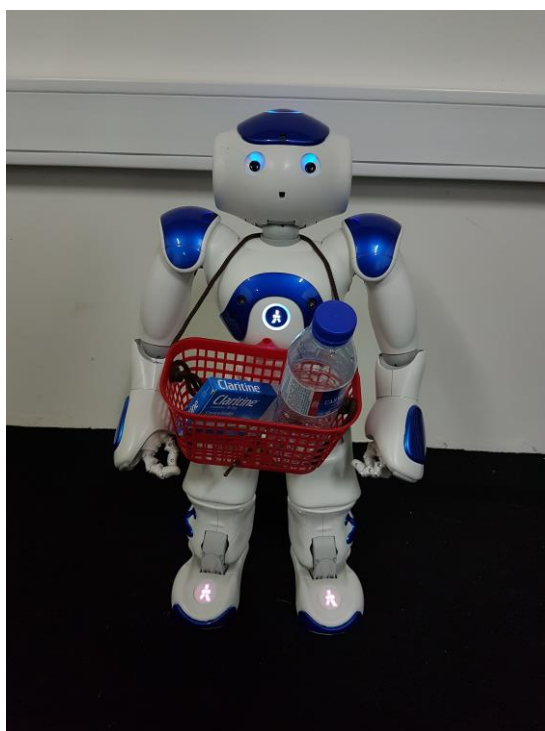


Figura 50 - Imagem do robô NAO a carregar o medicamento e a água com o cesto preso no pescoço.

5.4. Reconhecimento da pessoa

Depois de já ter selecionado o medicamento é necessário que o robô reconheça o idoso que vai tomar o medicamento e se dirija a ele para lhe entregar a embalagem e a água.

O reconhecimento do idoso é feito com os algoritmos LBP e LBPH. Sucintamente, este processo consiste na identificação facial (reconhecimento). A identificação facial consiste na correspondência de um para muitos, em que as imagens de todas as faces presentes na imagem obtida pelo robô são comparadas com as faces de todos os idosos de forma a determinar a identidade das faces adquiridas (ver ponto 2.3). O objetivo deste processo é verificar se alguma das faces presentes na imagem obtida pelo robô corresponde ao idoso que se pretende reconhecer. O processo divide-se em duas fases: treino e teste.

Na fase de treino o algoritmo LBPH é treinado com imagens dos idosos que podem estar na habitação e se pretendem reconhecer. Neste processo são criados os histogramas de todas as imagens de treino através dos códigos LBP.

Na fase de teste, reconhecimento de um idoso, o robô direciona o campo de visão das câmaras para o local onde é previsto estarem os idosos e obtém uma imagem através da câmara superior.

Na figura 51 é apresentado um diagrama de atividades que demonstra as etapas efetuadas para o reconhecimento do idoso.

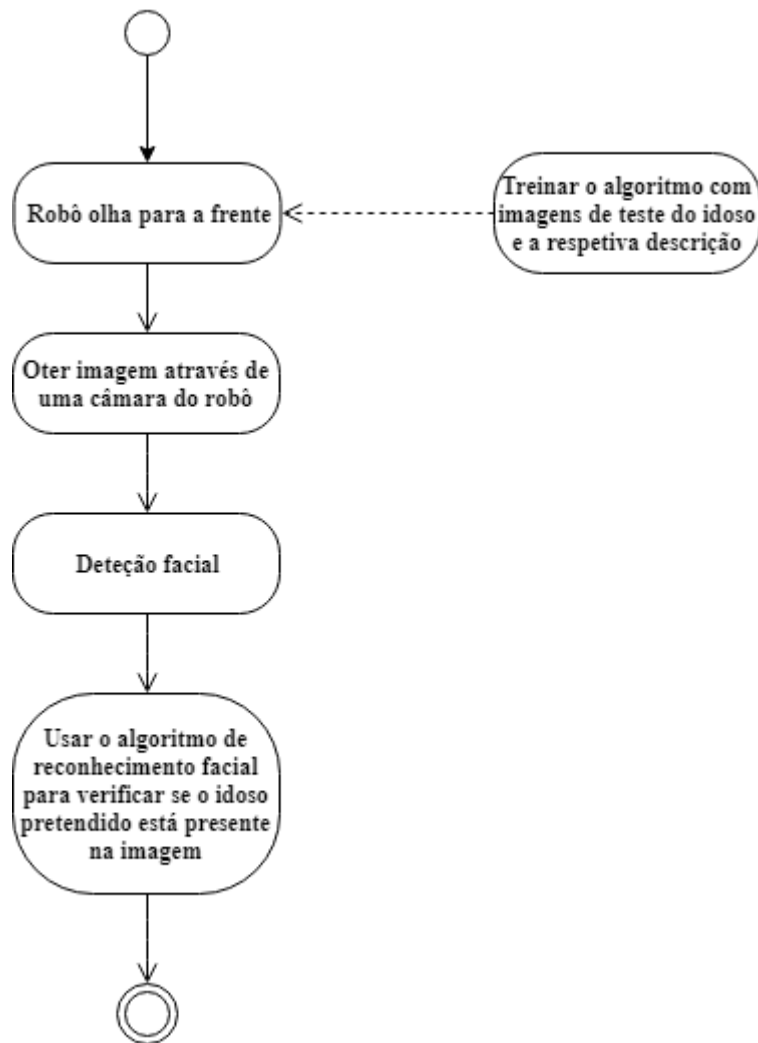


Figura 51 - Diagrama de atividades de todos os passos no reconhecimento de um idoso.

5.4.1. Aquisição da imagem

Esta etapa é efetuada exatamente da mesma forma que a aquisição da imagem no reconhecimento de objetos descrita no ponto 5.3.1, apenas diferindo no campo de visão para onde a câmara se direciona, sendo neste caso o local onde se deverão encontrar os idosos.

5.4.2. Deteção facial

A etapa de reconhecimento de idosos está dependente da deteção de uma face tanto no treino no algoritmo como no reconhecimento do idoso, por isso, é necessário detetar as faces e recortar a região da imagem onde a face foi detetada.

O OpenCV fornece dois classificadores previamente treinados e prontos a utilizar. Estes dois classificadores são o *Haar Classifier* e *LBP Classifier*.

De modo a detetar faces, nesta tese foi aplicado o classificador LBP. Foi escolhido este classificador por ser computacionalmente mais simples e bastante robusto a variações de iluminação nas imagens.

5.4.3. Treino

Com o objetivo de treinar o algoritmo LBPH são usadas imagens de cada utilizador obtidas através da câmara do robô. Estas imagens, juntamente com o identificador do idoso são utilizadas pelo algoritmo LBPH para criar histogramas das ocorrências dos códigos LBP nas imagens de treino que possuem informações sobre os padrões da face de cada indivíduo.

5.4.4. Reconhecimento

O reconhecimento é efetuado através da diferença entre os histogramas das imagens de teste (faces detetadas) com os histogramas das imagens de treino. Esse cálculo da diferença atribui um valor de confiança à verificação da face do idoso. Quanto maior a confiança mais baixa será o valor da mesma, o que significa que existem menos diferenças entre os dois histogramas.

Dessa forma, a face de teste é classificada na categoria com a qual se obtém a menor diferença entre os respetivos histogramas (classificador de distância mínima).

De forma a permitir ao algoritmo LBPH verificar uma face automaticamente, foi atribuído um valor de *threshold* que permite ter um grau de confiança no reconhecimento. O valor (*threshold* igual 50) foi definido empiricamente, de forma a aumentar a taxa de reconhecimento. Este valor de *threshold* foi necessário pois apesar de se considerar a face com maior semelhança às imagens de treino, esta semelhança pode não ser satisfatória.

6. Resultados e testes

De forma a concluir com sucesso o objetivo, é necessário que as várias etapas do processamento sejam efetuadas com sucesso.

Após a fase de implementação do sistema, procedeu-se à elaboração de um conjunto de testes às várias etapas de forma a obter métricas qualitativas do desempenho do sistema. Foram efetuados testes aos algoritmos usados: deteção facial, reconhecimento facial e reconhecimento de objetos.

