

APLICAÇÃO DE TÉCNICAS DE FUSÃO/INTEGRAÇÃO SENSORIAL DE DADOS NO LEVANTAMENTO DO RELEVO DE OBJECTOS

Jaime Francisco Cruz Fonseca

Tese submetida na Universidade do Minho
para obtenção do grau
de Doutor
em Engenharia de Electrónica Industrial



Universidade do Minho

Outubro 1999

Dedico este trabalho:

À minha esposa Carla e à minha filha Ana Cristina com a idade deste trabalho e tendo crescido ao ritmo dele, é contudo muito melhor e maior que ele.

Agradecimentos

Na realização de trabalhos que se estendem por longos períodos de tempo, é natural contactar com numerosas pessoas que de uma forma ou de outra contribuíram para o desenvolvimento do mesmo. Este trabalho não fugiu a esta regra e por esse facto quero deixar aqui expresso o meu agradecimento às contribuições que a memória ainda preserva e pedir compreensão às que esquecer.

Em primeiro lugar queria agradecer aos meus orientadores, Prof. Doutor Carlos Alberto Caridade Monteiro e Couto e Prof. Doutor Júlio Manuel Sousa Barreiro Martins toda a confiança em mim depositada, encorajamento, apoio e orientação prestada ao longo deste trabalho. Não queria ainda deixar passar em claro esta oportunidade para agradecer aos meus orientadores, a consideração e amizade com que sempre me trataram desde a minha chegada ao departamento de Electrónica Industrial.

Queria agradecer ao Prof. Doutor Paulo Guimarães Garrido e à Eng. Cristina Santos toda a colaboração e sugestões efectuadas ao longo da preparação das provas de mestrado da última, assim como, toda a disponibilidade demonstrada por ambos ao longo de todo o período de duração deste trabalho, para comigo discutir as mais diversas opções de implementação do mesmo.

Ao Sr. Abel Oliveira e ao Carlos Torres queria deixar o meu agradecimento por toda a assistência técnica, sugestões e disponibilidade demonstrada ao longo de todo o trabalho.

Aos meus colegas Eng. João Luiz Afonso, Eng. Carlos Manuel Gregório dos Santos Lima, e na fase final ao Eng. José de Araújo Mendes todas as conversas, opiniões e fundamentalmente o apoio ao longo destes anos. Queria ainda expressar o meu agradecimento a todos os restantes colegas que de alguma forma contribuíram para este trabalho.

Agradeço ainda aos restantes funcionários do Departamento de Electrónica Industrial nomeadamente à Sra. Maria Beatriz Antunes e à menina Teresa Peixoto, toda a colaboração e disponibilidade que sempre evidenciaram.

Queria ainda agradecer aos meus sogros Valentim Rodrigues e Natália Vieira Rodrigues toda a disponibilidade que sempre demonstraram em me ajudar durante estes anos, mas também toda a consideração e amizade com que sempre me distinguiram desde o instante em que os conheci.

Finalmente queria deixar aqui o meu agradecimento às pessoas que têm um lugar muito especial no meu coração, os meus pais e irmã a quem tudo devo.

Resumo

O presente trabalho de investigação situa-se na área da robótica fixa inteligente apoiada pela fusão/integração sensorial de dados. A utilização de sistemas baseados em multi-sensores tem permitido o aparecimento de novas áreas e o desenvolvimento de outras já existentes. O reconhecimento de objectos, a robótica móvel com aplicações essencialmente ao nível da navegação, as técnicas de navegação para aeronaves e as tarefas industriais automatizadas, como a montagem, a rebarbagem ou o polimento, são alguns exemplos de aplicações em que os sistemas multi-sensor apresentam não só vantagens como são imprescindíveis à sua implementação.

A tese descreve um sistema experimental de levantamento do relevo de superfícies com base na aplicação de técnicas de fusão/integração sensorial de dados. Este sistema é constituído por um manipulador PUMA 560, cuja função é o posicionamento dos sensores nele montados por forma a que seja possível efectuar a aquisição do relevo; por uma rede CAN (*Controller Area Network*) para interligação dos sensores à estrutura de controlo; sensores de distância por ultrassons e uma câmara de vídeo.

No âmbito deste trabalho foram efectuados os seguintes estudos e realizadas as seguintes tarefas:

- 1) Avaliação das características do manipulador PUMA 560 e suas potencialidades, nomeadamente do seu controlador. Este foi substituído por apresentar limitações para o trabalho que se pretendia realizar, ficando o controlo do PUMA incorporado no PC. Foram desenvolvidas/corrigidas rotinas que implementam a cinemática directa e inversa, assim como algumas instruções úteis que facilitam a execução de determinadas tarefas por parte do robô.
- 2) Estudo da rede de comunicação de dados baseada no protocolo CAN para interligação dos sensores de ultrassons. Foi implementado o protocolo da camada de aplicação.
- 3) Estudo e implementação de técnicas de processamento de imagem para extracção dos pontos dominantes do contorno do objecto. A aplicação destas técnicas tem como finalidade a restrição da área de trabalho do robô.
- 4) Estudo dos problemas associados com a significância dos dados provenientes de ultrassons, de que resultou a adopção de um novo modelo para os mesmos.
- 5) Estudo de técnicas de fusão/integração sensorial de dados, das quais foram implementadas a média, lógica *fuzzy*, filtro de *Kalman* e métodos heurísticos.

Destes estudos salientam-se os seguintes resultados: a adopção de um novo modelo para os ultrassons; a utilização de diferentes técnicas de fusão/integração sensorial de dados com realce para a implementação de um “decisor” e não de um controlador *fuzzy*, a sua forma “nativa”.

Com base no que anteriormente foi descrito, foi possível fazer o levantamento do relevo de diferentes superfícies, representando-o de seguida num mapa de superfície. Tendo este como suporte será possível definir trajectórias para o seguimento de superfícies necessária a aplicações industriais que não requerem contacto, como é o caso da pulverização à pistola quer para a pintura quer na aplicação de colas diluentes ou outros líquidos.

Abstract

The present research work scope is in the area of intelligent static robotic supported by multisensor data fusion and integration. The usage of multisensor systems allowed the emergence of new research areas and further development of current areas. Automatic target recognition, mobile robot navigation, aircraft navigation, industrial tasks such as assembly, burnishing and trimming are typical multisensor system applications. In these cases, multiple sensors are not only advantageous but they are essential to their implementation.

This thesis describes an experimental system for the acquisition of a surface profile using techniques of multisensor data fusion. This system is composed by the following elements: a PUMA 560 manipulator, which function is the positioning of the sensors mounted on the wrist flange of the robot, in order to acquire the surface profile; a controller area network (CAN) used for data acquisition and some basic control; ultrasonic sensors and a video camera.

For this research project the following studies and tasks were carried out:

- 1) Evaluation of the features and performances of the PUMA 560 manipulator and its controller, which was replaced due to its limitations for this work. In the new architecture a PC performs the control of the PUMA. The inverse and forward kinematics routines were implemented and corrected. A few instructions were also implemented for simplifying some tasks of the robot.
- 2) Study of a CAN protocol based network and the interface with the ultrasonic sensors. The protocol for the application level was implemented.
- 3) Study and implementation of image processing techniques for dominant points extraction from the object contour. The goal of these techniques is to limit the robot working area.
- 4) Study of the problem of ultrasonic data significance, which required the adoption of a new model for the sonar sensors.
- 5) Study of the multisensor data fusion and integration techniques. The average, fuzzy logic, Kalman filter and heuristic methods were implemented.

The most important results obtained were: a new model for ultrasonic sensors was adopted; different techniques for multisensor data fusion and integration were used and in particular an implementation of a fuzzy “arbitration” method.

As a result, it was possible to perform the acquisition of a surface profile that was represented in a surface map.

1. Introdução	1
1.1 Motivações e objectivos	2
1.2 Contribuições	2
1.3 Organização da tese	3
2. Fusão/Integração sensorial de dados	5
2.1 Introdução	5
2.2 Fundamentos da fusão sensorial	7
2.3 Revisão das técnicas mais utilizadas em fusão sensorial	11
2.3.1 Métodos ao nível do sinal	11
2.3.2 Métodos ao nível do ponto/imagem	14
2.3.3 Métodos ao nível do símbolo	16
2.4 Sumário	35
3. Arquitecturas de controlo para sistemas multi-sensoriais	36
3.1 Introdução	36
3.2 Aproximação cognitivista, comportamental e híbrida	38
3.2.1 Aproximação cognitivista	38
3.2.2 Aproximação comportamental	39
3.2.3 Aproximação híbrida	43
3.3 O Modelo de Referência RCS	46
3.3.1 Elementos de um sistema inteligente	48
3.3.2 Arquitectura para sistemas inteligentes	51
3.4 Arquitectura de controlo adoptada	53
3.5 Sumário	59
4. A célula de trabalho	60
4.1 Introdução	60
4.2 Robôs do tipo manipulador	60
4.2.1 O manipulador PUMA 560	61
4.3 Sensores	64
4.3.1 Sensor de visão	65
4.3.2 Sensores de ultrassons	66
4.3.3 Sensores de infravermelhos	73
4.4 Rede CAN	74
4.5 Sumário	76
5. Sistema de aquisição e comunicação de dados	78
5.1 Introdução	78
5.2 O CAN	78
5.2.1 Formato das mensagens	79
5.2.2 Mecanismos de detecção de erros no CAN	81
5.2.3 O protocolo CAN	82
5.3 Software da camada de aplicação	83
5.4 Sumário	85
6. Sistema de aquisição de imagem	86

6.1	Introdução	86
6.2	Visão por computador	86
6.2.1	Iluminação	87
6.2.2	Filtragem	89
6.2.3	Segmentação	90
6.2.4	Descrição	93
6.3	Detalhes da implementação	99
6.3.1	Resultados experimentais	101
6.4	Sumário	102
7.	<i>Aquisição do relevo</i>	<i>103</i>
7.1	Introdução	103
7.2	Introdução ao problema da aquisição do relevo	104
7.3	Modelo para o sensor de ultrassons	105
7.3.1	Modelo utilizado	106
7.4	Determinação das dimensões do objecto	110
7.5	Aquisição do relevo da superfície visível	113
7.5.1	Fusão/Integração sensorial de dados provenientes de ultrassons	116
7.6	Sumário	123
8.	<i>Resultados experimentais</i>	<i>124</i>
8.1	Introdução	124
8.2	Testes realizados	125
8.3	Apreciação dos resultados	137
9.	<i>Conclusões e sugestões para trabalho futuro</i>	<i>138</i>
9.1	Conclusões	138
9.2	Sugestões para trabalho futuro	140

Capítulo 1

1. Introdução

A inclusão de múltiplos sensores em máquinas e sistemas é um dos principais senão o principal requisito para introduzir alguma inteligência nessas entidades, permitindo-lhes agir em ambientes não estruturados sem o completo controlo do operador humano. Os sensores permitem ao sistema apreender o estado do meio ambiente, sendo essa informação posteriormente representada por um modelo. Este necessita de uma actualização contínua emergindo de imediato duas necessidades: grande capacidade de armazenamento para que a representação do ambiente seja o mais fidedigna possível; grande velocidade de processamento para tratar os dados provenientes das várias fontes em tempo útil.

A entidade mais eficiente no processamento de dados é o ser humano. A quantidade de informação que este pode processar e recordar é absolutamente espantosa e nenhum computador, apesar dos grandes avanços tecnológicos, quer ao nível da capacidade de processamento, quer das capacidades de memória consegue competir. Por outro lado, a tecnologia de fabrico dos sensores tem igualmente tido uma grande evolução nos últimos anos, tendo como consequências principais o aparecimento de uma maior diversidade de sensores e um decréscimo geral do seu custo. Do ponto de vista económico, passou a ser interessante avaliar a substituição de um sensor de grande precisão mas de elevado custo por alguns sensores de baixo custo mas de menor precisão. Esta conjuntura favoreceu o emergir de uma nova área de investigação que se denominou fusão/integração multi-sensorial de dados, que tem sob o seu domínio o estudo da combinação de dados provenientes de múltiplos sensores. O termo fusão sensorial é normalmente utilizado para designar uma fase do processo de integração onde é combinada a informação proveniente de múltiplas fontes e o termo integração sensorial significa a combinação/representação dos resultados de processos de fusão sensorial.

A relação estreita entre a utilização de múltiplos sensores e a necessidade de processar grande quantidade de dados implicou não só um aumento, como também uma melhoria da electrónica de pós-processamento, de forma a dotar esta de capacidade para realçar as vantagens de cada um dos sensores e suavizar as suas limitações. A combinação dos dados provenientes de múltiplas fontes pode ser efectuada com recurso a diferentes técnicas, sendo as seguintes as mais usuais: a média pesada, o filtro de *Kalman*, a estimação *Bayesiana*, o método de *Dempster-Shafer* e os métodos heurísticos. Presentemente começam a ser utilizadas técnicas de *soft-computing* nomeadamente a lógica *fuzzy* e as redes neuronais. Todas estas técnicas são aplicadas com frequência conjuntamente ou sequencialmente.

1.1 Motivações e objectivos

O objectivo do trabalho descrito nesta tese é o levantamento do relevo a partir dos dados provenientes de múltiplos sensores de baixo custo e posterior representação dessa informação num mapa de superfície.

No respeitante à parte sensorial, optou-se pela utilização de sensores de ultrassons, infravermelhos (posteriormente abandonados por razões que mais tarde se apresentam) e uma câmara de vídeo monocromática. Em termos de infra-estrutura experimental os sensores foram montados num manipulador PUMA 560, sendo a câmara de vídeo colocada sobre o ombro do robô e os ultrassons no respectivo punho. Os objectos cujo relevo se pretende adquirir são colocados sobre uma mesa semi-circular, com uma altura igual à do ombro do manipulador e dentro da área de trabalho do mesmo (anexo V).

Para delimitar a área de trabalho do robô o único sensor utilizado foi a câmara de vídeo. Na execução desta tarefa o robô efectua um movimento circular através de incrementos de posição na sua junta 1, posicionando a câmara de vídeo a apontar para diferentes localizações. A câmara encontra o objecto por comparação de imagens. Por este motivo a cor da mesa e a do objecto terão que ser suficientemente contrastantes. Após a conclusão desta tarefa são utilizadas técnicas de processamento de imagem, cuja finalidade é a extracção dos pontos dominantes do contorno.

A partir da informação fornecida pela câmara, o robô é inicialmente posicionado de forma a que os ultrassons fiquem a apontar para o objecto. Em seguida, alguns sensores de ultrassons são activados, sendo os dados por estes adquiridos utilizados para complementar a informação da câmara, obtendo-se desta forma as dimensões mais próximas das reais do objecto (altura x largura).

Em seguida posiciona-se novamente o robô tendo em consideração os resultados anteriores. Procede-se então ao varrimento da superfície do objecto com os ultrassons, sendo estes apontados para novos pontos da superfície por deslocamentos do manipulador segundo um plano paralelo ao plano de projecção da imagem. Assim, o robô nesta primeira fase é utilizado como um “digitalizador”.

As motivações para o desenvolvimento deste trabalho abrangeram diversas vertentes, podendo ser resumidas nos seguintes pontos:

1. Estudo da problemática associada a sensores como os de ultrassons e infravermelhos, nomeadamente no que respeita à interpretação das medidas que fornecem.
2. Selecção de um barramento que possibilitasse a interligação de diferentes sensores à estrutura de controlo e ao mesmo tempo apresentasse características que tornasse viável a sua transposição para ambientes industriais.
3. Exploração de técnicas de fusão/integração sensorial.
4. Utilização do conhecimento adquirido nos pontos anteriores para dotar um manipulador de inteligência, permitindo-lhe posteriormente o desempenho de diferentes tarefas ou da mesma tarefa em ambientes distintos de uma forma mais flexível (sem necessidade de reprogramação).

1.2 Contribuições

A principal contribuição deste trabalho resume-se na concepção e desenvolvimento de um sistema para levantamento do relevo de superfícies com base em informação proveniente de diferentes sensores de baixo custo. Este sistema, apresenta todas as condições para ser

utilizado em tarefas que não exijam uma grande precisão nem contacto como são por exemplo os casos da pulverização à pistola quer para a pintura quer na aplicação de colas, diluentes ou outros líquidos em superfícies, sem necessidade de reprogramação do robô.

Os passos que foram efectuados e que permitiram no global concretizar esta contribuição foram os seguintes:

- Implementação de um novo modelo para os ultrassons com introdução de dois novos conceitos: pontos de profundidade constante (PPC) e regiões de profundidade constante (RPC). A utilização e os resultados obtidos com base neste modelo ajudam a desmistificar um pouco a usual má reputação dos ultrassons quando usados em tarefas que requerem maior precisão.
- Concepção e desenvolvimento de uma arquitectura para fusão/integração sensorial de dados, recorrendo a diferentes técnicas de fusão de uma forma não simultânea e sendo a selecção destas efectuada a partir de uma pré-avaliação dos dados recebidos.
- Utilização da lógica *fuzzy* como técnica de fusão sensorial de dados. Foi implementado um “decisor” *fuzzy* cuja finalidade é atribuir um grau de crença às medidas efectuadas num ponto com base na relação existente entre estas e as efectuadas em pontos vizinhos. Nesta situação a lógica *fuzzy* não foi utilizada na sua forma mais tradicional, isto é, na implementação de um controlador.
- Utilização de um filtro de *Kalman* em paralelo com o “decisor” *fuzzy* (sem simultaneidade) para estimar a distância a um ponto da superfície do objecto. O filtro atribui os graus de crença nas medidas efectuadas com base na variância do vector de medida adquirido.
- Utilização de técnicas de integração sensorial de dados (método heurístico) como método de decisão em zonas limites do objecto.

1.3 Organização da tese

Esta tese foi estruturada em três partes. A primeira compreende os capítulos 2 e 3, que abordam o estado da arte, nomeadamente no contexto da fusão/integração multi-sensorial e das arquitecturas de controlo para estes sistemas. No capítulo 2 são apresentados os fundamentos da fusão sensorial de dados e as técnicas mais utilizadas. No capítulo 3 são abordadas as arquitecturas de controlo para sistemas com multi-sensores. Neste capítulo são apresentadas as aproximações cognitivista, comportamental e híbrida, bem como o modelo de referência RCS (*real-time control system*) do Albus. Descreve-se ainda o modelo adoptado.

A segunda parte compreende os capítulos 4-7, que apresentam os detalhes da implementação. No capítulo 4 descreve-se a célula de trabalho utilizada nesta aplicação, assim como algumas das opções relativamente às configurações de *hardware* adoptadas. No capítulo 5 apresenta-se o sistema de aquisição e comunicação de dados sobre uma rede CAN (*Controller Area Network*). No capítulo 6 é abordado o sistema de visão por computador. Finalmente, o capítulo 7 descreve a aquisição do relevo, dando ênfase ao modelo adoptado para os ultrassons e às técnicas de fusão/integração sensorial de dados implementadas.

A terceira e última parte da tese compreende os capítulos 8 e 9. No capítulo 8 apresenta-se os resultados experimentais obtidos. O capítulo 9 diz respeito às conclusões e a sugestões para trabalhos futuros.

Inclui-se ainda um conjunto de anexos que proporcionam informações sobre as placas utilizadas, nomeadamente a placa da *Polaroid Inc.* que processa os dados dos ultrassons, as placas de comunicação de dados que possibilitaram a implementação da rede CAN e ainda as

placas que substituíram o antigo controlador do PUMA 560. Encontra-se ainda em anexo um conjunto de fotografias da infra-estrutura experimental montada para este trabalho.

Capítulo 2

2. Fusão/Integração sensorial de dados

2.1 Introdução

Nos últimos anos, a tecnologia de fabrico dos sensores sofreu um grande desenvolvimento tendo como principais consequências o aparecimento de uma maior diversidade de dispositivos e um decréscimo geral dos preços. Do ponto de vista económico passou a ser interessante substituir um sensor de grande precisão mas de elevado custo por vários sensores de baixo custo mas de menor precisão. A combinação de múltiplos sensores com estas características implicou um aumento da electrónica de pós-processamento com capacidade de realçar as suas vantagens individuais e suavizar as suas limitações, minimizando desta forma os efeitos da menor precisão dos mesmos.

Em consequência desta conjuntura favorável uma nova disciplina emergiu denominada fusão multi-sensorial de dados ou sensorização distribuída [Hall, 92]. A especificidade do vocabulário relacionado com a mesma não se encontra convenientemente definido. Numa tentativa de padronizar a terminologia a utilizar foi constituída uma comissão formada pelos directores de grupos de investigação na área, oficializada em 1986, que erigiu a seguinte definição [Tech Reach Inc, 95]:

A continuous process dealing with the association, correlation, and combination of data and information from multiple sources to achieve refined entity position and identify estimates and complete and timely assessments of resulting situations and threats, and their significance.

A **associação**, a **correlação** e a **combinação** são formas de relacionar os dados no processo de fusão sensorial. A escolha da técnica de fusão sensorial depende do tipo de dados a serem combinados e do processo de conclusão/inferência pretendido.

Luo e Kay em [Luo and Kay, 89] complementaram a definição anterior separando a fusão multi-sensorial de dados em duas partes distintas: fusão sensorial/dados e integração sensorial/dados. **Fusão sensorial** é referida como uma fase do processo de integração onde é combinada a informação proveniente de múltiplas fontes; **Integração sensorial** significa a combinação/representação dos resultados de processos de fusão sensorial.

A Figura 2.1 apresenta genericamente um sistema de fusão e integração sensorial de dados. Enquanto os processos de fusão são efectuados nos nós, a estrutura integral da rede

conjuntamente com as funções de integração fazem parte do processo de integração sensorial. As linhas a tracejado representam a possibilidade das funções de integração existentes no “sistema” enviar sinais para cada um dos nós. A “selecção de sensores”, o “modelo do ambiente” e a “transformação de dados” são funções típicas de um processo de integração.

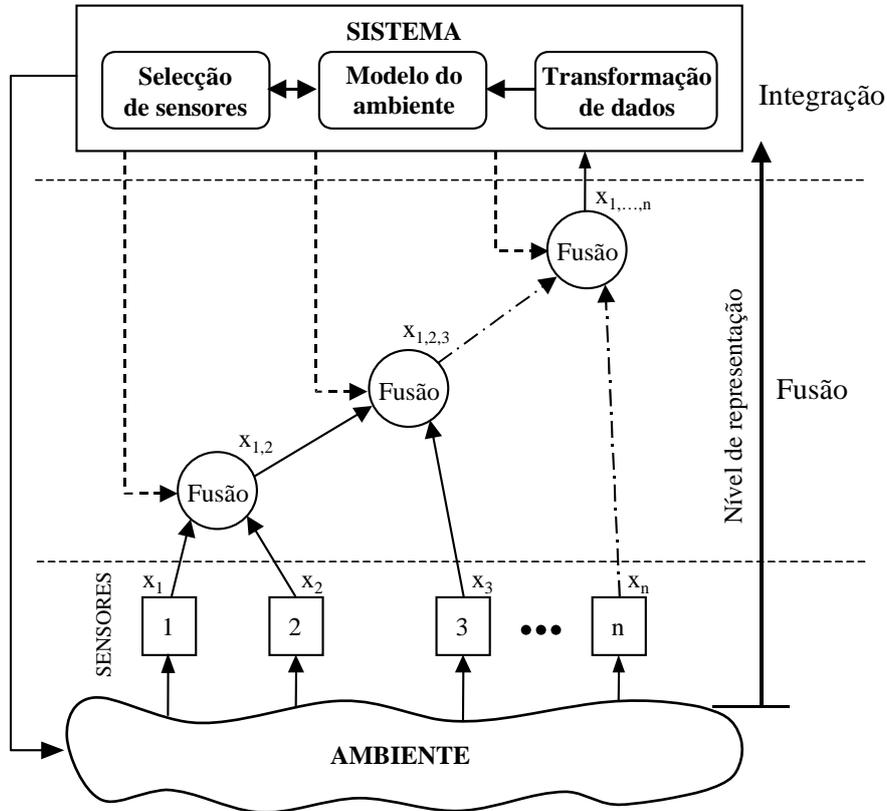


Figura 2.1 Sistema genérico de fusão e integração sensorial de dados.

A função “selecção de sensores” permite a um sistema multi-sensorial seleccionar as configurações mais apropriadas (ou estratégias de sensorização), a partir do grupo de sensores disponíveis no sistema. Podem ser distinguidas duas aproximações para a selecção do tipo, número e configuração dos sensores a serem utilizados por um sistema: **pré-selecção**, efectuada na fase de projecto e por conseguinte dependente de um estudo prévio quer das necessidades globais do sistema quer das características individuais dos sensores a utilizar; **tempo real**, utilizada para escolher dentro do grupo existente de sensores os mais apropriados para responder a alterações no ambiente ou a condições do sistema. É ainda de salientar que um sistema pode utilizar a pré-selecção como um processo de inicialização passando de seguida a um processo de selecção em tempo-real.

A função “modelo do ambiente” permite a um sistema multi-sensorial guardar a informação sensorial previamente adquirida, decidindo com base nessa informação. Esta função corresponde a um processo de representação de mais alto nível estando normalmente associada com o modelo de arquitectura adoptado.

Finalmente, os dados resultantes de diferentes processos de fusão podem necessitar de ser transformados antes da sua representação no modelo do ambiente, estando essa tarefa a cargo da função “transformação de dados”.

Neste capítulo são unicamente abordados os fundamentos e técnicas da fusão sensorial tendo sido dividido em duas secções. Na primeira secção é feita uma apresentação dos fundamentos da fusão sensorial. Na segunda secção são apresentadas as técnicas mais usuais

na combinação de dados, destacando-se o filtro de *Kalman* e a lógica *fuzzy* pelo facto de serem utilizados neste trabalho.

A função de integração “modelo do ambiente” pelo facto de estar associada à arquitectura de controlo adoptada é apresentada no capítulo 3 (“Arquitecturas de controlo para sistemas multi-sensoriais”); a função “selecção de sensores” mais concretamente a aproximação denominada pré-selecção é demonstrada no capítulo 4 (“célula de trabalho”); a função “transformação de dados” utiliza as mesmas técnicas do processo de fusão sensorial.

2.2 Fundamentos da fusão sensorial

O objectivo de um sistema de fusão sensorial é transformar os dados provenientes de vários sensores em processos de inferência ou decisão abstractos, de uma forma similar ao que o homem faz com os dados provenientes dos seus sentidos (visão, tacto, olfacto, audição e sabor), tal como se apresenta na Figura 2.2.

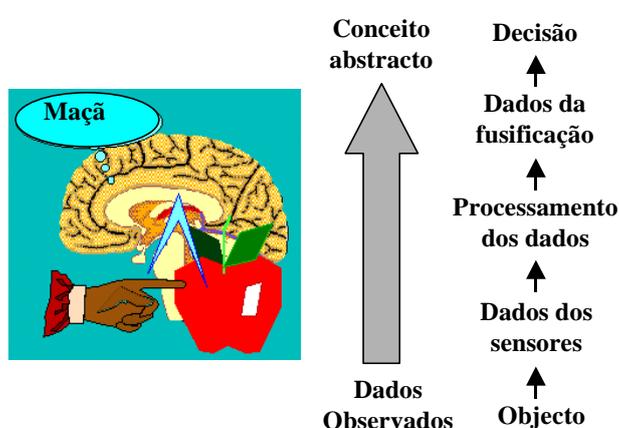


Figura 2.2 Relação entre o que o ser humano faz e um sistema de fusão sensorial [Tech Reach Inc, 95].

O termo fusão sensorial, apesar de parecer bastante complexo, envolve poucas ideias essenciais: múltiplas entradas, uma ou mais técnicas de fusão, o resultado desejado (ou inferência) e um mecanismo de controlo do sistema (Figura 2.3).

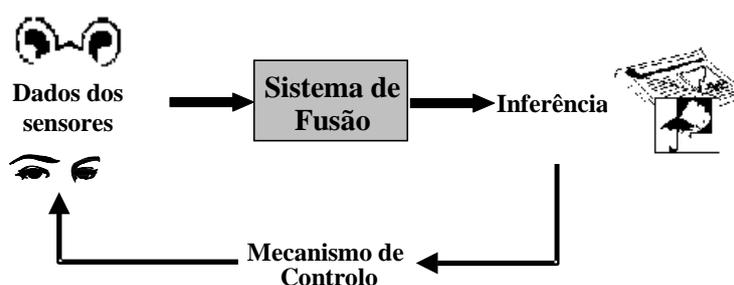


Figura 2.3 Elementos que constituem um sistema de fusão sensorial.

Os dados provenientes dos múltiplos sensores foram classificados por Nashman [Nashman, 93] em três categorias baseando-se na interacção estabelecida entre eles. Essas interacções são denominadas de **interacções competitivas**, **complementares** e **cooperativas**.

A **interacção competitiva** é definida como a relação entre sensores que observam o mesmo objecto adquirindo características dependentes. A informação fornecida por estes sensores é denominada de redundante quando não contribuem com qualquer conhecimento adicional para o sistema. A integração de sensores competitivos resulta num incremento da confiança na validade da informação extraída quando existe uma relação entre os dados adquiridos e num abaixamento da confiança quando nenhuma relação existe. A Figura 2.4

descreve este tipo de interacção num sistema constituído por 2 sensores S_1 e S_2 , os quais produzem as saídas redundantes d_1 e d_2 . Os processos T_1 e T_2 são constituídos por algoritmos que convertem as saídas em unidades relacionadas com a representação de sensores individuais no sistema. Estes resultados são enviados para um terceiro processo que transporta as saídas para um referencial comum.

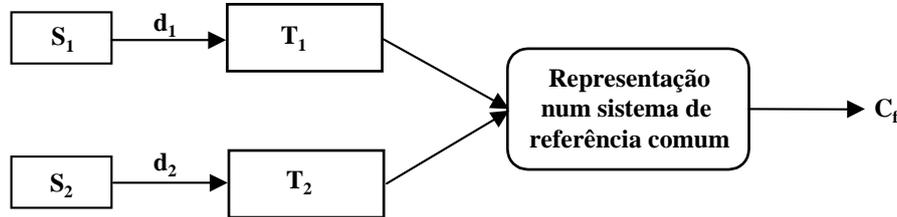


Figura 2.4 Interacção de informação competitiva.

A **interacção complementar** ocorre quando dois ou mais sensores obtêm informação acerca do mesmo objecto mas as características adquiridas por cada sensor são independentes umas das outras. Neste caso cada sensor fornece informação parcial acerca de determinadas características do ambiente. A Figura 2.5 descreve um exemplo deste tipo de interacção em que os processos T_1 e T_2 produzem informação independente. Não é necessário um processo para combinar as saídas de T_1 e T_2 , mas a observação conjunta das saídas C_a e C_b permite uma melhor percepção do objecto observado.

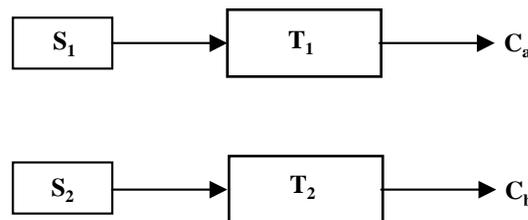


Figura 2.5 Interacção de informação complementar.

A **interacção cooperativa** ocorre quando as observações de um ou mais sensores guiam as acções de um outro. A Figura 2.6 descreve este tipo de interacção. A saída do sensor S_1 (d_1) é processada em T_1 . A saída deste é usada para guiar as acções do sensor S_2 , o qual processa a sua saída d_2 em T_2 produzindo a saída C_i . A linha a tracejado que vai do processo T_1 para S_2 representa uma realimentação. O sensor S_1 utiliza essa entrada para se auto-posicionar. O ciclo de realimentação não tem que estar presente num sistema cooperativo mas a sua presença adiciona capacidades ao processo de sensorização.

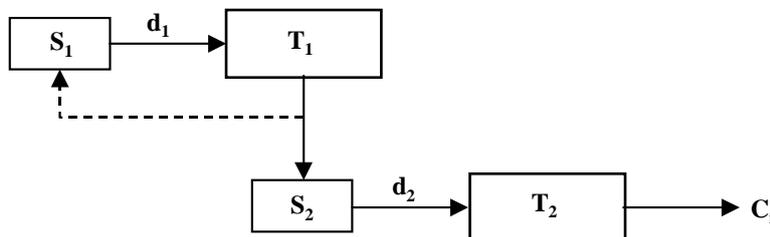


Figura 2.6 Interacção de informação cooperativa.

A fusão sensorial da informação adquirida por múltiplos sensores ou por um único sensor num determinado instante pode efectuar-se em diferentes níveis de representação. Uma categorização usual é considerar o processo de fusão sensorial dividido em 4 níveis [Luo and Kay, 95]: ao nível do **signal**, ao nível da **característica**, ao nível do **ponto/imagem** e ao nível do **símbolo**. A grande maioria dos sensores utilizados na prática fornece informação que pode ser combinada em um ou mais destes níveis, podendo envolver a utilização de diversas

técnicas e abranger várias disciplinas (Figura 2.7). Estas técnicas são frequentemente aplicadas em conjunto ou sequencialmente.