Além dos testes aos algoritmos, procedeu-se também a um teste em cenário real com o robô e dois sujeitos.

6.1. Métricas

Os algoritmos de visão por computador foram avaliados usando as seguintes métricas de desempenho (Bradley, 1997):

Verdadeiro positivo (VP) – O padrão é verdadeiro e o algoritmo classifica-o como verdadeiro.

Verdadeiro Negativo (VN) - O padrão é falso e o algoritmo classifica-o como falso.

Falso Positivo (FP) – O padrão é falso e o algoritmo classifica-o como verdadeiro.

Falso Negativo (FN) - O padrão é verdadeiro e o algoritmo classifica-o como falso.

Para simplificar o entendimento das 4 classificações possíveis, na tabela 8 é apresentada a matriz de confusão que permite a visualização das possíveis situações apresentadas.

Tabela 8 - Matriz de confusão. Adaptado de (Bradley, 1997).

		Classificação real do padrão	
		Positivo	Negativo
Classificação do padrão atribuída pelo algoritmo	Positivo	VP	FP
	Negativo	FN	VN

Baseado na matriz apresentada, são de seguida apresentadas as métricas utilizadas para a medição do desempenho nos testes realizados aos algoritmos de reconhecimento de medicamentos, deteção facial e reconhecimento facial. Estas métricas são:

Taxa de aceitação falsa (False Acceptance Rate) ou taxa de falsos positivos: Representa a probabilidade de um padrão falso ser classificado como verdadeiro. Esta métrica resulta da divisão do número de falsos positivos pelo número total de padrões negativos, ou seja, a soma dos falsos positivos com os verdadeiros negativos.

$$\frac{FP}{FP + VN}$$

Taxa de falsa rejeição (False Rejection Rate) ou taxa de falsos negativos: Representa a probabilidade de um padrão verdadeiro ser classificado como falso. Esta métrica resulta da divisão do número de falsos negativos pelo número total de padrões positivos, ou seja, a soma dos verdadeiros positivos com os falsos negativos.

$$\frac{FN}{VP + FN}$$

Taxa de verdadeiros positivos (True Positive Rate) ou sensibilidade: Representa a probabilidade de um padrão verdadeiro ser classificado corretamente. Esta métrica

resulta da divisão do número de verdadeiros positivos pelo número total de padrões positivos, ou seja, a soma dos verdadeiros positivos com os falsos negativos.

$$\frac{VP}{VP + FN}$$

Taxa de verdadeiros negativos (True Negative Rate) ou especificidade: Representa a probabilidade de um padrão falso ser classificado corretamente. Esta métrica resulta da divisão do número de verdadeiros negativos pelo número total de padrões negativos, ou seja, a soma dos verdadeiros negativos com os falsos positivos.

$$\frac{VN}{VN + FP}$$

Precisão (Precision): Representa a taxa com que todos os padrões classificados como positivo são na realidade positivos, ou seja, não se consideram os resultados negativos. Esta métrica resulta da divisão de verdadeiros positivos pelo número total de padrões classificados como positivos, ou seja, a soma dos verdadeiros positivos com os falsos positivos.

$$\frac{VP}{VP + FP}$$

Exatidão (Accuracy): Representa proporção entre as classificações reais e as classificações obtidas pelo algoritmo, ou seja, consiste na proporção de predições corretas. Esta métrica resulta da divisão da soma de verdadeiros positivos com os verdadeiros negativos pelo número de classificações obtidas pelo algoritmo, ou seja, a soma dos falsos positivos e negativos com os verdadeiros positivos e negativos.

$$\frac{VP + VN}{VP + VN + FP + FN}$$

6.2. Testes de avaliação dos algoritmos de visão por computador

Durante a avaliação aos algoritmos de visão por computador, foram testados os três algoritmos utilizados no sistema desenvolvido. De forma a facilitar o objetivo de contagem dos resultados alcançados, foi desenvolvida uma aplicação em C# (figura 52) que permite selecionar a pasta que possui as imagens que serão utilizadas no teste. Após a seleção da pasta, as imagens da base de dados são carregadas e o algoritmo é executado. Posteriormente, é apresentado o resultado do algoritmo que é avaliado e categorizado como falso negativo, falso positivo, verdadeiro negativo ou verdadeiro positivo, através de um dos botões do canto inferior direito que permite a contabilização dos resultados obtidos.



Figura 52 - Aplicação desenvolvida para facilitar na contagem dos dados obtidos nos testes efetuados.

6.2.1. Reconhecimento de objetos

O primeiro conjunto de testes possui como objetivo avaliar o sucesso da integração do SIFT no reconhecimento de medicamentos.

Para este teste foram utilizadas 50 imagens, entre elas caixas de medicamentos, cereais e latas de refrigerantes recolhidas na web. Foram selecionadas 40 imagens de treino (figura 53) para os vários produtos e 10 imagens de teste contendo várias

embalagens (figura 54). Cada imagem de treino foi comparada com todas as imagens de treino de modo a obter valores das métricas que classificam o desempenho do algoritmo.



Figura 53 - Exemplos de imagens de treino utilizadas no decorrer deste teste.



Figura 54 - Exemplos de imagens de teste utilizadas no decorrer deste teste.

Na tabela 9 estão apresentados todos os resultados obtidos no teste.

Tabela 9 - Matriz de confusão com os resultados obtidos no teste de reconhecimento de objetos efetuado.

		Classificação real	
		Encontrado	Não encontrado
Classificação atribuída pelo algoritmo	Encontrado	193 (87,7%)	31 (17,2%)
	Não encontrado	27 (12,3%)	149 (82,8%)

Após a obtenção destes resultados foram calculados e analisados os valores das métricas apresentadas anteriormente.

Taxa de aceitação falsa: Esta métrica consiste na probabilidade de uma imagem que não contém um objeto seja classificada como imagem que contém esse objeto. Foi obtido o valor aproximado de 0,172 nesta métrica, o que representa 17,2%.

Taxa de falsa rejeição: Esta métrica consiste na probabilidade de uma imagem que contém um determinado objeto ser classificada como imagem que não contém esse objeto. Foi obtido o valor aproximado de 0,123 nesta métrica, o que representa 12,3%.

Sensibilidade: Esta métrica consiste na probabilidade de uma imagem que contém um objeto seja classificado corretamente. Foi obtido o valor aproximado de 0,877 nesta métrica, o que representa 87,7%.

Especificidade: Esta métrica consiste na probabilidade de uma imagem que não contém um determinado objeto seja classificada corretamente. Foi obtido o valor aproximado de 0,828 nesta métrica, o que representa 82,8%.

Precisão: Esta métrica consiste na probabilidade de uma imagem que contém um certo objeto seja classificada corretamente, sem considerar as imagens que são classificadas como não contendo o objeto. Foi obtido o valor aproximado de 0,862 nesta métrica, o que representa 86,2%.

Exatidão: Esta métrica consiste proporção entre as classificações reais e as classificações obtidas pelo algoritmo, ou seja, consiste na proporção de predições corretas. Foi obtido o valor aproximado de 0,855 nesta métrica, o que representa 85,5%.

Para melhor visualização dos resultados obtidos, na tabela 10 são apresentados os valores de todas as métricas calculadas.

Tabela 10 - Avaliação de desempenho do algoritmo de reconhecimento de objetos.

Métricas	Valores obtidos (%)
Taxa de aceitação falsa	17,2
Taxa de falsa rejeição	12,3
Sensibilidade	87,7
Especificidade	82,8
Precisão	86,2
Exatidão	85,5

Os resultados obtidos são bastante positivos, no entanto, dois dos valores deveriam ser diferentes.

Uma vez que o algoritmo do reconhecimento de objetos será utilizado para o reconhecimento de medicamentos, o valor da taxa de aceitação falsa deveria ser menor. Num caso real, caso ocorra uma aceitação falsa, o robô poderá entregar a um idoso o medicamento errado e isso poderá trazer complicações para a sua saúde.

Por outro lado, o valor obtido através da métrica de precisão deveria ser mais elevado, o que significaria que existiriam um menor número de ocorrências de “falsos positivos”. Uma vez mais, os falsos positivos, significariam que o robô poderia entregar a um idoso o medicamento errado.

6.2.2. Detecção facial

Os testes relativos à detecção facial possuem como objetivo avaliar o sucesso do classificador LBP utilizado na detecção da face.

Visto que a implementação deste trabalho é focada nas pessoas idosas, os testes ideais para esta etapa seriam a utilização de imagens de idosos obtidas pelo robô no cenário de teste utilizado. Como tal não foi possível, foi utilizada uma base de dados do Park Aging Mind Lab da Universidade do Texas em Dallas. Esta base de dados contém 1142 imagens de faces de 576 pessoas de diferentes idades. A base de dados contém imagens com expressões faciais neutras, expressões de felicidade ou imagens de perfil. Embora esta base de dados possua imagens de pessoas dos 18 aos 93 anos de idade, apenas foram utilizadas imagens de pessoas com idades compreendidas entre os 50 e os 93 anos de idade.

Deste grupo restrito de imagens selecionadas, encontram-se 281 pessoas, sendo 76 do sexo masculino e 205 do sexo feminino, somando um total de 374 imagens. Relativamente ao grupo de imagens utilizadas, 93 imagens possuem faces com expressões faciais de felicidade, 281 imagens possuem faces com expressões faciais

neutras (Minear & Park, 2004). As imagens da face de perfil não foram utilizadas porque o classificador LBP utilizado apenas permite detetar faces frontais.

Embora este teste utilize imagens de idosos, o fundo da imagem é bastante simples. Com o objetivo de também avaliar o algoritmo em imagens com fundos mais complexos, foi acrescentado ao *dataset* as imagens de uma outra base de dados disponibilizada publicamente pelo Markus Weber do Instituto de tecnologia da Califórnia (“Computational Vision: Archive,” n.d.). Esta base de dados possui 450 imagens de faces de 27 pessoas diferentes com expressões, condições de iluminação e fundos de imagem diferentes. Foi escolhida esta base de dados pois possui imagens com situações de maior complexidade que são esperadas quando o sistema é usado em condições reais.

Com objetivo de avaliar todas as métricas anteriormente apresentadas, foram adicionadas ao *dataset* 50 imagens de animais aleatórias da base de uma base de dados disponibilizada pela Universidade de Oxford (O. M. Parkhi, A. Vedaldi, A. Zisserman, 2012), permitindo assim a obtenção de “verdadeiros negativos”.

Em suma, no total foram utilizadas nos testes 824 imagens contendo faces e 50 imagens de animais que não contem faces.

Na tabela 11 estão apresentados todos os resultados obtidos no teste.

Tabela 11 - Matriz de confusão com os resultados obtidos no teste de deteção facial efetuado.

		Classificação real	
		Face	Não tem face
Classificação atribuída pelo algoritmo	Face	797 (96,7%)	2 (4%)
	Não tem face	27 (3,3%)	48 (96,0%)

Após a obtenção destes resultados foram calculados e analisados os valores das métricas apresentadas anteriormente.

Taxa de aceitação falsa: Esta métrica consiste na probabilidade de uma imagem que não contém uma face seja classificada como imagem que contém face. Foi obtido o valor aproximado de 0,04 nesta métrica, o que representa 4,0%.

Taxa de falsa rejeição: Esta métrica consiste na probabilidade de uma imagem que contém uma face seja classificada como imagem que não contém face. Foi obtido o valor aproximado de 0,033 nesta métrica, o que representa 3,3%.

Sensibilidade: Esta métrica consiste na probabilidade de uma imagem que contém uma face seja classificada corretamente. Foi obtido o valor aproximado de 0,967 nesta métrica, o que representa 96,7%.

Especificidade: Esta métrica consiste na probabilidade de uma imagem que não contém uma face seja classificada corretamente. Foi obtido o valor aproximado de 0,96 nesta métrica, o que representa 96,0%.

Precisão: Esta métrica consiste na probabilidade de uma imagem que contém uma face seja classificada corretamente, sem considerar as imagens que são classificadas como não contendo faces. Foi obtido o valor aproximado de 0,997 nesta métrica, o que representa 99,7%.

Exatidão: Esta métrica consiste proporção entre as classificações reais e as classificações obtidas pelo algoritmo, ou seja, consiste na proporção de predições corretas. Foi obtido o valor aproximado de 0,966 nesta métrica, o que representa 96,6%.

Para melhor visualização dos resultados obtidos, na tabela 12 são apresentados os valores de todas as métricas calculadas.

Tabela 12 – Avaliação de desempenho do algoritmo de detecção facial.

Métricas	Valores obtidos (%)
Taxa de aceitação falsa	4,0
Taxa de falsa rejeição	3,3
Sensibilidade	96,7
Especificidade	96,0
Precisão	99,7
Exatidão	96,6

Os resultados obtidos são bastante positivos e não trazem grandes problemas para o sistema final desenvolvido pois, após a etapa da detecção facial ainda é efetuada a etapa de reconhecimento facial.

6.2.3. Reconhecimento facial

Na última etapa dos testes, foi avaliado o algoritmo de reconhecimento facial integrado no sistema desenvolvido.

Para os testes executados no algoritmo de reconhecimento facial utilizou-se uma base de dados disponibilizada publicamente pela Universidade Nacional de Cheng Kung (NCKU) (“Databases for Face Detection,” n.d.). Esta base de dados de imagens, contém 6660 imagens de 90 pessoas diferentes, ou seja, 74 imagens por pessoa. Cada pessoa possui 37 imagens de com intervalos de 5° desde a visão de perfil do lado direito (90°) até à visão de perfil do lado direito (-90°) obtidas através de rotação panorâmica. As restantes 37 imagens de cada pessoa foram obtidas através do uso de um software, rodando as imagens horizontalmente. Para os testes efetuados foram consideradas as imagens obtidas através da rotação panorâmica.

Uma vez que o algoritmo de detecção facial não deteta faces com visualização de perfil, foram usadas apenas as imagens entre o ângulo -35° e o ângulo 35° inclusive, o que gerou um total de 15 imagens por pessoa.

Deste conjunto de imagens, foram utilizadas apenas 50 pessoas. Sendo utilizadas 12 imagens para treinar o algoritmo e 3 imagens para testar o algoritmo.

Antes de proceder a este teste, o valor do *threshold* de confiança do algoritmo de detecção facial teve de ser ajustado devido à diferença das condições de iluminação e qualidade entre as imagens das bases de dados usadas e o ambiente laboratorial. O valor foi estabelecido em 20.

Em suma, foram efetuados por cada pessoa da base de dados 3 testes de reconhecimento facial, ou seja, 150 testes no total.

Este teste consiste em reconhecer as pessoas presentes nas imagens de teste. Visto que todas as imagens de teste pertencem a uma das pessoas contida no *dataset*, os resultados possíveis neste teste são:

- Reconhecimento correto (VP) – Neste resultado, o algoritmo reconhece a pessoa de forma correta.
- Reconhecimento errado (FP) – Neste resultado, o algoritmo reconhece uma pessoa que não corresponde à pessoa presente na imagem de teste.
- Não reconhecimento (FN) – Neste resultado, o algoritmo não reconhece nenhuma pessoa como sendo a pessoa na imagem de teste.

Na tabela 13 estão apresentados todos os resultados obtidos no teste.

Tabela 13 – Resultados obtidos no teste de reconhecimento facial.

Reconhecimento correto (VP)	Reconhecimento errado (FP)	Não reconhecimento (FN)
123 (82%)	23 (15,3%)	4 (2,7%)

Após a obtenção destes resultados foram calculados e analisados os valores de algumas das métricas apresentadas anteriormente.

Sensibilidade: Esta métrica consiste na probabilidade de uma pessoa ser reconhecida corretamente pelo sistema. Foi obtido o valor aproximado de 0,966 nesta métrica, o que representa 96,6%.

Precisão: Esta métrica consiste na probabilidade de uma pessoa ser corretamente reconhecida, sem considerar as imagens que não são reconhecidas. Foi obtido o valor aproximado de 0,842 nesta métrica, o que representa 84,2%.