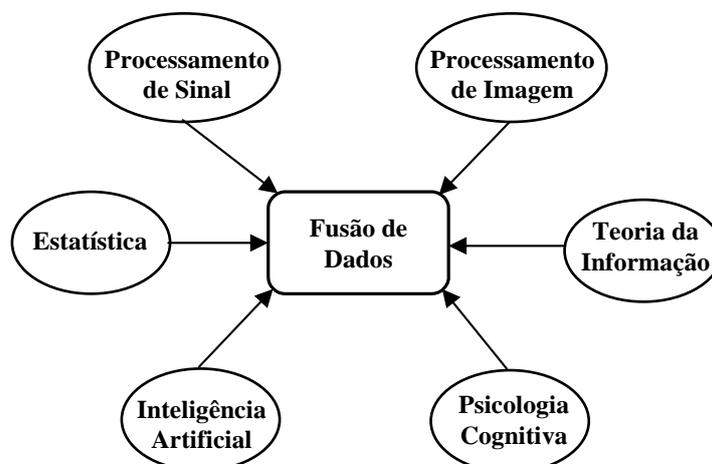


Figura 2.7 Disciplinas que integram diferentes técnicas de fusão de dados.

A fusão ao nível do **senal** refere-se à combinação dos sinais provenientes de um grupo de sensores com o objectivo de fornecer um sinal que é usualmente do mesmo tipo dos originais, mas com melhor qualidade. Um exemplo das técnicas de fusão utilizadas ao nível do sinal é a média pesada dos sinais, onde os pesos são baseados na variância estimada dos mesmos. Este tipo de fusão pode ser usado em aplicações em tempo real podendo considerar-se como um passo adicional em todo o processamento de sinal.

A fusão ao nível do **ponto/imagem** é exclusivamente utilizada para fornecer uma melhoria no desempenho de algumas tarefas de processamento de imagem, como por exemplo a segmentação. Métodos de fusão característicos deste nível são, por exemplo, os filtros lógicos e os operadores morfológicos.

A fusão ao nível da **característica** pode ser utilizada com duas finalidades: para incrementar a probabilidade de a característica extraída a partir da informação fornecida por um sensor corresponder a um aspecto importante do ambiente; ou como meio de gerar características compostas adicionais para posterior utilização pelo sistema. Características típicas extraídas a partir de uma imagem e utilizadas no processo de fusão são os contornos e regiões de similar intensidade ou profundidade. Métodos ao nível da característica são, por exemplo, a estimação *Bayesiana* e o método de *Dempster-Shafer*.

A fusão ao nível do **símbolo** permite que a informação de vários sensores possa ser utilizada com um alto nível de abstracção. Este tipo de fusão pode ser somente um meio de combinar a informação sensorial quando os sensores são muito diferentes ou quando observam diferentes características do ambiente. As decisões são normalmente tomadas com base na comparação dessas características com a informação retirada do modelo. Os símbolos usados têm normalmente associado com eles um grau de crença que indica o peso que essa informação sensorial tem no modelo. Métodos ao nível do símbolo são, por exemplo, a estimação *Bayesiana*, método de *Dempster-Shafer*, os factores de confiança e a lógica *fuzzy*.

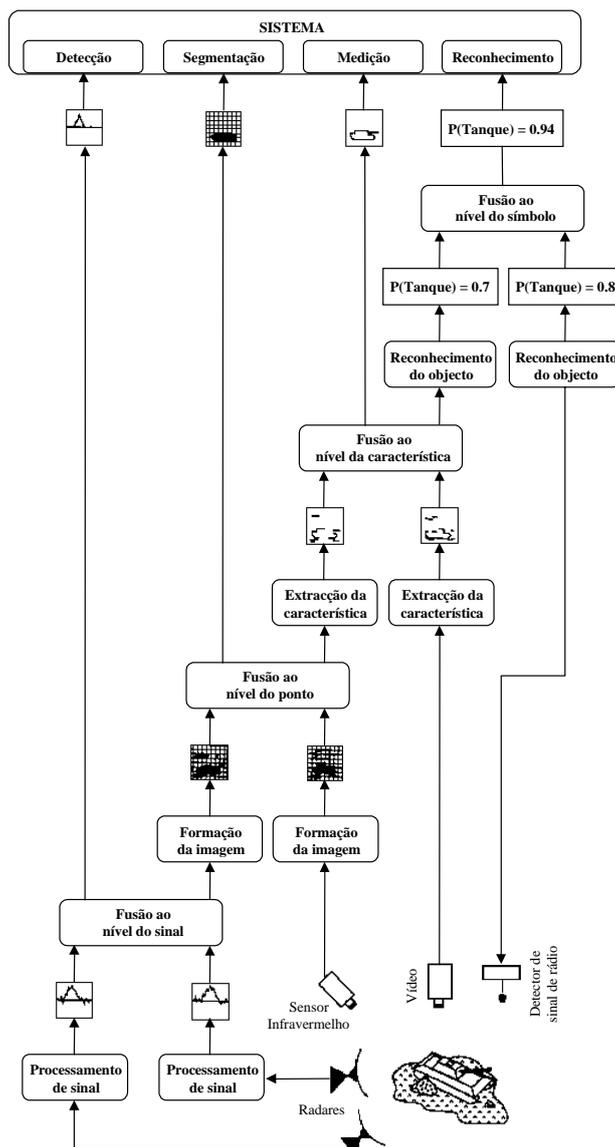


Figura 2.8 Diferentes níveis de fusão sensorial (nível do sinal, ponto, característica e símbolo) no reconhecimento automático de um tanque [Luo and Kay, 95].

Os diferentes níveis de fusão sensorial diferenciam-se em função das seguintes características: tipo de informação fornecida ao sistema, o modo como a informação é modelada, os métodos usados no processo de fusão e a técnica de fusão utilizada para melhorar a qualidade da informação fornecida ao sistema. A Figura 2.8 fornece um exemplo de como os diferentes níveis de fusão sensorial podem ser utilizados no reconhecimento automático de objectos.

Os procedimentos de inferência são projectados para trabalhar com representações particulares da informação disponível. A selecção de um método específico baseia-se normalmente no tipo de informação disponível para o processo de raciocínio, o tipo de saída pretendido, a complexidade computacional e até que ponto as hipóteses são necessárias e passíveis de serem transportadas para se obter uma saída. A inferência pode então definir-se como o processo de derivar “inteligência” a partir da fusão sensorial usando premissas com uma conclusão estritamente lógica ou com um intervalo de confiança que pode ser probabilístico ou intervalo de evidência (Figura 2.9).

O primeiro tipo de premissas é particularmente usado para raciocinar em situações onde a informação disponível é lógica, isto é estados de crença ou descrença.

O segundo tipo de premissas é fundamentalmente baseado em modelos da teoria probabilística. Eles fornecem meios para representar a incerteza em torno do estado de crença absoluta e com frequência caminhos para representar a imprecisão ou ignorância.

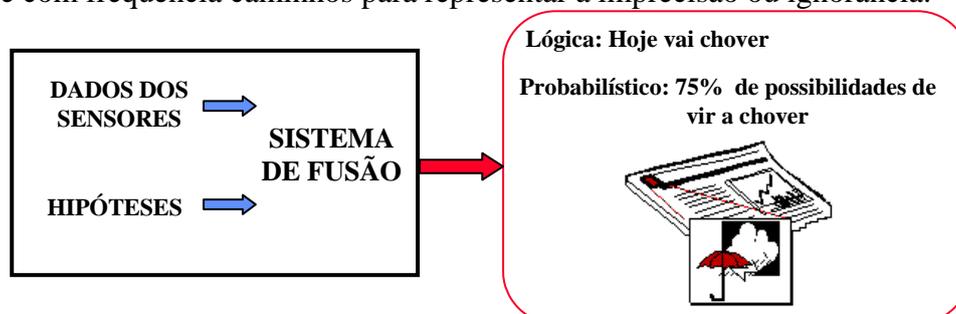


Figura 2.9 Processo de inferência.

2.3 Revisão das técnicas mais utilizadas em fusão sensorial

Neste ponto são discutidas as técnicas de fusão sensorial mais usuais divididas pelos níveis anteriormente referidos, sendo dado um maior realce aos métodos ao nível do sinal e do símbolo por serem os de maior relevância para este trabalho.

2.3.1 Métodos ao nível do sinal

A fusão sensorial ao nível do sinal é considerada como um processo de estimação quando os sinais provenientes de cada sensor são modelados como variáveis aleatórias. Os modelos mais comuns incluem o uso de um modelo de medida para a informação adquirida por cada sensor, com a inclusão de um erro Gaussiano estatisticamente independente ou ruído periódico. As técnicas de fusão sensorial mais usuais neste nível são: Média pesada e Filtro de *Kalman*.

Média Pesada

Um dos mais simples e intuitivos métodos de fusão ao nível do sinal é a média pesada da informação redundante fornecida por um grupo de sensores. Este método foi usado para fusão sensorial por exemplo em [Bauzil et al, 81] para eliminar medidas errôneas.

A média pesada da medida de n sensores x_i com pesos $0 \leq w_i \leq 1$ é dada por

$$\bar{x} = \sum_{i=1}^n w_i x_i \quad (2.1)$$

onde,

$$\sum_i w_i = 1 \quad \text{e} \quad w_i = 0 \quad \text{se} \quad x_i \quad \text{não se encontrar dentro de um limite pré-estabelecido.}$$

Os pesos podem ser usados para distinguir diferenças de precisão entre sensores, sendo ainda possível utilizar-se uma média “deslizante” para combinar uma sequência de medidas adquiridas por um único sensor, atribuindo-se às mais recentes um maior peso.

Filtro de Kalman

Em 1960, R. E. Kalman publicou um artigo descrevendo uma solução recursiva para o problema da filtragem linear de dados discretos. Desde essa publicação, devido em grande

parte aos avanços da computação digital, o filtro de *Kalman* foi objecto de intensa investigação e aplicação, particularmente na área da navegação autónoma ou assistida.

No filtro de *Kalman* o comportamento de um sistema é caracterizado por um conjunto de relações lineares (expressão 2.2) em que o pãrametro que indica o estado do sistema é recursivamente actualizado através da aquisição de novas medidas. Como as medidas também são susceptíveis de erro é igualmente utilizado um segundo modelo, também linear (expressão 2.3), para o processo de medida. A partir deste modelo é calculada uma estimativa para as medidas e se estas forem iguais às novas medidas adquiridas o sistema atinge a estabilidade. O filtro usa então as características estatísticas do modelo de medida de um processo para recursivamente calcular uma estimativa do seu estado. Para tal, as expressões que caracterizam o filtro de *Kalman* dividem-se em dois grupos: expressões de predição e expressões de correcção. As primeiras determinam uma estimativa à *priori* do estado actual do processo e da covariância do erro (estado actual – estado anterior); as segundas são responsáveis pela recursividade, isto é, inserem as novas medidas na estimativa efectuada à *priori* para o estado do processo gerando uma estimativa à *posteriori* melhorada (Figura 2.10). Um factor de ganho é igualmente calculado em cada iteração para minimizar a covariância do erro. A estimativa à *posteriori* é em seguida utilizada para determinar uma nova estimativa à *priori*. A natureza recursiva do filtro torna-o apropriado para ser usado em sistemas sem grande capacidade de armazenamento de dados.

As medidas obtidas por um grupo de n sensores podem ser combinadas usando um filtro de *Kalman*, de forma a este fornecer uma estimativa do estado corrente e uma predição do futuro estado do sistema. O estado estimado pode corresponder, por exemplo, à posição actual de um robô móvel, à posição e velocidade de um objecto no ambiente ou a características extraídas a partir de dados dos sensores (por exemplo: contorno de uma imagem).

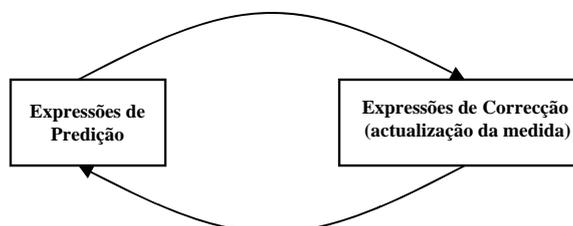


Figura 2.10 Relação entre as expressões de predição e de correcção.

O processo a estimar

O filtro de *Kalman* constitui uma das possibilidades para se encontrar uma solução para o problema genérico da estimativa do estado de uma variável $x \in \mathfrak{R}^n$, inserida num processo de controlo discreto no tempo, caracterizado por uma equação de diferenças estocástica linear [Welch and Bishop, 95]

$$x_{k+1} = A_k x_k + B u_k + w_k \quad (2.2)$$

com um modelo de medida descrito por

$$z_k = H_k x_k + v_k \quad (2.3)$$

x_k : vector de estado ($x_k \in \mathfrak{R}^n$).

A_k : matriz $n \times n$ que relaciona o estado no instante k com o estado no instante $k + 1$.

B : matriz $n \times 1$ que relaciona a entrada de controlo.

u_k : vector de entrada.

z_k : vector de medida ($z_k \in \mathfrak{R}^m$).

H : matriz de medida $m \times n$ que relaciona o estado com a medida z_k .

w_k : vector de ruído do processo.

v_k : vector de ruído da medida.

As variáveis aleatórias w_k e v_k representam respectivamente o ruído do processo e da medida. Assumindo que estas variáveis possuem as seguintes características: são independentes uma da outra; ruído branco; e apresentam uma característica de distribuição de probabilidade normal

$$P(w) \sim N(0, Q)$$

$$P(v) \sim N(0, R)$$

onde, Q e R são respectivamente as matrizes de covariância do processo e da medida.

Parte computacional do filtro

Definindo $\hat{x}_{k-1} \in \mathfrak{R}^n$ como a estimativa do estado à *priori* do instante k dado o conhecimento já existente do processo e $\hat{x}_k \in \mathfrak{R}^n$ como a estimativa do estado à *posteriori* do instante k dado o vector de medida z_k . Pode-se então definir uma estimativa para os erros à *priori* e à *posteriori* através das seguintes expressões

$$e_{k-1} = x_k - \hat{x}_{k-1}, \quad e \quad e_k = x_k - \hat{x}_k \quad (2.4)$$

A estimativa da covariâncias dos erros são expressos por

$$P_{k-1} = E[e_{k-1} e_{k-1}^T] \quad (2.5)$$

$$P_k = E[e_k e_k^T] \quad (2.6)$$

O processo de dedução das equações do filtro de *Kalman* inicia-se com a procura de uma expressão que descreva a estimativa do estado à *posteriori* \hat{x}_k , a partir da combinação linear da estimativa à *priori* \hat{x}_{k-1} e da diferença pesada entre a medida actual z_k e a medida prevista $H_k \hat{x}_{k-1}$. A expressão que descreve a predição do estado \hat{x}_k é a seguinte

$$\hat{x}_k = \hat{x}_{k-1} + K(z_k - H_k \hat{x}_{k-1}). \quad (2.7)$$

A diferença entre $(z_k - H_k \hat{x}_{k-1})$ é denominada de medida residual, em que o termo residual se refere à discrepância entre a medida prevista $H_k \hat{x}_{k-1}$ e a medida actual z_k . O K (matriz $n \times m$) é escolhido para ser um factor de ganho que permita minimizar a covariância do erro P_k .

Os passos necessários para efectuar a minimização são os seguintes:

1. Substituir a expressão 2.7 em e_k (expressão 2.4)

$$e_k = x_k - \hat{x}_{k-1} + K(z_k - H_k \hat{x}_{k-1}) \quad (2.8)$$

2. Substituir a expressão 2.8 na expressão 2.6

$$P_k = E \left[\left(x_k - \hat{x}_{k-1} + K(z_k - H_k \hat{x}_{k-1}) \right) \times \left(x_k - \hat{x}_{k-1} + K(z_k - H_k \hat{x}_{k-1}) \right)^T \right] \quad (2.9)$$

e determinar a esperança matemática.