Para melhor visualização dos resultados obtidos, na tabela 14 são apresentados os valores de todas as métricas calculadas.

Tabela 14 - Avaliação de desempenho do algoritmo de detecção facial.

Métricas	Valores obtidos (%)
Sensibilidade	96,6
Precisão	84,2

Os resultados obtidos, no geral, são positivos. No entanto, o reconhecimento facial é um processo importante pois, o reconhecimento errôneo de um idoso origina dois problemas, a administração errada de medicamentos a um idoso e a privação da toma da medicação a outro idoso.

6.3. Teste em cenário real

Com o objetivo de testar e demonstrar a funcionalidade do sistema desenvolvido, foi executado um teste em cenário real com dois utilizadores com idades compreendidas entre os 50 e os 60 anos.

Este teste foi efetuado num cenário preparado para duas pessoas que se encontram sentados numa cadeira e três medicamentos que se encontram numa prateleira preta em frente ao robô (figura 55).

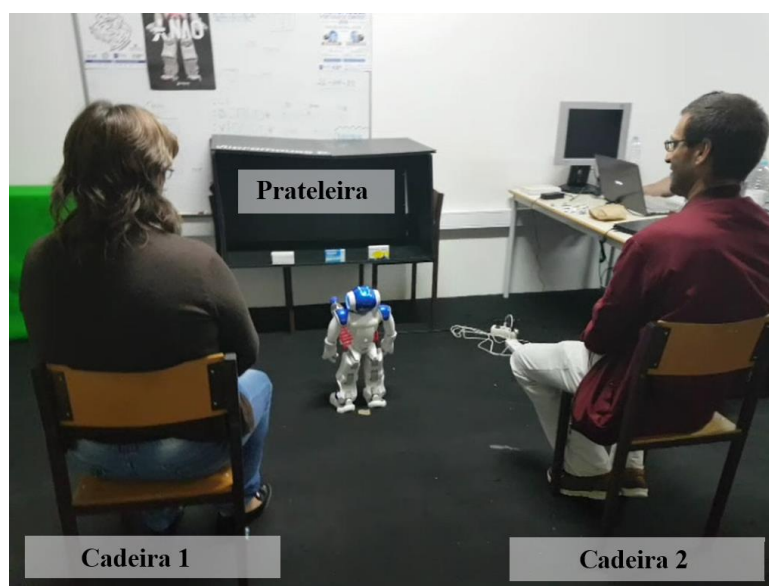


Figura 55 - Imagem do cenário de teste em ambiente real.

Nos testes executados foram utilizados os medicamentos *Bisoltussin*, *Mirtazapina Psidep* e *Claritine* (ver imagem 56). No entanto podem ser utilizados outros medicamentos, sendo necessário inserir os seus dados na base de dados e inserir imagens de treino do novo medicamento.



(a)



(b)



(c)

Figura 56 - Medicamentos utilizados no teste em cenário real. (a) Mirtazapina Psidep. (b) Claritine. (c) Bisoltussin.

Numa fase inicial, foram obtidas 12 fotos de cada um dos utilizadores através da câmara do robô NAO e guardadas na base de dados para o treino do modelo do reconhecimento facial.

Posteriormente foi agendada a toma de um medicamento para o utilizador com maior idade (denominado de “Dona Maria”) e adicionado à base de dados (tabela 15).

Tabela 15 - Tabela demonstrativa do procedimento experimental.

Idoso	Medicamento	Horário
Dona Maria ²⁵	Claritine	14:05

Atingido o horário da toma de medicação previamente agendada, o robô deu início à atividade reconhecendo o medicamento e colocando-o no cesto. Posteriormente o robô direcionou o seu campo de visão para a 1ª cadeira, verificando se se tratava da pessoa correta. Uma vez que não era a pessoa correta, o robô direcionou o seu campo de visão para a 2ª cadeira e verificou novamente. Neste caso reconheceu a pessoa que deve tomar o medicamento e como tal dirigiu-se até ela para lhe entregar o medicamento e a água. Ao longo deste processo experimental o robô comunica algumas informações ao idoso como representado na figura 57.

²⁵ Nome fictício

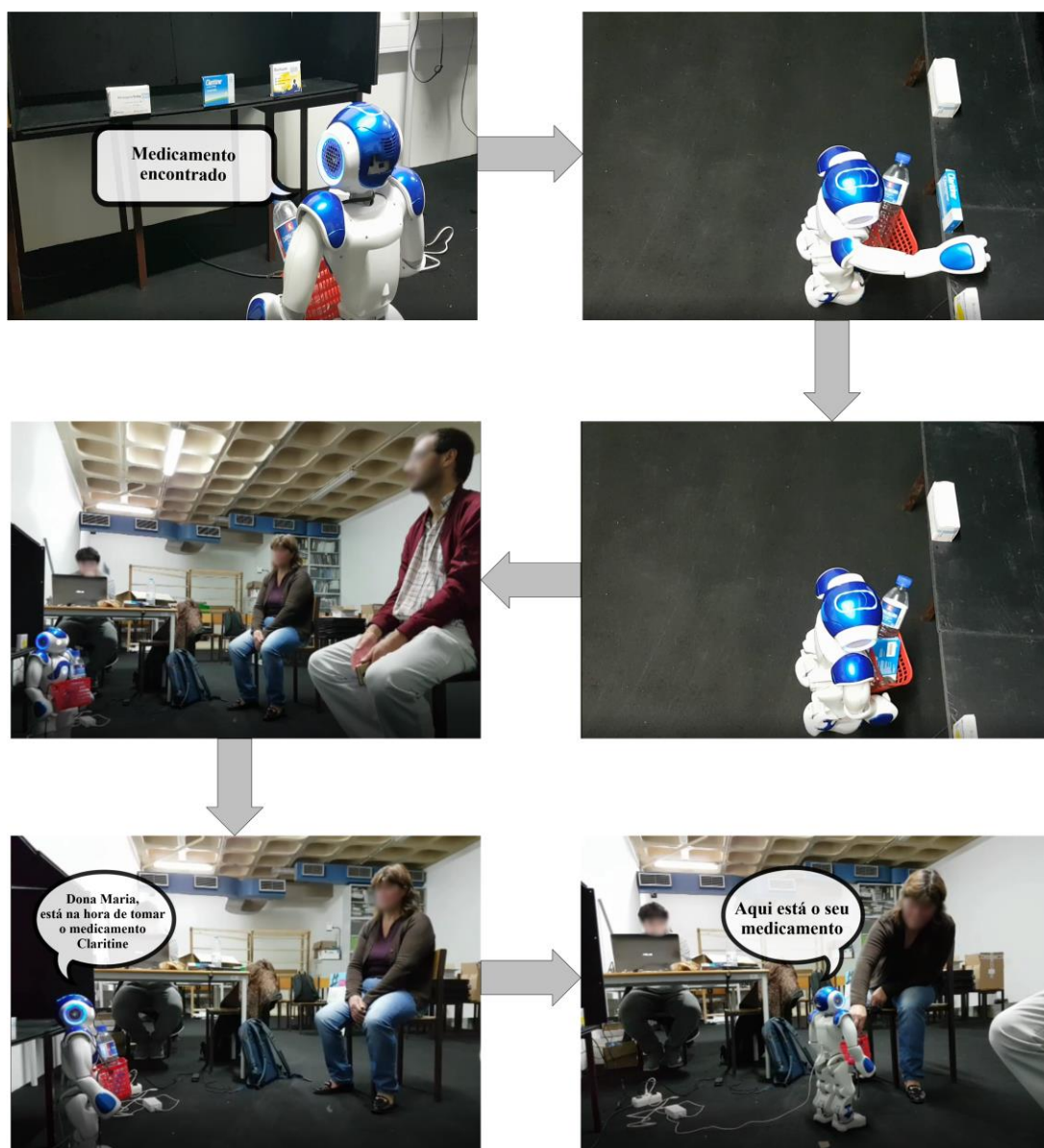


Figura 57 - Representação gráfica das várias etapas do teste em cenário real.

No teste efetuado, os resultados foram os esperados e o robô identificou o medicamento e a pessoa correta e entregou com sucesso o medicamento.

7. Conclusões

Atualmente, a área da robótica encontra-se em constante desenvolvimento. Esta evolução tem permitido o aumento da possibilidade do uso de sistemas robóticos para o apoio a idosos. Por outro lado, o atual envelhecimento da população mundial e o implícito aumento do regime de medicação pode conduzir a um problema caso o regime não seja rigorosamente cumprido. Levando em consideração estes aspetos, os sistemas robóticos podem fornecer apoio a idosos na toma de medicamentos, bem como outras tarefas diárias.

Embora já existam algumas soluções para o apoio à toma de medicação, nenhuma solução é capaz de solucionar totalmente o problema.

Com o intuito de combater as limitações que as soluções já existentes possuem, foi projetada uma arquitetura de um sistema que permita automaticamente, no horário correto, entregar o medicamento ao idoso para que este cumpra o seu regime de medicação. Para cumprir tal objetivo, no sistema desenvolvido foi utilizado o robô NAO, que através do seu sistema de locomoção e visão permite o movimento e o reconhecimento de objetos e facial.

Durante a implementação foram utilizados vários algoritmos para as várias etapas do processo. Para o reconhecimento de medicamentos foi utilizado o algoritmo SIFT, por outro lado para a deteção facial e reconhecimento dos idosos foram utilizados o LBP e o LPBH respetivamente.

Após o desenvolvimento do sistema foram efetuados um conjunto de testes aos três algoritmos de forma a obter métricas qualitativas do desempenho do sistema. Estes testes efetuados revelaram resultados bastante positivos, no entanto, o reconhecimento erróneo de uma pessoa levará o robô a entregar um certo medicamento a uma pessoa errada. Este erro origina dois problemas, a administração errada de medicamentos a um idoso e a falta da toma da medicação a outro idoso. Por outro lado, de igual forma, o reconhecimento de medicamentos de forma errónea levará à administração de um medicamento errado a um idoso.

Em termos globais os resultados obtidos na avaliação dos algoritmos podem-se considerar satisfatórios para o fim pretendido.

7.1. Limitações e trabalho futuro

A grande limitação deste trabalho consiste no reduzido número de testes em cenário real. Desta forma, as conclusões deste trabalho baseiam-se maioritariamente na avaliação dos algoritmos individualmente.

Apesar do sucesso na conclusão de todos os objetivos previamente definidos, existem algumas melhorias que seriam necessárias implementar para que o uso deste tipo de sistemas possa ser viável. Essas melhorias são:

- Desenvolver a componente da cinemática
- Após a conclusão da toma de medicamento, o robô voltar a meter o medicamento na prateleira
- Implementar e testar outros algoritmos de forma a melhorar os resultados obtidos
- Desenvolver testes com idosos em ambiente real como por exemplo em lares de idosos
- Desenvolver uma aplicação que permita gerir os regimes de medicação dos idosos, os próprios idosos e os medicamentos.
- Implementar mecanismos que permitam verificar se o idoso realmente procede à toma do medicamento

Referências bibliográficas

- About - OpenCV library. (n.d.). Retrieved June 22, 2018, from <https://opencv.org/about.html>
- About us. (n.d.). Retrieved June 8, 2010, from <https://www.softbankrobotics.com/emea/en/about-us>
- Ahonen, T., Hadid, A., & Pietikainen, M. (2006). Face Description with Local Binary Patterns: Application to Face Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(12), 2037–2041. <https://doi.org/10.1109/TPAMI.2006.244>
- Alaliyat, S. (2008). *Video -based Fall Detection in Elderly's Houses*. Gjøvik University College. Retrieved from https://www.researchgate.net/publication/267953942_Video_-based_Fall_Detection_in_Elderly%27s_Houses
- Bay, H., Ess, A., Tuytelaars, T., & Van Gool, L. (2008). Speeded-Up Robust Features (SURF). *Computer Vision and Image Understanding*, 110(3), 346–359. <https://doi.org/10.1016/j.cviu.2007.09.014>
- Belongie, S., Malik, J., & Puzicha, J. (2000). Shape context: A new descriptor for shape matching and object recognition. *Advances in Neural Information Processing Systems*, 546, 831–837. <https://doi.org/10.1.1.27.8567>
- Bradley, A. P. (1997). The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognition*, 30(7), 1145–1159. [https://doi.org/10.1016/S0031-3203\(96\)00142-2](https://doi.org/10.1016/S0031-3203(96)00142-2)
- Broadbent, E., Stafford, R., & MacDonald, B. (2009). Acceptance of healthcare robots for the older population: Review and future directions. *International Journal of Social Robotics*, 1(4), 319–330. <https://doi.org/10.1007/s12369-009-0030-6>
- Broekens, J., Heerink, M., & Rosendal, H. (2009). Assistive social robots in elderly care: a review. *Gerontechnology*, 8(2). <https://doi.org/10.4017/gt.2009.08.02.002.00>
- Burel, G., & Carel, D. (1994). Detection and localization of faces on digital images. *Pattern Recognition Letters*, 15(10), 963–967. [https://doi.org/10.1016/0167-8655\(94\)90027-2](https://doi.org/10.1016/0167-8655(94)90027-2)
- Butter, M., Rensma, A., van Boxxel, J., Kalisingh, S., Schoone, M., Leis, M., ... Korhonen, I. (2008). Robotics in Helthcare, Final Report. *Robotics in Helthcare*, 179.
- Buytaert, D. (2010). The history of MySQL AB. Retrieved July 12, 2018, from <https://dri.es/the-history-of-mysql-ab>
- Cabral, M., Ferreira, P., Silva, P., Jerónimo, P., & Marques, T. (2013). *Processos de Envelhecimento em Portugal - Usos do tempo, redes sociais e condições de vida*.
- Campbell, N. L., Boustani, M. A., Skopelja, E. N., Gao, S., Unverzagt, F. W., & Murray, M. D. (2012). Medication adherence in older adults with cognitive impairment: A systematic evidence-based review. *American Journal of Geriatric Pharmacotherapy*, 10(3), 165–177. <https://doi.org/10.1016/j.amjopharm.2012.04.004>
- Care-O-bot 4 - Technical Data. (n.d.). Retrieved May 20, 2018, from <https://www.care-o->

- bot.de/en/care-o-bot-4/technical-data.html
- Choregraphe overview — NAO Software 1.14.5 documentation. (n.d.). Retrieved June 16, 2018, from http://doc.aldebaran.com/1-14/software/choregraphe/choregraphe_overview.html
- Clarke, R. (1994). Asimov ' s Laws of Robotics : Implications for Information Techno, 57–66. <https://doi.org/10.1109/2.247652>
- Clasificacion de robots | Wiki de Robótica. (n.d.). Retrieved April 3, 2018, from <http://wiki.robotica.webs.upv.es/wiki-de-robotica/introduccion/clasificacion-de-robots/>
- Computational Vision: Archive. (n.d.). Retrieved August 15, 2018, from <http://www.vision.caltech.edu/html-files/archive.html>
- Computer Programming for Everybody | Python.org. (n.d.). Retrieved June 20, 2018, from <https://www.python.org/doc/essays/cp4e/>
- Conheça Pillo, o robô que promete cuidar da sua saúde. (n.d.). Retrieved May 26, 2018, from <https://www.freetheessence.com.br/inovacao/saude/pillo-robo-saude/>
- Connectivity — Aldebaran 2.1.4.13 documentation. (n.d.). Retrieved June 10, 2018, from http://doc.aldebaran.com/2-1/family/robots/connectivity_ao.html
- Cover, T., & Hart, P. (1967). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1), 21–27. <https://doi.org/10.1109/TIT.1967.1053964>
- Databases for Face Detection. (n.d.). Retrieved August 17, 2018, from http://robotics.csie.ncku.edu.tw/Databases/FaceDetect_PoseEstimate.htm
- Dautenhahn, K. (2007). *Socially intelligent robots: Dimensions of human-robot interaction. Philosophical transactions of the Royal Society of London. Series B, Biological sciences* (Vol. 362). <https://doi.org/10.1098/rstb.2006.2004>
- Dautenhahn, K., & Billard, A. (1999). Bringing up robots or—the psychology of socially intelligent robots: From theory to implementation. *Proceedings of the Third Annual Conference on Autonomous Agents*, 366–367. <https://doi.org/10.1145/301136.301237>
- Deary, I. J., Corley, J., Gow, A. J., Harris, S. E., Houlihan, M., Marioni, R. E., ... Rafnsson, S. B. (2009). Age-associated cognitive decline, 135–152. <https://doi.org/10.1093/bmb/ldp033the>
- Definição ou significado de robô no Dicionário Infopédia da Língua Portuguesa. (n.d.). Retrieved April 2, 2018, from <https://www.infopedia.pt/dicionarios/lingua-portuguesa/robô>
- Definition of robot - Merriam-Webster's Student Dictionary. (n.d.). Retrieved April 2, 2018, from <http://wordcentral.com/cgi-bin/student?robot>
- Faust, R. A. (2007). *Robotics in Surgery: History, Current and Future Applications*. (R. A. Faust, Ed.). New York: Nova Science Publishers, Inc.
- Feil-seifer, D., & Matari, M. J. (2005). Defining Socially Assistive Robotics. In *9th International Conference on Rehabilitation Robotics* (pp. 465–468). Chicago.
- Feil-Seifer, D., & Mataric, M. (2009). *Human-Robot Interaction. Encyclopedia of Complexity*