3. Calcula-se a derivada da expressão resultante do passo anterior em ordem a K e iguala-se a zero. Deste processo obtêm-se uma expressão para K (expressão 2.10) que minimiza a expressão 2.9.

$$K_k = P_{k-1} H_k^T (H_k P_{k-1} H_k^T + R_k)^{-1} = \frac{P_{k-1} H_k^T}{H_k P_{k-1} H_k^T + R_k} \quad (2.10)$$

Da expressão 2.10 é possível concluir que à medida que o R_k tende para 0 o ganho K é cada vez maior, ou seja

$$\lim_{R_k \rightarrow 0} K_k = \frac{1}{H_k}.$$

Quando a covariância do erro estimado P_{k-1} se aproxima de zero, o ganho K é cada vez menor, ou seja

$$\lim_{P_{k-1} \rightarrow 0} K_k = 0.$$

Uma outra forma de pensar acerca da variação do K é a seguinte: quando a covariância do erro de medida R_k se aproxima de zero, a confiança na medida efectuada z_k é cada vez maior enquanto a confiança na medida prevista $H_k \hat{x}_{k-1}$ diminui; quando a covariância da estimativa do erro P_{k-1} se aproxima de zero, a confiança na medida efectuada z_k é cada vez menor enquanto aumenta a confiança na medida prevista $H_k \hat{x}_{k-1}$.

2.3.2 Métodos ao nível do ponto/imagem

A fusão sensorial ao nível do ponto utiliza-se para melhorar o conteúdo da informação fornecida por uma imagem, de forma a que por exemplo o processo de segmentação seja mais preciso e discriminador das características relevantes a serem extraídas em futuros processamentos. A fusão neste nível efectua-se em diferentes níveis de representação: a fusão de sinais provenientes de múltiplos sensores e sua associação com um ponto específico; a fusão dos pontos correspondentes de múltiplas imagens para obtenção de uma imagem composta; o uso de pontos ou grupos de pontos localizados em múltiplas imagens para segmentação e extracção da característica ao nível do ponto (por exemplo: extracção do contorno).

Duas técnicas são particularmente usadas neste nível, filtros lógicos e morfologia matemática.

Filtros lógicos

Um dos métodos mais intuitivos para a combinação de dados a partir de pontos é o dos operadores lógicos. Estes são ferramentas básicas no processamento de imagens binárias, sendo usados em tarefas como detecção de características e análise de contornos. As principais operações lógicas são o AND, o OR e o NOT. A Figura 2.11 apresenta alguns

exemplos da aplicação destes operadores lógicos, onde o 1 e o 0 correspondem respectivamente ao preto e ao branco.

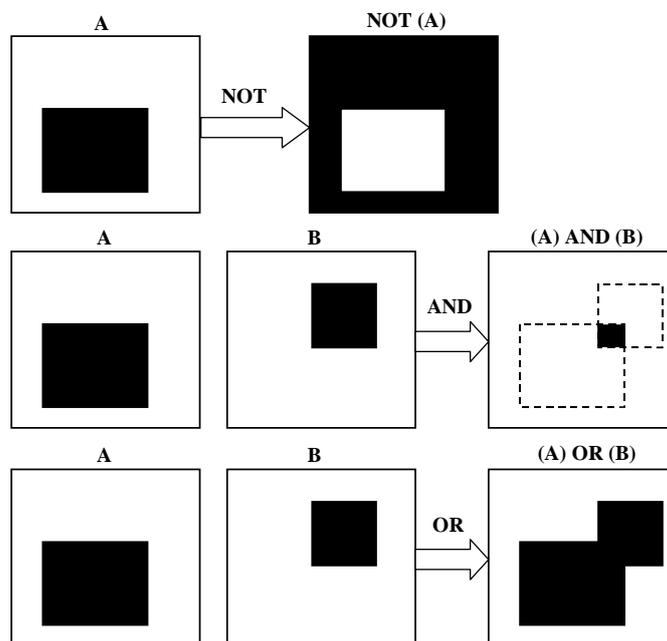


Figura 2.11 Alguns exemplos de operações lógicas em imagens binárias [Gonzalez and Woods, 92].

Morfologia matemática

A morfologia matemática é um método de análise da imagem que transforma cada ponto de uma imagem através do uso de um conjunto de operadores morfológicos derivados das operações básicas com conjuntos: união, intersecção, diferença e suas combinações condicionais. Estes operadores constituem uma ferramenta útil na extracção de partes da imagem que representam e descrevem a região do contorno. As operações morfológicas mais usuais são as operações de **dilatação** e **erosão** (Figura 2.12), servindo estas igualmente de base a outras operações [Gonzalez and Woods, 92].

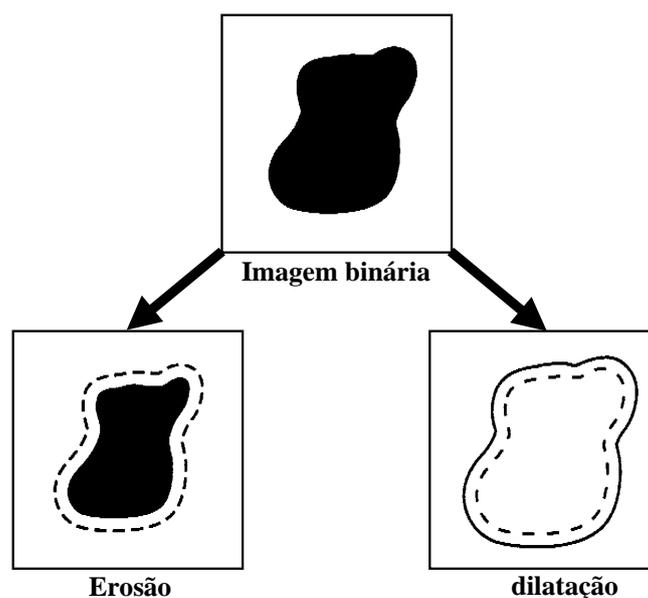


Figura 2.12 Erosão e dilatação.

2.3.3 Métodos ao nível do símbolo

O método de fusão ao nível do símbolo é o que permite o mais elevado grau de abstracção de todos os métodos já referidos. Quatro métodos são particularmente usados: estimação *Bayesiana*, teoria da evidência de *Dempster-Shafer*, factores de confiança e lógica *Fuzzy*. Estes métodos produzem sistemas baseados em regras.

Estimação Bayesiana

A estimação *Bayesiana* fornece um formalismo para a fusão sensorial que permite combinar a informação proveniente dos vários sensores de acordo com as regras da teoria das probabilidades. Neste formalismo, são atribuídos valores numéricos às proposições significando a crença em cada uma delas, de acordo com algum conhecimento previamente existente. A quantidade de crença a atribuir obedece a três axiomas básicos da teoria probabilística [Pearl, 88]:

$$0 \leq p(A) \leq 1 \quad (2.11)$$

$$P(\text{crença absoluta}) = 1 \quad (2.12)$$

$$P(A \text{ OR } B) = P(A) + P(B) \quad \text{Se } A \text{ e } B \text{ forem mutuamente exclusivos} \quad (2.13)$$

O terceiro axioma (expressão 2.13) exprime que a atribuição de uma quantidade de crença a um conjunto de eventos é igual à soma da quantidade de crença dos seus elementos que não se intersectam. Como consequência desta propriedade é possível escrever qualquer evento A como a união de um conjunto de eventos ($A \text{ AND } B$) e ($A \text{ AND } \bar{B}$), sendo as suas probabilidades associadas expressas por

$$P(A) = P(A, B) + P(A, \bar{B}) \quad (2.14)$$

onde $P(A, B)$ é a representação de $P(A \text{ AND } B)$. Generalizando para um conjunto de proposições mutuamente exclusivas representadas por B_i com $i = 1, 2, \dots, n$; então $P(A)$ pode ser calculado a partir de $P(A, B_i)$ através da expressão

$$P(A) = \sum_i P(A, B_i) \quad \text{com } i = 1, 2, \dots, n \quad (2.15)$$

A consequência directa das expressões (2.12) e (2.14) é que a soma da proposição com a sua negação é igual à crença absoluta (valor 1).

$$P(A) + P(\bar{A}) = 1 \quad (2.16)$$

As expressões básicas do formalismo *Bayesiano* são as probabilidades condicionais - $P(A|B)$ - a qual atribui uma quantidade de crença a A desde que a hipótese B seja conhecida com uma certeza absoluta. Se os eventos A e B são independentes então $P(A|B) = P(A)$ ou se os eventos A e B são condicionalmente independentes dado um evento C então $P(A|B, C) = P(A|C)$. A **probabilidade condicional** em termos dos eventos conjuntos é dada por a expressão

$$P(A|B) = \frac{P(A, B)}{P(B)} \quad \Leftrightarrow \quad (2.17)$$

$$P(A, B) = P(A | B)P(B) \quad (2.18)$$

De forma similar ao da expressão (2.15), é igualmente possível estabelecer uma expressão para calcular a probabilidade de qualquer evento A em função de um conjunto de eventos mutuamente exclusivos B_i com $i = 1, 2, \dots, n$:

$$P(A) = \sum_i P(A | B_i)P(B_i) \quad (2.19)$$

Esta decomposição serve de base ao formalismo *Bayesiano*, sendo denominada de **regra da eliminação**. A partir da expressão (2.19), é possível observar que a quantidade de crença atribuída a um qualquer evento A é igual à soma pesada das crenças individuais. Uma outra generalização útil para a regra do produto é a denominada *chain rule*: considerando um conjunto de n eventos, E_1, E_2, \dots, E_n , então a probabilidade dos eventos conjuntos (E_1, E_2, \dots, E_n) pode ser escrito como o produto de n probabilidades condicionais

$$P(E_1, E_2, \dots, E_n) = P(E_n | E_{n-1}, \dots, E_2, E_1) \dots P(E_2 | E_1)P(E_1) \quad (2.20)$$

Este produto pode ser determinado por aplicação repetida da expressão 2.18 segundo qualquer ordem.

O coração da técnica *Bayesiana* baseia-se na expressão conhecida como **fórmula da inversão ou regra de Bayes** [Pearl, 88]

$$P(H | e) = \frac{P(e | H)P(H)}{P(e)} \quad (2.21)$$

onde,

$P(H | e)$: É a probabilidade à *posteriori*. Representa a crença atribuída à hipótese H dada a informação representada pela evidência e .

$P(e | H)$: É a verosimilhança (*likelihood*).

$P(H)$: É a probabilidade à *priori* ou verdade prévia.

$P(e)$: É uma constante de normalização em que $P(e) = P(e | H)P(H) + P(e | \bar{H})P(\bar{H})$.

Contudo, para um matemático é incompreensível considerar a regra de *Bayes* resultante da definição de probabilidade condicional. Enquanto um adepto da teoria *Bayesiana* considera a regra de *Bayes* como um padrão para actualizar o valor da crença em resposta a uma evidência. Por outras palavras, os matemáticos consideram as probabilidades condicionais como uma construção matemática (expressão 2.17), os aderentes a *Bayes* consideram-nas como primitivas da linguagem e como uma tradução dedicada da expressão inglesa “... *given that I know A*” [Pearl, 88].

Combinação do suporte predictivo e de diagnóstico

A essência da regra de *Bayes* é convenientemente descrita usando os parâmetros “*ratio likelihood*” e “*odds*”. Dividindo a expressão (2.21) por o complementar $P(\bar{H} | e)$, obtem-se:

$$\frac{P(H | e)}{P(\bar{H} | e)} = \frac{P(e | H) P(H)}{P(e | \bar{H}) P(\bar{H})} \quad (2.22)$$

Definindo o *odds* à *priori* de H como

$$O(H) = \frac{P(H)}{\overline{P(H)}} = \frac{P(H)}{1 - P(H)}, \quad (2.23)$$

o “*ratio likelihood*”

$$L(e|H) = \frac{P(e|H)}{\overline{P(e|H)}}, \quad (2.24)$$

e os “*odds*” à *posteriori*

$$O(H|e) = \frac{P(H|e)}{\overline{P(H|e)}} = \frac{P(e|H)P(H)}{P(e|\overline{H})P(\overline{H})} = L(e|H) \times O(H). \quad (2.25)$$

Os “*odds*” e as probabilidades à *posteriori* estão relacionados pela seguinte expressão:

$$P(H|e) = \frac{P(e|H)P(H)}{P(e|H)P(H) + P(e|\overline{H})P(\overline{H})} = \frac{O(H|e)}{1 + O(H|e)}. \quad (2.26)$$

Assim, a regra de *Bayes* determina a força da crença para a hipótese H , com base no conhecimento prévio e na evidência observada, partindo do produto de dois factores: o *odds* à *priori* $O(H)$ e o *ratio likelihood* $L(e|H)$. O primeiro factor mede o suporte predictivo em relação a H partindo do conhecimento adquirido, enquanto que o segundo representa o diagnóstico dado para H pela evidência observada.

Teoria de Dempster-Shafer

A teoria da evidência de *Dempster-Shafer* (D-S) é uma extensão da aproximação *Bayesiana* que torna explícita qualquer falta de informação relacionada com a probabilidade das proposições/hipóteses por separação da crença em uma proposição da sua plausibilidade [Luo and Kay, 95]. A teoria *Bayesiana* requer a especificação completa do modelo probabilístico antes de se iniciar o “raciocínio”, usando métodos de aproximação para completar o modelo nos casos em que a especificação é parcial. O método de D-S consegue gerir modelos especificados parcialmente. Contudo, estes modelos são exclusivamente usados na representação de relações qualitativas de compatibilidades entre as proposições envolvidas. Estas relações são posteriormente utilizadas para acumular “provas” que permitam inferir as conclusões a partir das evidências.

Na teoria da evidência de D-S um conjunto Θ , denominado de **sistema de discernimento**, é composto por proposições mutuamente exclusivas denominadas **singulares**. O nível de detalhe que representa uma proposição singular corresponde ao nível mais baixo de informação que é possível diferenciar através da fusão de informação.

Assim, podemos dizer que a quantidade de crença atribuída a um subconjunto A de Θ pode em parte ser também atribuída a um ou mais subconjuntos próprios de A , sendo o restante atribuído exclusivamente a A , o que nos conduz à definição de **atribuição de probabilidade básica** (também denominada **função de massa**).

Definição – Se Θ é um sistema de discernimento, então a função $m : 2^\Theta \rightarrow [0,1]$ é chamada atribuição de probabilidade básica sempre que:

$$1. m(\emptyset) = 0 \quad \emptyset - \text{Conjunto vazio} \quad (2.27)$$

$$2. \sum_{A \subseteq \Theta} m(A) = 1 \quad (2.28)$$

em que $m(A)$ é denominado **valor de probabilidade básica** (ou **massa**) de A e é entendido como a quantidade de crença atribuída a A . Qualquer subconjunto A de Θ com $m(A) \neq 0$ é denominado de **elemento focal**.

A quantidade total de crença atribuída a A deve incluir a crença em proposições contidas em A (seus subconjuntos próprios), e é expressa pela função $Bel: 2^\Theta \rightarrow [0,1]$, denominada função de crença sobre Θ , em que $m : 2^\Theta \rightarrow [0,1]$ é a atribuição de probabilidade básica:

$$Bel(A) = \sum_{B \subseteq A} m(B) \quad (2.29)$$

Contudo, de acordo com Shafer, uma proposição A só fica completamente representada em termos de informação/ignorância se, para além de quantificarmos a nossa crença nela ($Bel(A)$), expressarmos também o quanto duvidamos dela (descrença), ou seja, o quanto acreditamos na sua negação \bar{A} . Este valor, denominado **grau de dúvida**, é expresso por

$$Dou(A) = Bel(\bar{A}) \quad (2.30)$$

No entanto, a quantidade mais utilizada para expressar esta dúvida, é aquela que traduz o quanto nós consideramos A credível ou plausível. A este valor, inicialmente denominado probabilidade superior de A , chama-se **função de plausibilidade** de A , $Pl: 2^\Theta \rightarrow [0,1]$, que é dada pela expressão

$$Pl(A) = 1 - Bel(\bar{A}) = 1 - Dou(A) \quad (2.31)$$

em que Bel é a função de crença sobre Θ .

Usando a expressão (2.29) podemos expressar Pl em função da atribuição de probabilidade básica da função de crença Bel :

$$\begin{aligned} Pl(A) &= 1 - Bel(\bar{A}) = \sum_{B \subseteq \Theta} m(B) - \sum_{B \subseteq \bar{A}} m(B) \\ &= \sum_{B \cap A \neq \emptyset} m(B) \end{aligned} \quad (2.32)$$

Comparando a expressão (2.32) com a expressão (2.29) pode concluir-se que $Bel(A) \leq Pl(A)$ para qualquer subconjunto A de Θ .

Num dado subconjunto A , a informação contida nas funções de evidência Bel e Pl pode ser representado pelo intervalo de crença $[Bel(A), Pl(A)]$. Onde $Bel(A)$ significa o suporte de evidência do subconjunto A e $Pl(A)$ traduz o quanto A é plausível. A diferença $Pl(A) - Bel(A)$ representa a denominada **ignorância residual** ou **incerteza**, aquilo que permanece ignorado. A incerteza de A representa a probabilidade básica que não é atribuída.

Na aproximação *Bayesiana* a incerteza é igual a 0 para todo o subconjunto A de Θ . No formalismo de D-S uma representação de ignorância total acerca da proposição A , desde $Bel(A) = 0$ não implica que $Dou(A) > 0$, assim como $Dou(A) = 1$ não implica que $Bel(A) = 0$.

É ainda de salientar que $Bel(A)$ é a probabilidade inferior, enquanto $Pl(A)$ é a probabilidade superior daquilo que a evidência nos permite saber. A Figura 2.13 ilustra a noção de intervalo de evidência proposto por D-S.

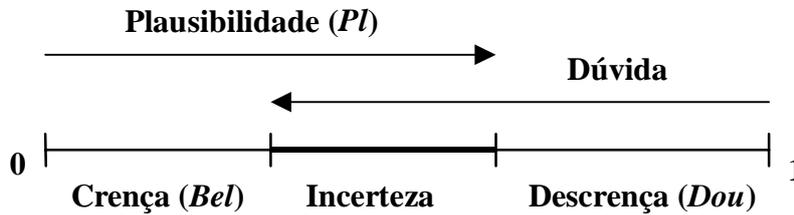


Figura 2.13 Ilustração da noção de intervalo de evidência proposto por D-S.

Regra de combinação Dempster-Shafer

Uma tomada de decisão implica uma acumulação de informação que conduza a essa decisão. Mas neste processo, o emergir de novas evidências (parciais ou não) deve levar-nos a refazer as nossas crenças à luz destes novos dados. Assim, é fundamental a qualquer método que lide com evidências, possuir capacidade para as combinar.

Supondo duas funções de crença $Bel_1: 2^{\Theta} \rightarrow [0,1]$ com os elementos focais A_1, \dots, A_k e $Bel_2: 2^{\Theta} \rightarrow [0,1]$ com os elementos focais B_1, \dots, B_k , cuja a atribuição de probabilidade básica é m_1 e m_2 respectivamente. A combinação de m_1 com m_2 será representada através de um rectângulo de área unitária, em que Bel_1 atribui tiras verticais aos seus elementos focais, enquanto Bel_2 atribui tiras horizontais.

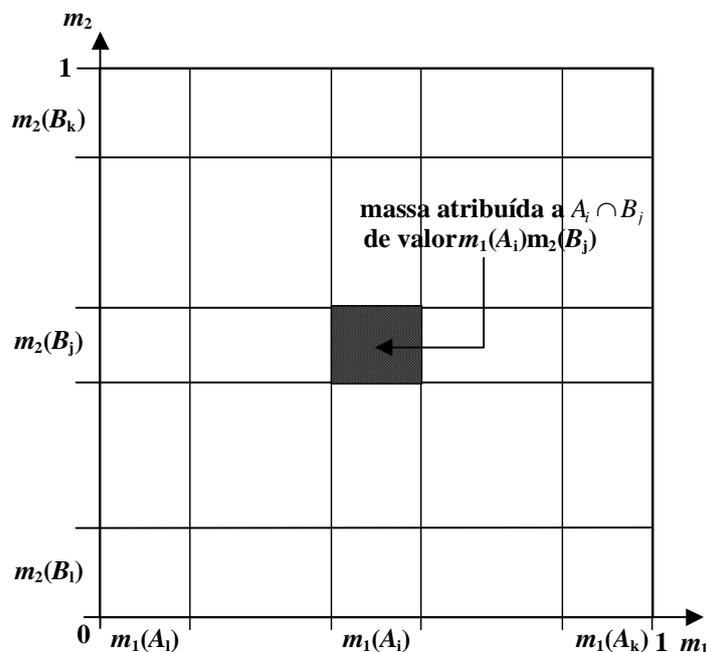


Figura 2.14 Relação entre as massas m_1 e m_2 .

Na Figura 2.14 o rectângulo assinalado tem o valor de $m_1(A_i)m_2(B_j)$ e, uma vez que corresponde simultaneamente a A_i e B_j , pode dizer-se que o efeito conjunto de Bel_1 e Bel_2 é a sua atribuição a $A_i \cap B_j$. Da mesma forma é possível especificar o valor e a atribuição para

todos os rectângulos existentes. No final, determinado o subconjunto de Θ pode ter atribuídos mais do que um destes rectângulos. A massa total que lhe será atribuída é pois

$$m(A) = \sum_{i,j,A_i \cap B_j = A} m_1(A_i)m_2(B_j). \quad (2.33)$$

É no entanto possível que alguns elementos focais A_i e B_j não se intersectem resultando um conjunto vazio \emptyset . Sendo necessário garantir que $m(\emptyset) = 0$ para que esta nova função seja ainda uma atribuição de probabilidade básica os rectângulos correspondentes devem ser eliminados. Quando assim for, um novo problema emerge: o total das áreas atribuídas não é igual a um. É então necessário afectar os valores de massa obtidos por um factor de correcção $1/N$, onde

$$N = 1 - \sum_{i,j,A_i \cap B_j = \emptyset} m_1(A_i)m_2(B_j) \quad (2.34)$$

é um factor de normalização. Assim, a função m , obtida através da combinação das funções de crença Bel_1 e Bel_2 , é uma atribuição de probabilidade básica, desde que Bel_1 e Bel_2 sejam tais que nem todos os rectângulos fiquem atribuídos ao conjunto vazio. Afectando então a expressão (2.33) do factor N obtemos a denominada **regra de combinação de Dempster**.

$$m(A) = \frac{\sum_{i,j,A_i \cap B_j = A} m_1(A_i)m_2(B_j)}{1 - \sum_{i,j,A_i \cap B_j = \emptyset} m_1(A_i)m_2(B_j)}. \quad (2.35)$$

Alguns exemplos da aplicação desta técnica são os trabalhos de Hutchinson [Hutchinson and Kak, 89], Thomas D. Garvey [Luo and Kay, 95] e Harry E. Stephanou [Luo and Kay, 95].

Factores de Confiança

A produção de regras usadas na fusão sensorial pode ser representada como uma implicação lógica da conclusão Y dada uma premissa X , expressa por

$$\text{If } X \text{ then } Y \quad \text{ou} \quad X \rightarrow Y$$

A premissa X pode ser composta por uma simples proposição ou uma conjunção, disjunção ou negação de um grupo de proposições.

A incerteza é representada num sistema que usa regras, através da associação de um **factor de confiança** (CF – *certainty factor*) a cada proposição e regra. Cada CF é uma medida da crença ou descrença, tomando valores entre -1 e 1 ($-1 \leq CF \leq 1$), onde $CF = 1$ corresponde à crença absoluta, $CF = -1$ corresponde a uma descrença total e $CF = 0$ corresponde a um peso idêntico entre crença e descrença.

A cada proposição X está associado um CF que se pode expressar como

$$X \text{ com } (CF [X])$$

onde $(CF [X])$ é inicialmente conhecido ou assume-se que é igual a 0. Dado um conjunto \mathfrak{R} de regras de um sistema, cada regra $r_i \in \mathfrak{R}$ e seu associado CF são expressos por

$$r_i : X \rightarrow Y \text{ com } (CF_i [X]).$$

O CF da premissa X em r_i pode ser definido como

$$CF_i[X] = \begin{cases} CF[X] & \text{se } X = X_1 \\ \min(CF[X_1], \dots, CF[X_n]) & \text{se } X = X_1 \wedge \dots \wedge X_n \\ \max(CF[X_1], \dots, CF[X_n]) & \text{se } X = X_1 \vee \dots \vee X_n \\ -CF[\bar{X}] & \text{senão,} \end{cases} \quad (2.36)$$

onde cada X_i é uma proposição em X . O CF da conclusão Y em r_i pode ser determinado usando

$$CF_i[Y] = \begin{cases} -CF_i[X] \times CF_i[X, Y] & \text{se ambos os } CF's < 0 \\ CF_i[Y] \times CF_i[X, Y] & \text{senão.} \end{cases} \quad (2.37)$$

O $CF_i[X, Y] = CF_i[Y]$ se r_i for evocada e $CF_i[X] = 1$.

Se existir apenas uma regra r_y para o qual uma proposição desconhecida Y é a sua conclusão, então $CF[Y] = CF_y[Y]$. Se existir mais que uma regra, então $CF[Y]$ é determinado pela fusificação de todos os $CF_i[Y]$'s das regras r_i para as quais Y é a sua conclusão. Seja

$$\mathfrak{R}_y = \{r_i : x \rightarrow y \in \mathfrak{R} \mid CF_i[x] \neq 0 \text{ e } y = Y\}$$

um conjunto de regras com premissas conhecidas e Y a sua conclusão. Dadas $N = |\mathfrak{R}_y|$ o número de regras,

$$CF[Y] = CF[Y]_{j=N},$$

onde, para todos os $r_i \in \mathfrak{R}_y$, $CF[Y]_0 = 0$ e

$$CF[Y]_j = \begin{cases} CF[Y]_{j-1} + CF_i[Y] \times (1 - CF[Y]_{j-1}) & \text{ambos } CF > 0 \\ CF[Y]_{j-1} + CF_i[Y] \times (1 + CF[Y]_{j-1}) & \text{ambos } CF < 0 \\ \frac{CF[Y]_j + CF_i[Y]}{1 - \min(CF[Y]_{j-1}, CF_i[Y])} & \text{senão} \end{cases} \quad (2.38)$$

para $j = 1$ até N .

Um exemplo da aplicação de factores de confiança é o trabalho de Luo em [Luo et al, 88].

Lógica Fuzzy

A lógica *Fuzzy* foi introduzida pelo Dr. Lotfi Zadeh em meados dos anos 60 como meio de modelar as incertezas da linguagem natural. Segundo Zadeh é muito frequente encontrar classes de objectos no mundo real sem contornos bem definidos. Por exemplo a classe dos animais inclui claramente cães, cavalos e pássaros como seus membros, e exclui decisivamente móveis, líquidos e plantas. Contudo, seres vivos como a estrela-do-mar e a bactéria encontram-se num estado ambíguo respeitante à classe dos animais [Zadeh, 65]. A mesma espécie de ambiguidade pode-se encontrar na definição da “classe das mulheres

bonitas” ou na “classe dos homens altos”. Este tipo de classes não se consegue caracterizar através do senso matemático usual. Podem no entanto ser caracterizadas através da lógica *fuzzy* uma vez que esta permite gerir o conceito de parcialmente verdade (com um número “infinito” de níveis de verdade) – **lógica multi-valor**, enquanto que os sistemas convencionais apenas gerem dois níveis de verdade – **lógica bivalor**, ou três níveis de verdade – **lógica trivalor**.

Conceitos básicos e terminologia

Considere-se X como um universo de objectos e x como um elemento genérico de X e um conjunto clássico A , em que $A \subset X$, é definido como uma colecção de elementos ou objectos $x \in X$, em que cada x pode pertencer ou não a A .

Para definir a **função característica** de cada elemento de x em X , pode-se representar o conjunto clássico A por um conjunto de pares ordenados do tipo $(x, 1)$ e $(x, 0)$, os quais indicam respectivamente se x pertence ou não pertence ao conjunto A . Em contrapartida um conjunto *fuzzy* indica o grau com que um elemento pertence ao conjunto. Consequentemente, a função característica de um conjunto *fuzzy* toma valores entre 0 e 1, os quais determinam o grau de pertença de um elemento num dado conjunto.

A Figura 2.15 apresenta a função característica de $A =$ “Conjunto dos homens altos” para um conjunto clássico e um conjunto *fuzzy*.

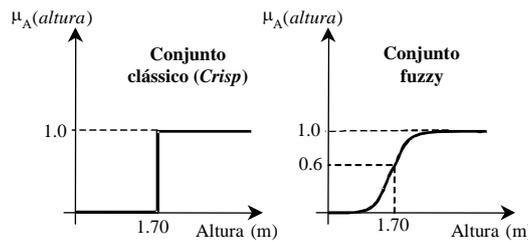


Figura 2.15 Representação de A por um conjunto clássico e por um conjunto *fuzzy*.

Definição 2.1 Conjuntos *Fuzzy* e funções pertença (MF – *Membership function*).

Se X é uma colecção de objectos designados genericamente por x , então o conjunto *fuzzy* A de X é definido como um conjunto de pares ordenados:

$$A = \{(x, \mu_A(x)) \mid x \in X\} \tag{2.39}$$

onde $\mu_A(x)$ é a função pertença do conjunto *fuzzy* A . Esta função atribui um grau de pertença entre 0 e 1 a cada elemento de X . Este é denominado de **universo do discurso**, podendo ser discreto ou contínuo (Figura 2.16).

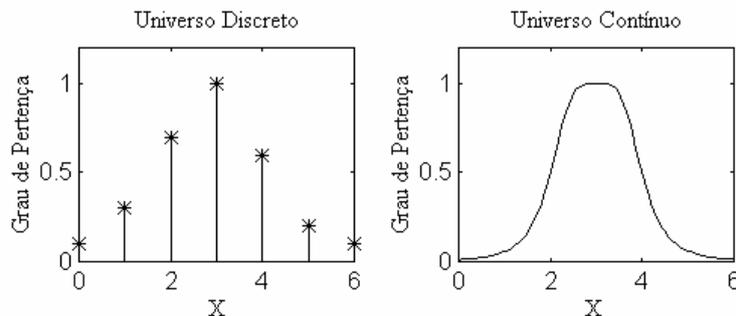


Figura 2.16 Representação de um conjunto *fuzzy* num universo discreto e num universo contínuo.

Um conjunto *fuzzy* é completamente caracterizado pela sua função de pertença, sendo contudo impraticável listar todos os pares que definem essa mesma função. Devido a esta dificuldade é frequente a utilização de expressões matemáticas na definição das funções de pertença. Dentro destas as mais comuns são as funções triangulares, trapezoidais, gaussianas e *bell*.

A função de pertença **triangular** é especificada por 3 parâmetros {a, b, c}:

$$\text{Triângulo}(x; a, b, c) = \begin{cases} 0 & x \leq a. \\ \frac{x-a}{b-a} & a \leq x \leq b. \\ \frac{c-x}{c-b} & b \leq x \leq c \\ 0 & c \leq x \end{cases} \quad (2.40)$$

A função de pertença **trapezoidal** é especificada por 4 parâmetros {a, b, c, d}:

$$\text{Trapezóide}(x; a, b, c, d) = \begin{cases} 0 & x \leq a. \\ \frac{x-a}{b-a} & a \leq x \leq b. \\ 1 & b \leq x \leq c. \\ \frac{d-x}{d-c} & c \leq x \leq d. \\ 0 & d \leq x. \end{cases} \quad (2.41)$$

A função de pertença **Gaussiana** é especificada por 2 parâmetros {c, σ }:

$$\text{Gaussiana}(x; c, \sigma) = e^{-\frac{1}{2} \left(\frac{x-c}{\sigma} \right)^2} \quad (2.42)$$

A função **Bell** é especificada por 3 parâmetros {a, b, c}:

$$\text{bell}(x; a, b, c) = \frac{1}{1 + \left| \frac{x-c}{a} \right|^{2b}} \quad (2.43)$$

A Figura 2.17 ilustra os quatro tipos de funções pertença referidos [Jang et al, 97].

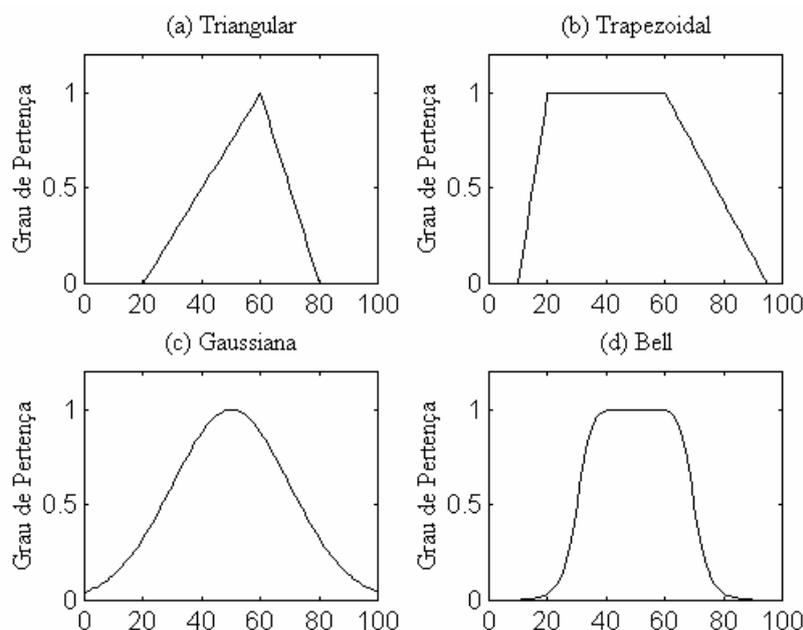


Figura 2.17 Exemplo dos quatro tipos de funções pertença mais comuns. (a) $Triângulo(x; 20, 60, 80)$; (b) $Trapezoidal(x; 10, 20, 60, 95)$; (c) $Gaussiana(x; 50, 20)$; (d) $Bell(x; 20, 4, 50)$.