- and Systems Science. https://doi.org/10.1007/978-0-387-30440-3_274
- Feil-Seifer, D., & Mataric, M. J. (2011). Socially assistive robotics: Ethical issues related to technology. *IEEE Robotics and Automation Magazine*, 18(1), 24–31. <https://doi.org/10.1109/MRA.2010.940150>
- Feil-Seifer, D., & Mataric, M. J. (2005). Defining socially assistive robotics. *Proceedings of the 2005 IEEE 9th International Conference on Rehabilitation Robotics*, 2005(May), 465–468. <https://doi.org/10.1109/ICORR.2005.1501143>
- Fleming, M. K., & Cottrell, G. W. (1990). Categorization of faces using unsupervised feature extraction. In *1990 IJCNN International Joint Conference on Neural Networks* (pp. 65–70 vol.2). <https://doi.org/10.1109/IJCNN.1990.137696>
- Fong, T., Nourbakhsh, I., & Dautenhahn, K. (2003). A survey of socially interactive robots. *Robotics and Autonomous Systems*, 42(3–4), 143–166. [https://doi.org/10.1016/S0921-8890\(02\)00372-X](https://doi.org/10.1016/S0921-8890(02)00372-X)
- Forsyth, D. A., & Ponce, J. (2012). Finding templates using classifiers. In *Computer Vision: A Modern Approach* (p. 595).
- Gallego Pérez, J., Lohse, M., & Evers, V. (2014). D6.3: Acceptability of a home companion robot Project acronym: ACCOMPANY Project title: Acceptable robotiCs COMPanions for AgeiNg Years Doc. Status: Final Doc. Nature: Report Version: 1.31. Retrieved from <https://cordis.europa.eu/docs/projects/cnect/4/287624/080/deliverables/001-ACCOMPANYD63Acceptabilityofhomerobotfinal.pdf>
- General Python FAQ — Python 2.7.15 documentation. (n.d.). Retrieved June 21, 2018, from <https://docs.python.org/2/faq/general.html>
- Graf, B., Reiser, U., Hägele, M., Mauz, K., & Klein, P. (2009). Robotic home assistant Care-O-bot®3 - product vision and innovation platform. In *2009 IEEE Workshop on Advanced Robotics and its Social Impacts* (pp. 139–144). <https://doi.org/10.1109/ARSO.2009.5587059>
- Grauman, K., & Darrell, T. (2005). The Pyramid Match Kernel : Discriminative Classification with Sets of Image Features. In *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1* (pp. 1458–1465). Beijing, China: IEEE. <https://doi.org/10.1109/ICCV.2005.239>
- Hadid, A., Pietikainen, M., & Ahonen, T. (2004). A discriminative feature space for detecting and recognizing faces. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.* (Vol. 2, pp. II–II). <https://doi.org/10.1109/CVPR.2004.1315246>
- Hans, M., Graf, B., & Schraft, R. D. (2002). Robotic home assistant Care-O-bot: past-present-future. In *Proceedings. 11th IEEE International Workshop on Robot and Human Interactive Communication* (pp. 380–385). <https://doi.org/10.1109/ROMAN.2002.1045652>
- Hassaballah, M., Ali, A., & Alshazly, H. (2016). *Image Features Detection, Description and Matching* (Vol. 630). https://doi.org/10.1007/978-3-319-28854-3_2
- Heerink, M., Krose, B., Evers, V., & Wielinga, B. (2006). The Influence of a Robot's Social Abilities on Acceptance by Elderly Users. In *ROMAN 2006 - The 15th IEEE*

- International Symposium on Robot and Human Interactive Communication* (pp. 521–526). <https://doi.org/10.1109/ROMAN.2006.314442>
- Heerink, M., Krose, B., Evers, V., & Wielinga, B. (2007). Observing conversational expressiveness of elderly users interacting with a robot and screen agent. In *2007 IEEE 10th International Conference on Rehabilitation Robotics* (pp. 751–756). <https://doi.org/10.1109/ICORR.2007.4428509>
- Heikkilä, M., & Pietikainen, M. (2006). A texture-based method for modeling the background and detecting moving objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(4), 657–662. <https://doi.org/10.1109/TPAMI.2006.68>
- Hjelmås, E., & Low, B. K. (2001). Face Detection: A Survey. *Computer Vision and Image Understanding*, 83(3), 236–274. <https://doi.org/10.1006/cviu.2001.0921>
- Hockstein, N. G., Gourin, C. G., Faust, R. A., & Terris, D. J. (2007). A history of robots: from science fiction to surgical robotics. *Journal of Robotic Surgery*, 1(2), 113–118. <https://doi.org/10.1007/s11701-007-0021-2>
- Howe, R. D., & Matsuoka, Y. (1999). Robotics for Surgery. *Annual Review of Biomedical Engineering*, 211–240. <https://doi.org/10.1146/annurev.bioeng.1.1.211>
- Huang, D., Shan, C., Ardabilian, M., Wang, Y., & Chen, L. (2011). Local Binary Patterns and Its Application to Facial Image Analysis: A Survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 41(6), 765–781. <https://doi.org/10.1109/TSMCC.2011.2118750>
- Jimmy, B., & Jose, J. (2011). Patient Medication Adherence: Measures in Daily Practice. *Oman Medical Journal*, 26(3), 155–159. <https://doi.org/10.5001/omj.2011.38>
- Joints — Aldebaran 2.1.4.13 documentation. (n.d.). Retrieved June 8, 2018, from http://doc.aldebaran.com/2-1/family/robots/joints_robot.html
- Juell, P., & Marsh, R. (1996). A hierarchical neural network for human face detection. *Pattern Recognition*, 29(5), 781–787. [https://doi.org/10.1016/0031-3203\(95\)00129-8](https://doi.org/10.1016/0031-3203(95)00129-8)
- Kachouie, R., Sedighadeli, S., Khosla, R., & Chu, M. T. (2014). Socially Assistive Robots in Elderly Care: A Mixed-Method Systematic Literature Review. *International Journal of Human-Computer Interaction*, 30(5), 369–393. <https://doi.org/10.1080/10447318.2013.873278>
- Kidd, C. D., & Breazeal, C. (2005). Sociable Robot Systems for Weight Maintenance. *Health (San Francisco)*.
- Kidd, C. D., Taggart, W., & Turkle, S. (2006). A Sociable Robot to Encourage Social Interaction among the Elderly. In IEEE (Ed.), *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006*. (pp. 3972–3976). Orlando, FL, USA. <https://doi.org/10.1109/ROBOT.2006.1642311>
- Kite-Powell, J. (n.d.). This Healthcare Companion Robot Will Remind You To Take Your Medicine. Retrieved May 26, 2018, from <https://www.forbes.com/sites/jenniferhicks/2018/02/27/this-healthcare-companion-robot-will-remind-you-to-take-your-medicine/#3a782c7111cc>
- Kittmann, R., Fröhlich, T., Schäfer, J., Reiser, U., Weißhardt, F., & Haug, A. (2015). Let me