Definição 2.2 Suporte.

O **suporte** de um conjunto *fuzzy* A é o conjunto de todos os pontos $x \in X$ para os quais $\mu_A(x) > 0$:

$$Suporte(A) = \{x \mid \mu_A(x) > 0\} \tag{2.44}$$

Definição 2.3 Núcleo.

O **núcleo** de um conjunto *fuzzy* A é o conjunto de todos os pontos $x \in X$ para os quais $\mu_A(x) = 1$:

$$Núcleo(A) = \{x \mid \mu_A(x) = 1\} \tag{2.45}$$

Definição 2.4 Normalidade.

Um conjunto *fuzzy* A é **normal** se no seu núcleo existir pelo menos um ponto $x \in X$ para o qual $\mu_A(x) = 1$.

Definição 2.5 Pontos de *crossover*.

Um ponto de *crossover* de um conjunto *fuzzy* A é o ponto $x \in X$ para o qual $\mu_A(x) = 0.5$:

$$crossover(A) = \{x \mid \mu_A(x) = 0.5\} \tag{2.46}$$

A Figura 2.18 ilustra o núcleo, os pontos de *crossover* e o suporte de um conjunto *fuzzy* para uma função de pertença do tipo *bell*.

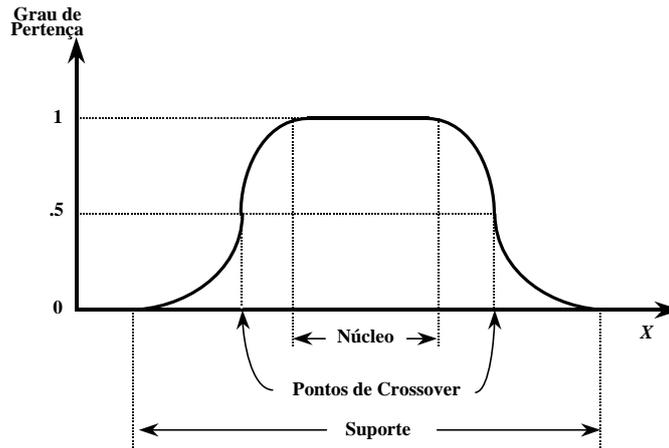


Figura 2.18 Núcleo, pontos de *crossover* e suporte de um conjunto *fuzzy*.

Definição 2.6 α -cut..

O α -cut de um conjunto *fuzzy* A é um conjunto *crisp* definido por

$$A_\alpha = \{x \mid \mu_A(x) \geq \alpha\} \tag{2.47}$$

Definição 2.7 Variáveis linguísticas e valores linguísticos.

As **variáveis linguísticas** são expressões que caracterizam grandezas físicas e, cujo valor é especificado por proposições da linguagem natural ou artificial em detrimento de um valor numérico. Expressões como “jovem” e “muito-aberta” são designadas como **valores linguísticos** respectivamente das grandezas físicas “idade” e “ Abertura de uma válvula de aquecimento”.

Um valor linguístico é um conjunto *fuzzy*. Todos os valores linguísticos especificam uma variável linguística. Considerando $X = \text{“Idade”}$. Então é possível definir os conjuntos *fuzzy* “Muito Jovem”, “Jovem”, “Velho” e “Muito Velho”, caracterizados por as funções de pertinência $\mu_{\text{Muito-Jovem}}(x)$, $\mu_{\text{Jovem}}(x)$, $\mu_{\text{Velho}}(x)$ e $\mu_{\text{Muito-velho}}(x)$. A Figura 2.19 apresenta as funções de pertinência que caracterizam a variável linguística “Idade” e os valores linguísticos que a especificam.

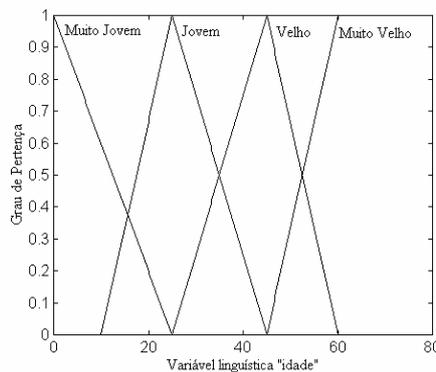


Figura 2.19 Funções de pertinência que caracterizam a variável linguística “Idade” e os valores linguísticos que a especificam.

Operações com conjuntos Fuzzy

As operações mais básicas sobre conjuntos clássicos são a união, a intersecção e o complemento. De forma correspondente a estas operações, os conjuntos *fuzzy* tem operações

similares, as quais foram inicialmente definidas por Zadeh [Zadeh, 65]. Contudo, antes da definição dos três operadores referidos é necessário definir a noção de subconjunto, uma vez que esta é uma função a ter em consideração quer nas operações com conjuntos clássicos quer com os conjuntos *fuzzy*. Esta definição, como não poderia deixar de acontecer, é uma extensão natural do caso dos conjuntos clássicos.

Definição 2.8 Subconjunto.

O conjunto *fuzzy* A está **contido** no conjunto *fuzzy* B (ou A é um subconjunto de B) se e somente se $\mu_A(x) \leq \mu_B(x)$ para todo e qualquer x .

$$A \subseteq B \Leftrightarrow \mu_A(x) \leq \mu_B(x) \tag{2.48}$$

Definição 2.9 União.

A união de dois conjuntos *fuzzy* A e B é um conjunto *fuzzy* C , expresso por $C = A \cup B$ ou $C = A$ OR B , onde a função de pertinência relaciona A e B por

$$\mu_C(x) = \max(\mu_A(x), \mu_B(x)) = \mu_A(x) \vee \mu_B(x) \tag{2.49}$$

Definição 2.10 Intersecção.

A intersecção de dois conjuntos *fuzzy* A e B é um conjunto *fuzzy* C , expresso por $C = A \cap B$ ou $C = A$ AND B , onde a função de pertinência relaciona A e B por

$$\mu_C(x) = \min(\mu_A(x), \mu_B(x)) = \mu_A(x) \wedge \mu_B(x) \tag{2.50}$$

Definição 2.11 Complemento.

O complemento de um conjunto *fuzzy* A , designado por \bar{A} , é expresso por

$$\mu_{\bar{A}}(x) = 1 - \mu_A(x) \tag{2.51}$$

A Figura 2.20 demonstra estas três operações básicas [Jang et al, 97].

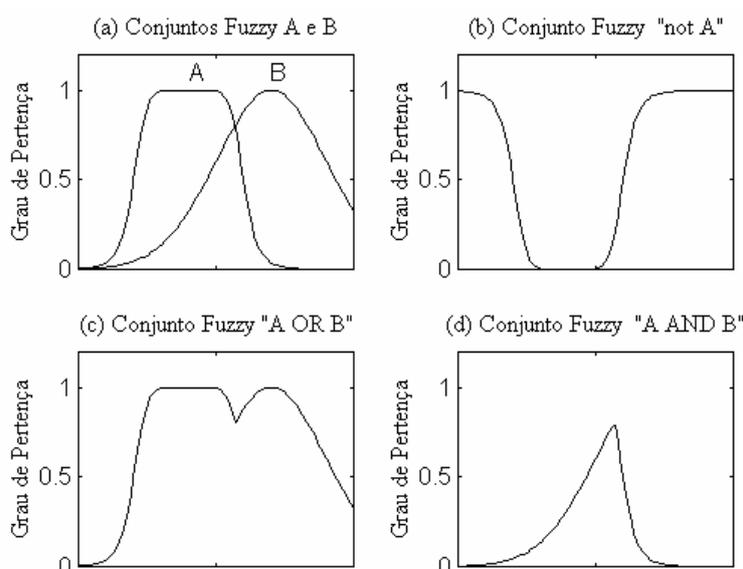


Figura 2.20 Operações com conjuntos *Fuzzy*: (a) dois conjuntos *fuzzy*; (b) \bar{A} ; (c) $A \cup B$; (d) $A \cap B$.

Normalmente a lógica *fuzzy* usa o operador clássico de complemento, mas os operadores de intersecção e união podem ser facilmente configurados se o pretendermos. A intersecção de dois conjuntos *fuzzy* A e B é geralmente especificada pela função $T : [0,1] \times [0,1] \rightarrow [0,1]$, a qual faz a agregação de dois graus de pertença da seguinte forma:

$$\mu_{A \cap B}(x) = T(\mu_A(x), \mu_B(x)) = \mu_A(x) \otimes \mu_B(x) \tag{2.52}$$

onde \otimes é um operador binário para a função T . Esta classe de operadores de intersecção *fuzzy* são usualmente referidos como *T-norm* (*triangular norm*).

Definição 2.12 *T-norm*.

Um operador *T-norm* é uma função de dois termos que apresente as seguintes propriedades:

$T(0, 0) = 0,$	$T(a, 1) = T(1, a) = a$	Limitada
$T(a, b) \leq T(c, d)$	se $a \leq c$ e $b \leq d$	Monotonicidade
$T(a, b) = T(b, a)$		Comutativa
$T(a, T(b, c)) = T(T(a, b), c)$		Associativa

A primeira propriedade impõe uma correcta generalização para os conjuntos clássicos. A segunda propriedade implica que uma decrementação nos valores de pertença de A ou B não origine uma incrementação no valor de pertença de $A \cap B$. A terceira propriedade indica que o operador é indiferente à ordem dos conjuntos *fuzzy* a serem combinados. Finalmente, a quarta propriedade permite-nos efectuar a intersecção de um número qualquer de conjuntos *fuzzy* sem ter em consideração a orientação do agrupamento. Os quatro operadores *T-norm* mais frequentemente usados são:

Mínimo	$T_{\min}(a, b) = \min(a, b) = a \wedge b.$
Produto Algébrico	$T_{\text{pa}}(a, b) = ab.$
Produto Limitado (<i>Bounded product</i>)	$T_{\text{pl}}(a, b) = 0 \vee (a + b - 1).$
Produto Drástico (<i>Drastic product</i>)	$T_{\text{pd}}(a, b) = \begin{cases} a, & \text{se } b = 1 \\ b, & \text{se } a = 1 \\ 0, & \text{se } a, b < 1 \end{cases}$

A Figura 2.21 apresenta as superfícies correspondentes aos 4 operadores *T-norm* [Jang et al, 97].

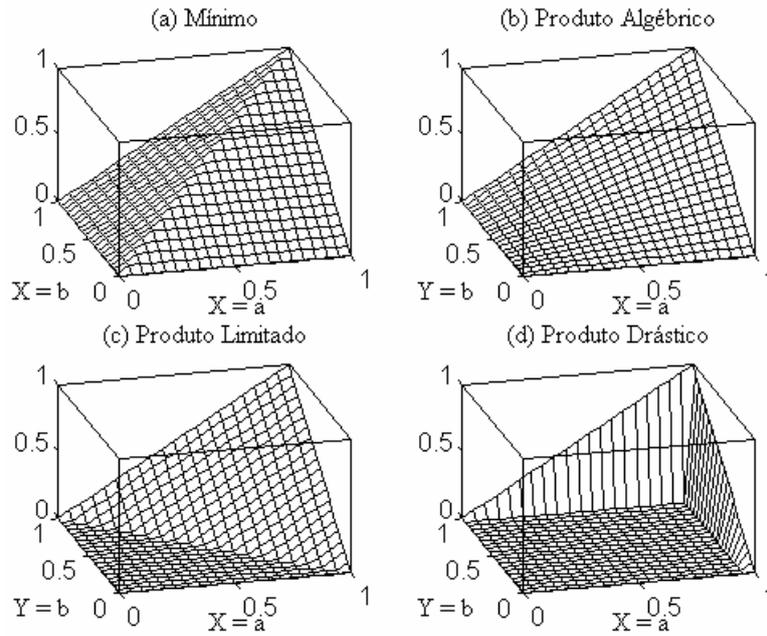


Figura 2.21 Os 4 operadores *T-norm*. (a) $T_{\min}(a, b)$; (b) $T_{pa}(a, b)$; (c) $T_{pl}(a, b)$; (d) $T_{pd}(a, b)$.

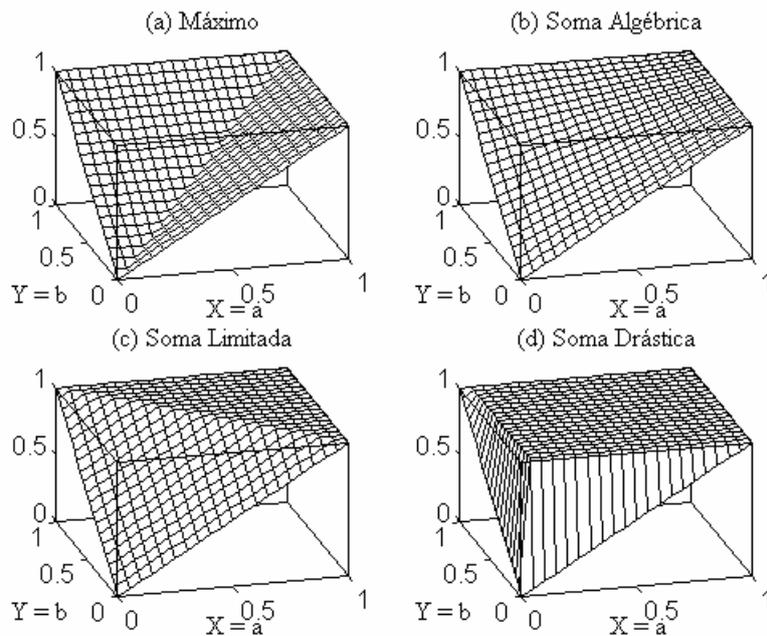


Figura 2.22 Os 4 operadores *T-conorm*. (a) $S_{\max}(a, b)$; (b) $T_{sa}(a, b)$; (c) $T_{sl}(a, b)$; (d) $T_{sd}(a, b)$.

Da mesma forma que o operador de intersecção *fuzzy*, o operador de união *fuzzy* é especificado pela função $S : [0,1] \times [0,1] \rightarrow [0,1]$, a qual faz a agregação de dois graus de pertinência da seguinte forma:

$$\mu_{A \cup B}(x) = S(\mu_A(x), \mu_B(x)) = \mu_A(x) \oplus \mu_B(x) \tag{2.53}$$

onde \oplus é um operador binário para a função S . Esta classe de operadores de união *fuzzy* são usualmente referidos como *T-conorm* (ou *S-norm*).

Definição 2.13 *T-conorm* (ou *S-norm*).

Um operador *T-conorm* é uma função de dois termos que apresente as seguintes propriedades:

$S(1, 1) = 1,$	$S(0, a) = S(a, 0) = a$	Limitada
$S(a, b) \leq S(c, d)$	se $a \leq c$ e $b \leq d$	Monotonicidade
$S(a, b) = S(b, a)$		Comutativa
$S(a, S(b, c)) = S(S(a, b), c)$		Associativa

A justificação é idêntica à do *T-norm*. Os quatro operadores *T-conorm* mais frequentemente utilizados são:

Máximo	$S_{\max}(a, b) = \max(a, b) = a \vee b.$
Soma Algébrica	$S_{sa}(a, b) = a + b - ab.$
Soma Limitada (Bounded sum)	$S_{sl}(a, b) = 1 \vee (a + b).$
Soma Drástica (Drastic sum)	$S_{sd}(a, b) = \begin{cases} a, & \text{se } b = 0 \\ b, & \text{se } a = 0 \\ 1, & \text{se } a, b > 0 \end{cases}$

A Figura 2.22 apresenta as superfícies correspondentes aos 4 operadores *T-conorm* [Jang et al, 97].

Sistema de inferência fuzzy

Um sistema de inferência *fuzzy* é um popular quadro de trabalho baseado na teoria dos conjuntos *fuzzy*. Este sistema consiste em três componentes conceptuais (Figura 2.23): uma **interface de fuzificação**, uma **base de regras** e uma **interface de desfuzificação**.

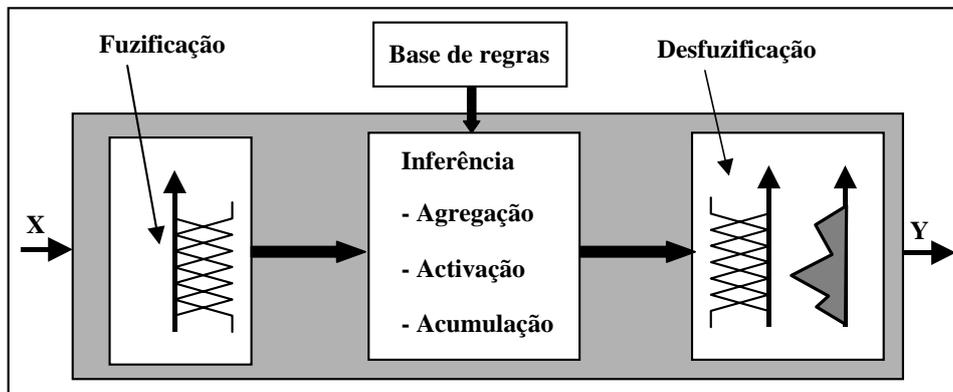


Figura 2.23 Estrutura e elementos funcionais de um sistema de inferência *fuzzy*.

Denomina-se de método de **fuzificação** à determinação da relação entre as variáveis de entrada e os valores linguísticos. O grau de pertença da variável de entrada é determinado para cada valor linguístico da correspondente variável linguística (Figura 2.24).

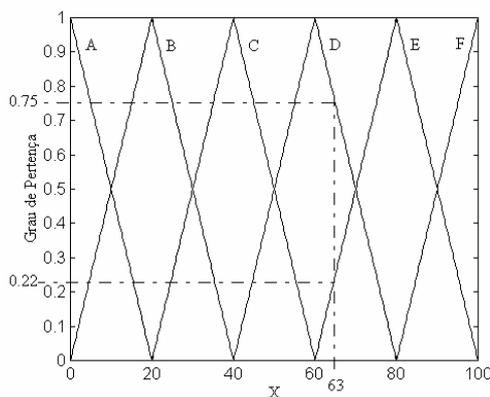


Figura 2.24 Princípio da fusificação.

É ainda de salientar que o valor de entrada pode pertencer a mais que um conjunto *fuzzy*, o que não era possível na teoria dos conjuntos clássica.

A **base de regras** contém o conhecimento empírico respeitante à operação de um processo particular em consideração. O conhecimento é representado por regras linguísticas. A definição textual de uma base de regras é efectuada da seguinte forma

- R1: IF Condição P₁ THEN Conclusão C₁
- R2: IF Condição P₂ THEN Conclusão C₂
- R3: IF Condição P₃ THEN Conclusão C₃
- ...
- R_n: IF Condição P_n THEN Conclusão C_n

onde, a condição pode consistir numa combinação de estados linguísticos, podendo ter mais que uma conclusão.

$$\text{IF } P_k = P_{k1} \text{ AND } P_{k2} \text{ OR } (\text{NOT } P_{k3}) \text{ THEN } C_{k1}, C_{k2}$$

O processo de **inferência** consiste em três subfunções **agregação**, **activação** e **acumulação**, correspondendo aos passos de execução da base de regras.

• **Agregação**

A subfunção de agregação consiste num processo de validação da condição existente em cada regra que constitui a base de dados. Se a condição é simples então o grau de pertinência obtido é igual ao da própria condição. Contudo, se a condição consiste numa combinação de subcondições, o grau de pertinência resultante terá que ser determinado por agregação dos valores individuais. Se a combinação das subcondições for efectuada por um AND o grau de pertinência resultante será determinado por aplicação do operador de lógica *fuzzy* AND. No caso de a combinação ser efectuada por um OR será aplicado o operador de lógica *fuzzy* OR.

Exemplo 2.1

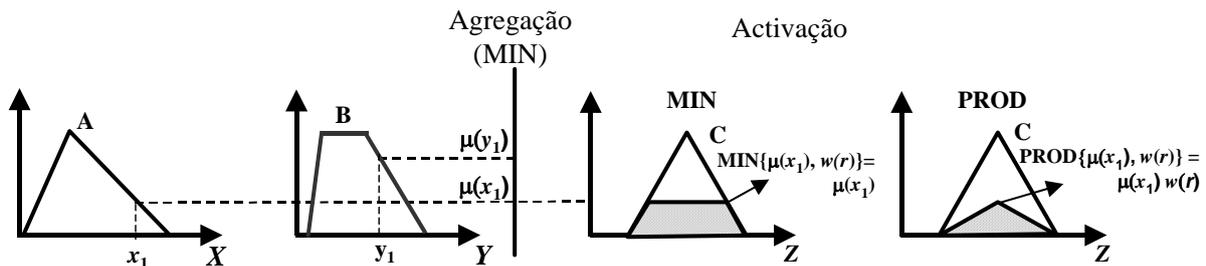
$$\begin{array}{c}
 P_k = P_{k1} \text{ AND } P_{k2} \text{ OR } (\text{NOT } P_{k3}) \\
 \underbrace{\hspace{10em}} \\
 \text{MIN}\{\mu_{k1}, \mu_{k2}\} \quad \text{OR} \quad (1 - \mu_{k3}) \\
 \downarrow \\
 \underbrace{\hspace{10em}} \\
 \mu_k = \text{MAX}[\text{MIN}\{\mu_{k1}, \mu_{k2}\}, (1 - \mu_{k3})]
 \end{array}$$

• **Activação**

Na subfunção de activação o grau de pertença da conclusão é determinado com base no grau de pertença das condições resultante do processo de agregação e na quantidade de crença (valor entre 0 e 1 que indica a credibilidade ou confiança na regra) atribuída à regra. Geralmente na activação são usados os operadores MIN ou PROD. Com a aplicação do operador MIN o conjunto *fuzzy* de saída é truncado enquanto com o operador PROD o conjunto *fuzzy* de saída é “escamado”.

Exemplo 2.2

IF $x = A$ AND $y = B$ THEN $z = C$ ($w(r) = 1$)

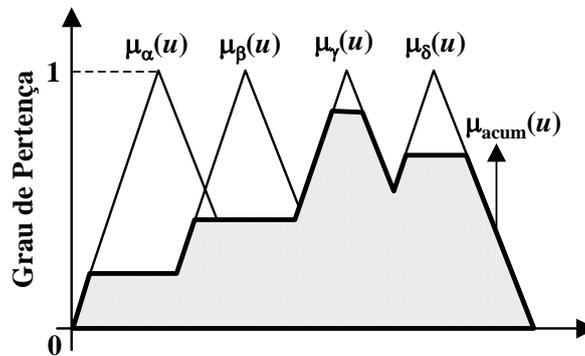


• **Acumulação**

Na subfunção de acumulação os resultados das regras são combinados para obter um resultado global. O operador MAX é normalmente usado na acumulação.

Exemplo 2.3

Utilização do operador MAX.



Dependendo da combinação de operadores nos passos individuais, estratégias diferentes de inferência podem ser efectuadas. As mais conhecidas são as denominadas **MAX-MIN** e **MAX-PROD**, as quais usam o operador MAX na acumulação e o operador MIN ou PROD na activação.

A Tabela 2-1 apresenta um resumo dos operadores mais usados no processo de inferência

Passos da inferência	Operadores	Algoritmos
Agregação		
AND	Mínimo	$\mu_{agr} = \text{MIN}\{\mu_{k1}(x), \mu_{k2}(x)\}$
OR	Máximo	$\mu_{agr} = \text{MAX}\{\mu_{k1}(x), \mu_{k2}(x)\}$
Activação		
Conversão da conclusão IF-THEN		
	Mínimo	$\mu_{act} = \text{MIN}\{\mu_{agr}, w_k\}$
Peso de cada regra		
	Multiplicação	$\mu_{act} = \text{PROD}\{\mu_{agr}, w_k\}$
Acumulação	Máximo	$\mu_{accu}(i) = \text{MAX}\{\mu_{act}(i)\}$

Tabela 2-1 Passos do processo de inferência e operadores mais usuais.

O processo de **desfuzificação** corresponde à extracção de um valor numérico representativo, normalmente real, a partir de um conjunto *fuzzy*. Muitos métodos foram e são propostos na literatura, não existindo contudo uma base científica para qualquer um deles, isto é, nenhum método derivou de um primeiro princípio como a maximização da informação *fuzzy* ou entropia; conseqüentemente, a desfuzificação é uma arte relativa da ciência. Em virtude do interesse em aplicações de lógica *fuzzy* em engenharia, um critério para a escolha do método de desfuzificação é a sua simplicidade computacional. Este critério levou a que fossem utilizados 6 métodos principais de desfuzificação: **Centro de Gravidade** (*Centre of Gravity CoG*) ou **Área do Centróide** (*Centre of Area CoA*), **Máximo Mais à Esquerda** (*Left Most Maximum LF*), **Máximo Mais à Direita** (*Right Most Maximum RM*), **Média dos Máximos** (*Mean of Maximum MoM*), **Centro de Área** (*Centre of Area CoA*) ou **Bisector de Área** (*Bisector of Area BoA*) e **Centro dos Máximos** (*Center of Maximum CoM*) ou **Centro de Gravidade para Singletons** (*Center of Gravity for Singleton COGS*).

- Método do **Centro de Gravidade** ou **Área do Centróide** – A variável de saída é determinada como o valor da abcissa u correspondente ao centro de gravidade debaixo da função de pertença (Figura 2.25).

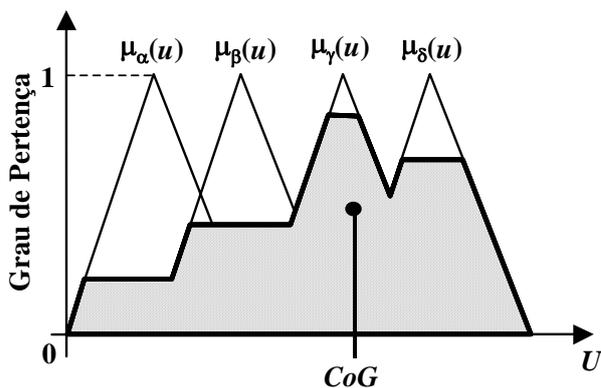


Figura 2.25 Método CoG.

$$CoG = \frac{\int_U u \times \mu_A(u) \times du}{\int_U \mu_A(u) \times du}$$

- Método **Máximo Mais à Esquerda** – O valor da variável de saída é o valor da abcissa correspondente ao valor máximo situado mais à esquerda da função de pertinência

de saída (Figura 2.26).

Método do **Máximo Mais à Direita** – O valor da variável de saída é o valor da abcissa correspondente ao valor máximo situado mais à direita da função de pertinência (Figura 2.26).

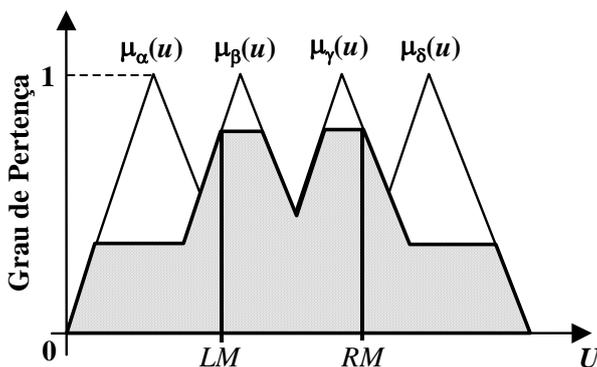


Figura 2.26 Método LM e RM.

- Método da **Média dos Máximos** – O valor da variável de saída é o valor da abcissa correspondente à média dos valores máximos (Figura 2.27).

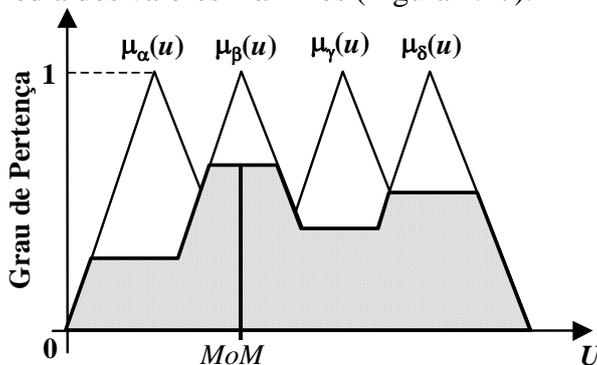
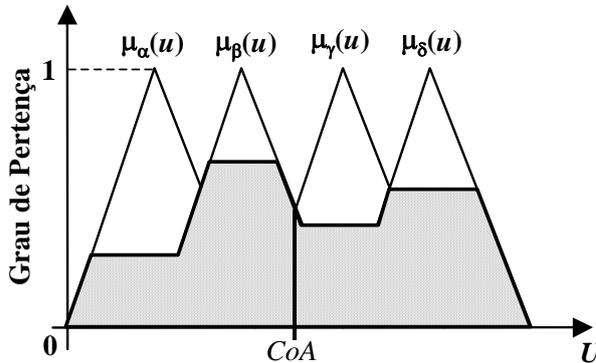


Figura 2.27 Método MoM.

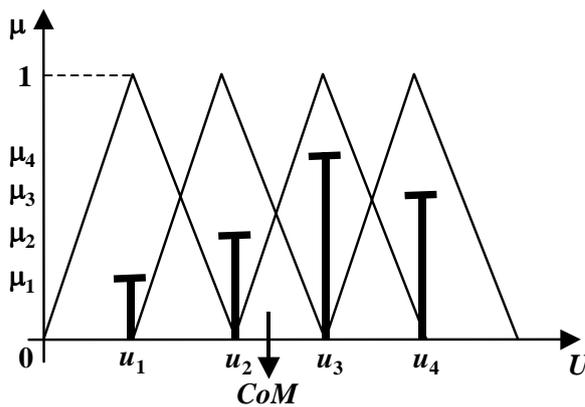
- Método do **Centro de Área** ou **Bisector de Área** – O valor da variável de saída é o valor da abcissa para o qual a área à sua esquerda debaixo da função de pertença é igual à área à sua direita.



$$\int_{Min}^X \mu_A(u)du = \int_X^{Max} \mu_A(u)du$$

Figura 2.28 Método CoA.

- Método do **Centro dos Máximos** ou Centro de Gravidade para *Singletons* – Este método é uma simplificação do método CoG uma vez que não toma em consideração a área mas sim o valor máximo da respectiva função de pertença activada.



$$CoM = \frac{\sum_{i=1}^{\gamma} u_i \times \mu_i(u_i)}{\sum_{i=1}^{\gamma} \mu_i(u_i)}$$

Figura 2.29 Método CoM.

Exemplos da aplicação de lógica *fuzzy* em fusão sensorial são os trabalhos de Russo [Russo and Ramponi, 94] e Roth [Roth and Schilling, 96].

2.4 Sumário

Neste capítulo foram inicialmente apresentados os conceitos de fusão e integração sensorial. Em seguida, foram abordados os fundamentos da fusão sensorial, sendo feita a analogia com o processo de fusão efectuado pelo homem. Foi ainda referido que este conceito envolve poucas ideias essenciais: múltiplas entradas, uma ou mais técnicas de fusão, o resultado desejado ou inferência e um mecanismo de controlo do sistema.

Na segunda parte foram apresentadas as técnicas mais frequentes em fusão sensorial divididos por três grandes grupos: métodos ao nível do sinal, métodos ao nível do ponto e métodos ao nível do símbolo. Nestes, foi dado realce às técnicas do filtro de *Kalman* e da lógica *fuzzy*, em virtude de serem estas juntamente com alguma heurística que serviram de base à implementação do processo de fusão sensorial utilizado neste trabalho.

Capítulo 3

3. Arquitecturas de controlo para sistemas multi-sensoriais

3.1 Introdução

Um sistema multi-sensorial é constituído por dois subsistemas: o sistema de sensação e o sistema de percepção. O sistema de sensação é constituído por um bloco de comunicação que interliga um conjunto de sensores possibilitando a transmissão de dados adquiridos entre os diferentes nós de processamento do sistema. Por seu lado, o sistema de percepção é entendido como o processo de estabelecimento e manutenção da relação entre o modelo do ambiente e o ambiente real. A sua função é a extracção de informação sobre entidades, estados, e relações existentes no ambiente real, com o objectivo de manter o modelo actualizado. Deste modo, o sistema de percepção comporta processos de fusão sensorial, de detecção e de reconhecimento, e em sistemas com inteligência superior suporta ainda aprendizagem e “raciocínio” para obtenção de novos conhecimentos [Nunes, 95].