- Introduce Myself: I am Care-O-bot 4, a Gentleman Robot. In *Mensch und Computer 2015* (pp. 223–232). Stuttgart.
- Koubaroulis, D., & Kittler, J. (2002). Evaluating Colour-Based Object Recognition Algorithms Using the SOIL-47 Database. In *The 5th Asian Conference on Computer Vision* (pp. 23–25). Melbourne, Australia.
- Kramer, O. (2013). K-Nearest Neighbors. In *Dimensionality Reduction with Unsupervised Nearest Neighbors* (pp. 13–23). Berlin, Heidelberg: Springer Berlin Heidelberg.
https://doi.org/10.1007/978-3-642-38652-7_2
- Krebs, H. I., Dipietro, L., Volpe, B. T., & Hogan, N. (2003). Rehabilitation Robotics : Performance-Based Progressive Robot-Assisted Therapy. *Autonomous Robots*, 15(1), 7–20.
- Kumar, G., & Bhatia, P. K. (2014). A Detailed Review of Feature Extraction in Image Processing Systems. In *2014 Fourth International Conference on Advanced Computing & Communication Technologies* (pp. 5–12). <https://doi.org/10.1109/ACCT.2014.74>
- Kung, S. Y., & Taur, J. S. (1995). Decision-based neural networks with signal/image classification applications. *IEEE Transactions on Neural Networks*, 6(1), 170–181.
<https://doi.org/10.1109/72.363439>
- Lafortune, G. (2007). Trends in Severe Disability Among Elderly People, (26).
<https://doi.org/10.1787/18152015>
- Lawrence, S., Giles, C. L., Tsoi, A. C., & Back, A. D. (1997). Face recognition: a convolutional neural-network approach. *IEEE Transactions on Neural Networks*, 8(1), 98–113. <https://doi.org/10.1109/72.554195>
- Li, S. Z., & Jain, A. K. (2011). *Handbook of Face Recognition* (2nd ed.). Springer Publishing Company, Incorporated.
- Lin, J., Sklar, G. E., Oh, V. M. Sen, & Li, S. C. (2008). Factors affecting therapeutic compliance: A review from the patient's perspective. *Therapeutics and Clinical Risk Management*, 4(1), 269–286. <https://doi.org/10.2147/TCRM.S1458>
- Lin, S.-H., Kung, S.-Y., & Lin, L.-J. (1997). Face recognition/detection by probabilistic decision-based neural network. *IEEE Transactions on Neural Networks*, 8(1), 114–132.
<https://doi.org/10.1109/72.554196>
- Loudspeakers — Aldebaran 2.1.4.13 documentation. (n.d.). Retrieved June 10, 2018, from http://doc.aldebaran.com/2-1/family/robots/loudspeaker_robot.html
- Lowe, D. G. (1999). Object recognition from local scale-invariant features. In *Proceedings of the Seventh IEEE International Conference on Computer Vision* (Vol. 2, pp. 1150–1157 vol.2). <https://doi.org/10.1109/ICCV.1999.790410>
- Lowe, D. G. (1999). Object recognition from local scale-invariant features. *Proceedings of the Seventh IEEE International Conference on Computer Vision*, 1150–1157 vol.2.
<https://doi.org/10.1109/ICCV.1999.790410>
- Lowe, D. G. (2004). Distinctive image features from scale invariant keypoints. *International Journal of Computer Vision*, 60, 91–11020042.
<https://doi.org/10.1023/B:VISI.0000029664.99615.94>

- Lüdtke, M. (2016). The service robot Care-O-bot 4. *CAN Newsletter*, 1(1/2016), 36–39. Retrieved from https://www.care-o-bot.de/content/dam/careobot/en/documents/Publications/2016_02_CANOpen-Newsletter_Care-O-bot4.pdf
- Malik, J., Belongie, S., Leung, T. K., & Shi, J. (2001). Contour and Texture Analysis for Image Segmentation. *International Journal of Computer Vision*, 43(1), 7–27. <https://doi.org/10.1023/A:1011174803800>
- Meet Pillo | Your Personal Home Health Companion. (n.d.). Retrieved May 25, 2018, from https://www.pillohealth.com/144/pillo_pillbox_reminder
- Microphones — Aldebaran 2.1.4.13 documentation. (n.d.). Retrieved June 10, 2018, from http://doc.aldebaran.com/2-1/family/robots/microphone_robot.html
- Minear, M., & Park, D. C. (2004). A lifespan database of adult facial stimuli. *Behavior Research Methods, Instruments, & Computers*, 36(4), 630–633. <https://doi.org/10.3758/BF03206543>
- Motherboard. (n.d.). Retrieved June 9, 2018, from http://doc.aldebaran.com/2-1/family/robots/motherboard_robot.html
- NAO - Battery — Aldebaran 2.1.4.13 documentation. (n.d.). Retrieved June 10, 2018, from http://doc.aldebaran.com/2-1/family/robots/battery_robot.html
- NAO - Construction. (n.d.). Retrieved June 9, 2018, from http://doc.aldebaran.com/2-1/family/robots/dimensions_robot.html
- NAO - Technical overview. (n.d.). Retrieved June 10, 2018, from http://doc.aldebaran.com/2-1/family/robots/index_robots.html#all-robots
- NAO - Video camera — Aldebaran 2.1.4.13 documentation. (n.d.). Retrieved June 11, 2018, from http://doc.aldebaran.com/2-1/family/robots/video_robot.html
- NAOqi Framework — NAO Software 1.14.5 documentation. (n.d.). Retrieved June 14, 2018, from <http://doc.aldebaran.com/1-14/dev/naoqi/index.html>
- NAOqi OS - Getting started. (n.d.). Retrieved June 9, 2018, from <http://doc.aldebaran.com/2-1/dev/tools/opennao.html?highlight=gentoo>
- Niederauer, J. (2008). Introdução ao Guia. In *Integrando Php 5 com Mysql* (p. 6). Novatec Editora Ltda. Retrieved from <http://www.martinsfontespaulista.com.br/anexos/produtos/capitulos/526209.pdf>
- Nixon, M. S., & Aguado, A. S. (2012). Chapter 8 - Introduction to texture description, segmentation, and classification. In *Feature Extraction & Image Processing for Computer Vision* (pp. 399–434). Academic Press. <https://doi.org/10.1016/B978-0-12-396549-3.00008-2>
- Numpy - PyScience-Brasil. (n.d.). Retrieved June 30, 2018, from <http://pyscience-brasil.wikidot.com/module:numpy>
- NumPy — NumPy. (n.d.). Retrieved June 30, 2018, from <http://www.numpy.org/>
- O. M. Parkhi, A. Vedaldi, A. Zisserman, C. V. J. (2012). Cats and Dogs. In *IEEE Conference on Computer Vision and Pattern Recognition*. Providence, RI, USA: IEEE.

- <https://doi.org/10.1109/CVPR.2012.6248092>
- Ojala, T., Pietikäinen, M., & Harwood, D. (1996). A comparative study of texture measures with classification based on featured distributions. *Pattern Recognition*, 29(1), 51–59. [https://doi.org/10.1016/0031-3203\(95\)00067-4](https://doi.org/10.1016/0031-3203(95)00067-4)
- Opelt, A., Pinz, A., Fussenegger, M., & Auer, P. (2006). Generic object recognition with boosting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(3), 416–431. <https://doi.org/10.1109/TPAMI.2006.54>
- OpenNAO - NAO OS — NAO Software 1.14.5 documentation. (n.d.). Retrieved June 13, 2018, from <http://doc.aldebaran.com/1-14/dev/tools/opennao.html#opennao-advanced>
- Overview — Pillow (PIL Fork) 5.2.0 documentation. (n.d.). Retrieved June 30, 2018, from <https://pillow.readthedocs.io/en/5.2.x/handbook/overview.html>
- Pan, Y., Ge, S. S., He, H., & Chen, L. (2009). Real-time face detection for human robot interaction. In *RO-MAN 2009 - The 18th IEEE International Symposium on Robot and Human Interactive Communication* (pp. 1016–1021). <https://doi.org/10.1109/ROMAN.2009.5326349>
- Parmar, D. N., & Mehta, B. B. (2013). Face Recognition Methods & Applications. *Int. J. Computer Technology & Applications*, 4(1), 84–86.
- PARO Therapeutic Robot. (n.d.). Retrieved May 10, 2018, from <http://www.parorobots.com/>
- Pillow — Pillow (PIL Fork) 5.2.0 documentation. (n.d.). Retrieved June 30, 2018, from <https://pillow.readthedocs.io/en/5.2.x/>
- Pogue, D. (2017). David Pogue tested 47 pill-reminder apps to find the best [Video]. Retrieved May 25, 2018, from <https://finance.yahoo.com/news/david-pogue-tested-47-pill-reminder-apps-to-find-the-best-154017023.html?guccounter=1>
- Programming — Aldebaran 2.1.4.13 documentation. (n.d.). Retrieved July 1, 2018, from http://doc.aldebaran.com/2-1/dev/programming_index.html
- Python SDK — NAO Software 1.14.5 documentation. (n.d.). Retrieved June 22, 2018, from <http://doc.aldebaran.com/1-14/dev/python/index.html>
- Reis, A., Paredes, H., Barroso, I., Monteiro, M. J., Rodrigues, V., Khanal, S. R., ... Ieee. (2016). Autonomous systems to support social activity of elderly people A prospective approach to a system design. In *2016 1st International Conference on Technology and Innovation in Sports, Health and Wellbeing (TISHW)*. Vila Real, Portugal: IEEE. <https://doi.org/10.1109/TISHW.2016.7847773>
- Ribeiro, L. S. (2008). *Object recognition for semantic robot vision*. Universidade de aveiro.
- Rublee, E., Rabaud, V., Konolige, K., & Bradski, G. (2011). ORB: An efficient alternative to SIFT or SURF. *Proceedings of the IEEE International Conference on Computer Vision*, 2564–2571. <https://doi.org/10.1109/ICCV.2011.6126544>
- Saerbeck, M. (2009). *Software Architecture for Social Robots*. Technische Universiteit Eindhoven. Retrieved from http://www.bartneck.de/publications/2009/softwareArchitectureSocialRobots/software_architecture_for_social_robots.pdf

- Sarle, W. S. (1994). Neural Networks and Statistical Modelse.
- schedule — schedule 0.4.0 documentation. (n.d.). Retrieved June 23, 2018, from <https://schedule.readthedocs.io/en/stable/>
- Scopelliti, M., & Vittoria, Æ. M. (2005). Robots in a domestic setting : a psychological approach. *Universal Access in the Information Society*, 4(2), 146–155. <https://doi.org/10.1007/s10209-005-0118-1>
- SDKs — Aldebaran 2.5.10.8 documentation. (n.d.). Retrieved June 14, 2018, from http://doc.aldebaran.com/2-5/dev/programming_index.html
- Shah, D. (2017). Socially assistive robotics. In *Socially Assistive Robotics*.
- Shapiro, L. G., & Stockman, G. C. (2001). Pattern recognition concepts. In *Computer Vision* (pp. 107–144).
- Shibata, T. (2012). Therapeutic seal robot as biofeedback medical device: Qualitative and quantitative evaluations of robot therapy in dementia care. *Proceedings of the IEEE*, 100(8), 2527–2538. <https://doi.org/10.1109/JPROC.2012.2200559>
- SHUKLA, D., Desai, A., & Publication, I. (2015). *REVIEW ON GENERIC OBJECT RECOGNITION TECHNIQUES: CHALLENGES AND OPPORTUNITIES*. *International Journal of Advanced Research in Engineering and Technology (IJARET)* (Vol. 6).
- Silva, J. I. S. da. (2015). *Reconhecimento facial em imagens de baixa resolução*. Universidade Federal de Pernambuco. Retrieved from <https://repositorio.ufpe.br/handle/123456789/16367>
- Simen, A. A., Bordner, K. A., Martin, M. P., Moy, L. A., & Barry, L. C. (2011). Cognitive Dysfunction with Aging and the Role of Inflammation. *Therapeutic Advances in Chronic Disease*, 2(3), 175–195. <https://doi.org/10.1177/2040622311399145>
- Tapus, A., Mataric, M., & Scassellati, B. (2007). Socially assistive robotics: The grand challenges in helping humans through social interaction. *IEEE Robotics & Automation Magazine*, 14(1), 35–42. <https://doi.org/10.1109/MRA.2007.339605>
- Tejima, N. (2001). Rehabilitation robotics : a review. *Advanced Robotics*, 14(7), 37–41. <https://doi.org/10.1163/156855301742003>
- Terrillon, J., David, M., & Akamatsu, S. (1998). Automatic Detection of Human Faces in Natural Scene Images by Use of a Skin Color Model and of Invariant Moments. In *Proceedings Third IEEE International Conference on Automatic Face and Gesture Recognition* (pp. 112–117). Nara, Japan: IEEE. <https://doi.org/10.1109/AFGR.1998.670934>
- Tinker, A., & Lansley, P. (2005). Introducing assistive technology into the existing homes of older people: Feasibility, acceptability, costs and outcomes. *Journal of Telemedicine and Telecare*, 11(1_suppl), 1–3. <https://doi.org/10.1258/1357633054461787>
- Tzafestas, S. (2016). *Sociorobot World* (Vol. 80). Springer International Publishing. <https://doi.org/10.1007/978-3-319-21422-1>
- Unimate - The First Industrial Robot. (n.d.). Retrieved April 3, 2018, from <https://www.robotics.org/joseph-engelberger/unimate.cfm>

- United Nations. (2007). *World Population Ageing 2007. World Population Ageing 2007 (Population Studies Series)*. New York. Retrieved from http://www.un.org/esa/population/publications/WPA2007/Full_text.pdf
- Vaillant, R., Monrocq, C., & Cun, Y. Le. (1994). Original approach for the localisation of objects in images. *IEE Proceedings - Vision, Image and Signal Processing*, 141(4), 245–250. <https://doi.org/10.1049/ip-vis:19941301>
- Viola, P., & Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*. Kauai, HI, USA, USA: IEEE. <https://doi.org/10.1109/CVPR.2001.990517>
- Wada, K., & Shibata, T. (2007). Social effects of robot therapy in a care house - Change of social network of the residents for two months. In *Proceedings - IEEE International Conference on Robotics and Automation* (pp. 1250–1255). Roma, Italy. <https://doi.org/10.1109/ROBOT.2007.363156>
- Who is NAO? (n.d.). Retrieved June 8, 2018, from <https://www.softbankrobotics.com/emea/en/robots/nao>
- Yanco, H. A., & Drury, J. L. (2002). A Taxonomy for Human-Robot Interaction. In *2004 IEEE International Conference on Systems, Man and Cybernetics*. The Hague, Netherlands: IEEE. <https://doi.org/10.1109/ICSMC.2004.1400783>
- Yang, M.-H., Kriegman, D. J., & Ahuja, N. (2002). Detecting faces in images: a survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(1), 34–58. <https://doi.org/10.1109/34.982883>
- Yuille, A. L., Cohen, D. S., & Hallinan, P. W. (1989). Feature extraction from faces using deformable templates. In *Proceedings CVPR '89: IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (pp. 104–109). San Diego, CA, USA, USA: IEEE. <https://doi.org/10.1109/CVPR.1989.37836>
- Zhao, W., & Rosenfeld, A. (2003). Face Recognition : A Literature Survey. *ACM Computing Surveys (CSUR)*, 35(4), 399–458. <https://doi.org/10.1145/954339.954342>