Tendo em consideração o anteriormente referido, pode-se considerar um sistema de controlo multi-sensorial genericamente constituído pelos seguintes elementos: sensores, sistema de integração sensorial e controlo, base de conhecimento e actuadores. Com base nestes elementos, surgiram diversas aproximações para arquitecturas de controlo de sistemas multi-sensoriais. De uma forma genérica é possível dividir estas aproximações em três grandes grupos: aproximações cognitivas, comportamentais e híbridas. Em qualquer destas arquitecturas sobressaem as funções de sensação, percepção, planeamento e controlo de execução de acções, ainda que sob diferentes formas organizativas.

Após o aparecimento da Inteligência Artificial (IA) em meados dos anos 50, a resolução de problemas (em particular a geração de planos de acção) ocupam um lugar privilegiado. Os algoritmos propostos neste domínio pela IA denominada de cognitivista (também denominada de funcional e simbólica), demonstram as capacidades de “raciocínio” global baseando-se em representações simbólicas que transformam uma máquina programada numa máquina “inteligente” aos olhos de um observador. Esta transformação inteligente é referida com frequência como “computação simbólica” para que se possa distinguir do processamento de dados [Malcolm and Smithers, 90]. É uma estratégia assente no planeamento, que utiliza uma estrutura centralizada para a representação do modelo do mundo, capaz de gerar acções a partir da informação proveniente de sensores. A informação do modelo do mundo é usada no planeamento com o objectivo de produzir sequências apropriadas de acções. Contudo e

embora tenham sido realizados numerosos trabalhos neste domínio, estes sistemas apresentam fortes limitações (tempos de resposta proibitivos, falta de robustez sempre que aparecem incertezas e falta de flexibilidade face ao imprevisto) o que torna impossível a sua utilização em sistemas de controlo evoluídos, ambientes complexos e condições reais. Pode dizer-se que estes sistemas estão desligados do mundo real. São essencialmente utilizados em ambientes de simulação, jogos ou num mundo extremamente simplificado que facilitem uma descrição simbólica [Rodriguez, 94].

Face às limitações da aproximação cognitiva, os anos 80 viram surgir a aproximação dita de comportamental ou reactiva. Esta aproximação associa autonomia à capacidade de evoluir num universo real (dinâmico, incerto, parcialmente desconhecido, talvez hostil). A aproximação comportamental revolucionou o mundo da IA porque contém as condições limites para a geração de planos de acção. Ela introduz as novas arquitecturas de controlo ligadas à percepção da acção, o que permite em casos extremos, agir por raciocínio local (simples, robusto e rápido) baseado em representações não simbólicas (próximas das fontes de sinal fornecidas pelos sensores e dos comandos enviados aos actuadores). Contudo, o que fez a originalidade desta aproximação - o abandono das representações simbólicas globais permitindo um raciocínio sobre tarefas a longo prazo, constitui também a sua maior limitação, uma vez que grande parte do tempo é dispendido a reagir ao ambiente dando origem a grandes demoras na execução da tarefa pretendida

O termo híbrido é usado aqui com o objectivo de distinguir as aproximações em que, conjuntamente, se consideram interligadas a componente cognitiva e comportamental. Uma arquitectura híbrida oferece um compromisso entre uma aproximação puramente reactiva e uma aproximação baseada no planeamento, através da utilização de um sistema reactivo para controlo de baixo nível e um sistema de planeamento para decisões de mais alto nível. Em resumo, um sistema híbrido oferece um compromisso entre os extremos “pensar” e “reagir”. Esta aproximação separa o sistema de controlo em duas ou mais partes basicamente independentes mas comunicantes. Em muitos casos, processos reactivos de baixo nível tomam em consideração a segurança imediata da entidade (na robótica - o robô), enquanto que os de alto nível usam o planeamento para seleccionar as sequências de acções [Mataric, 92a] [Mataric, 98].

Em meados dos anos 70 Barbera propôs uma arquitectura alternativa para sistemas inteligentes mais abrangente que permite a simultaneidade de estruturas hierárquicas e horizontais. Esta arquitectura foi denominada de *real-time control system* (RCS). Posteriormente viria a ser adoptada por Albus e outros para controlo de manufactura no *National Institute of Standards and Technology* (NIST). O RCS consiste num conjunto de camadas dispostas hierarquicamente, constituídas por nós com processamento inteligente. Em cada nó, tarefas são decompostas, planos são gerados, modelos do mundo são mantidos, dados provenientes de sensores processados, e ciclos de controlo são fechados. Os nós de nível mais elevado realizam tarefas de gestão e planeamento de alto nível, enquanto os nós de baixo nível realizam tarefas de coordenação de mecanismos e controlo de processos [Albus, 96b]. O resultado é um sistema que combina e distribui características reactivas e de planeamento em toda a arquitectura hierárquica, com capacidades discretas e contínuas fortemente integradas em todos os níveis. Cada nó da arquitectura RCS pode implementar um sistema de controlo híbrido e cada função pertencente aos nós pode ser implementada por uma máquina de estados finita [Albus and Proctor, 96].

Este capítulo começa com a apresentação das arquitecturas cognitivista, comportamental e híbrida, prosseguindo depois com a arquitectura RCS. Na terceira e ultima parte é discutida a arquitectura adoptada para este trabalho.

3.2 Aproximação cognitivista, comportamental e híbrida

3.2.1 Aproximação cognitivista

A aproximação cognitivista insere-se no quadro da IA simbólica. Ela é essencialmente suportada por uma forte hipótese, denominada “hipótese simbólica”, que se pode exprimir da seguinte forma [Rodríguez, 94]:

H1 : A inteligência pode ser descrita como uma sucessão de operações sobre estruturas simbólicas, interpretáveis proporcionalmente.

Esta posição audaciosa pretende que os teoremas da lógica formal sejam as leis do pensamento, tendo sido formulada e adoptada num contexto de expansão tecnológica. Por conseguinte, de um ponto de vista arquitectural, a aproximação cognitivista é sustentada por uma segunda hipótese:

H2 : A inteligência é separável do seu suporte físico.

Esta hipótese supõe a existência de um nível de abstracção em que os processos mentais podem ser considerados independentes dos mecanismos que os originaram. Esta metáfora computacional (H2) permitiu aos fundadores da IA extrair totalmente os mecanismos de inteligência do suporte físico, conduzindo à proposta de uma arquitectura funcional vertical para um sistema inteligente composto por 4 módulos (Figura 3.1): percepção, modelação, planeamento e acção.

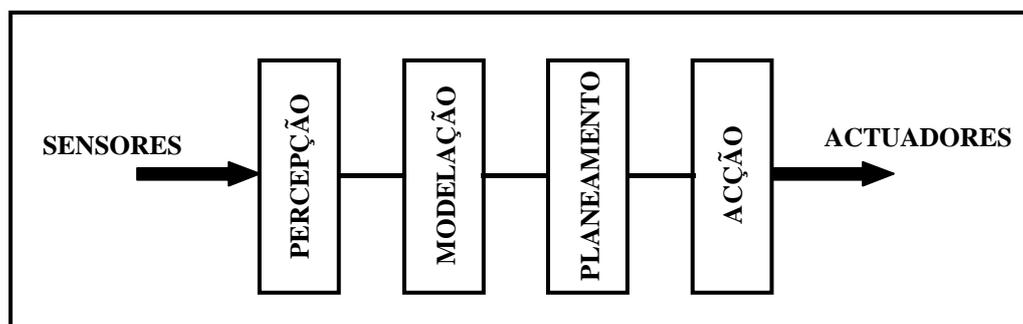


Figura 3.1 Arquitectura funcional proposta pela aproximação cognitivista.

O módulo de percepção, tem como função adquirir informação acerca do meio ambiente a partir dos sensores fornecendo ao módulo de modelação a descrição simbólica dessa realidade. A modelação tem como função a representação interna do conhecimento. O módulo de planeamento a partir da representação interna tenta estabelecer um conjunto de acções que permita ao sistema realizar a sua tarefa. Finalmente, o módulo de acção trata sequencialmente o conjunto de acções determinados pelo módulo de planeamento, controlando os actuadores do sistema de acordo com a tarefa a executar.

Nesta arquitectura, a informação é transmitida sequencialmente de módulo em módulo desde o princípio até ao fim. O modelo cognitivo é assim por natureza de construção descendente, consistindo a sua base de partida na procura de princípios generalistas. A

procura destes princípios é intrinsecamente um processo difícil cujo o objectivo é dar uma forma concreta a conceitos com elevado nível de abstracção. Por este motivo, a investigação relacionada com este modelo tem-se centrado no problema do processamento da informação ao nível simbólico. Esta dificuldade, colocada logo à partida na construção do modelo, é uma das razões pela qual a utilização do mesmo se tem revelado limitativo em sistemas de controlo evoluídos, ambientes complexos e condições reais.

3.2.2 Aproximação comportamental

A aproximação comportamental foca a sua atenção sobre a acção, reduzindo a “inteligência” ao mínimo, relegando-a mesmo para segundo plano. A sua característica principal encontra-se numa adaptação reactiva ao meio (logo após uma avaliação do mesmo), através de um processo de interacção contínuo. De uma forma geral, o mundo animal (particularmente os insectos) são a fonte de inspiração da aproximação baseada em comportamentos.

Os trabalhos da corrente comportamentalista segundo Rodriguez [Rodriguez, 94] apresentam as seguintes características:

1. O comportamento é o elemento central, onde a autonomia não resulta de uma sucessão de procedimentos funcionais e monolíticos entre a percepção e a acção mas preferencialmente da coordenação de comportamentos básicos. Esta alteração introduziu uma arquitectura diferente da arquitectura funcional apresentada no ponto anterior, que Brooks denominou de “arquitectura horizontal” (Figura 3.2).
2. O comportamento de base é suportado por uma estrutura mais ou menos directa que liga determinados sensores a determinados actuadores.
3. O processamento realizado entre os sinais e a parte de comando são reduzidos ao mínimo, tendo uma inteligência limitada ou mesmo inexistente.
4. O processamento situa-se a um nível subsimbólico, do género “estímulo-resposta” ou “condição-acção”, em oposição ao nível simbólico da aproximação cognitivista.
5. O problema principal encontra-se na coordenação dos comportamentos; possuindo um potencial de comportamentos básicos, é necessário organizar a sua influência na actividade global do sistema.
6. A coordenação tem tendência a efectuar-se através de um controlo distribuído e não centralizado como no caso da arquitectura cognitivista.
7. O conhecimento do sistema tende a aparecer na forma de uma representação distribuída, repartindo-se localmente pelos comportamentos a explorar.
8. De uma forma geral, da coordenação adaptativa dos comportamentos básicos coexistentes podem emergir novos comportamentos mais complexos.

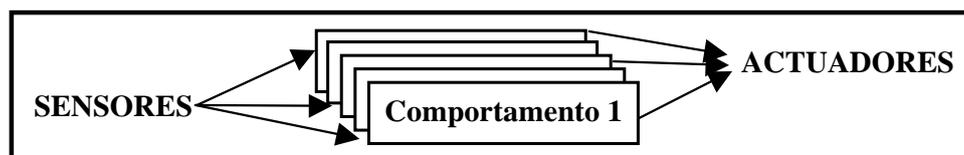


Figura 3.2 Arquitectura horizontal.

A aproximação comportamental pode então ser caracterizada pela existência de um modelo de informação distribuído onde o controlo global do sistema é decomposto num conjunto de módulos autónomos, sendo cada um deles responsável pela geração de um

comportamento. Esta metodologia baseia-se na decomposição de “inteligência” em módulos geradores de comportamentos individuais, de cuja coexistência e cooperação podem emergir comportamentos mais complexos. Nesta estrutura todos os módulos recebem entradas directas dos sensores e emitem comandos directamente para os actuadores.

Muita da investigação sobre o controlo baseado em comportamentos, nomeadamente na área da robótica, tem incidido no projecto de módulos comportamentais individuais apropriados e no desenho de técnicas de coordenação dos módulos geradores de comportamentos básicos, de forma a que o sistema autónomo possa executar acções flexíveis e robustas [Maes, 89] [Connell, 89] [Brooks, 90] [Anderson and Donath, 90] [Arkin, 90] [Maes and Brooks, 91] [Brooks, 91] [Goodridge et al., 96]. Em concreto, o problema é seleccionar quais os comportamentos que devem ser activados dando-lhes o controlo sobre os actuadores num determinado instante. O problema da coordenação dos comportamentos pode ser posicionada sobre duas aproximações diferentes que reflectem análises diferentes de observação do comportamento:

1. O primeiro ponto de vista consiste em considerar todos os comportamentos como independentes, podendo em qualquer instante um só comportamento desencadear uma acção.

Esta posição conduz à selecção de um comportamento entre N comportamentos possíveis independentes. Na literatura, esta problemática é descrita pela expressão “selecção de acções”. Muitos trabalhos foram realizados utilizando este ponto de vista, como são o caso dos de Brooks [Brooks, 85] [Brooks, 86] e Connell [Connell, 89].

2. O segundo ponto de vista, consiste em considerar a acção desencadeada como resultante de um compromisso entre as tendências de diferentes comportamentos.

Esta posição, conduz ao tratamento do problema da integração de um conjunto de potencialidades que se reflectem numa determinada acção. Esta problemática é denominada “integração comportamental”. Pode ser efectuada usando as mais diversas técnicas, como por exemplo a técnica dos campos potenciais [Anderson and Donath, 90] [Arkin, 90] ou lógica *Fuzzy* [Goodridge et al., 96]

Partindo destes dois pontos de vista duas filosofias de controlo emergiram como preponderantes:

1. Controlo por prioridades: a atribuição de prioridades aos comportamentos tem como consequência que os de mais alta prioridade fiquem com o controlo em detrimento de outros (Figura 3.3).

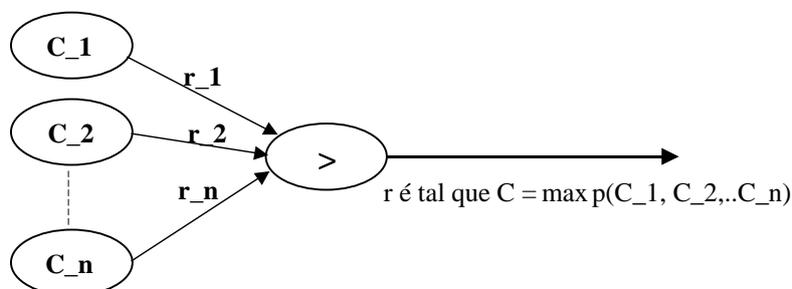


Figura 3.3 Controlo por prioridades.

2. Controlo por fusão: Em cada instante a acção realizada por o sistema é um compromisso entre as diferentes tendências especificadas pelos diversos comportamentos activos (Figura 3.4)

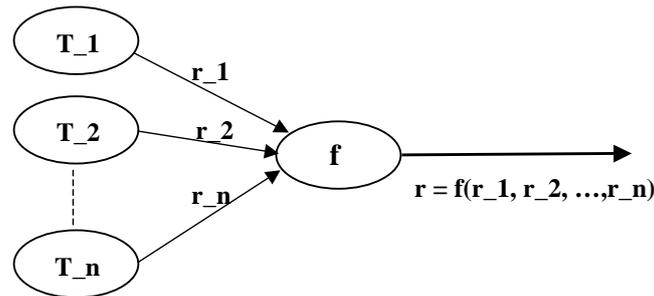


Figura 3.4 Controlo por fusão.

Controlo por prioridades: A *subsumption architecture* de Brooks

Rodney Brooks propõe em 1985 a *subsumption architecture* [Brooks, 85] como ferramenta de concepção de sistemas artificiais baseados em comportamentos.

O trabalho de Brooks propõe uma inteligência ascendente a partir de comportamentos reactivos simples e um raciocínio distribuído onde considera o ambiente do agente como modelo, em detrimento de uma representação interna.

A arquitectura baseia-se na decomposição comportamental em níveis de competência (Figura 3.5). Estes consistem numa especificação informal do comportamento que o robô deve ter sobre o ambiente em que se encontra. Um nível superior implica um comportamento mais específico, detendo uma certa influência ou prioridade sobre os níveis inferiores [Brooks, 85].

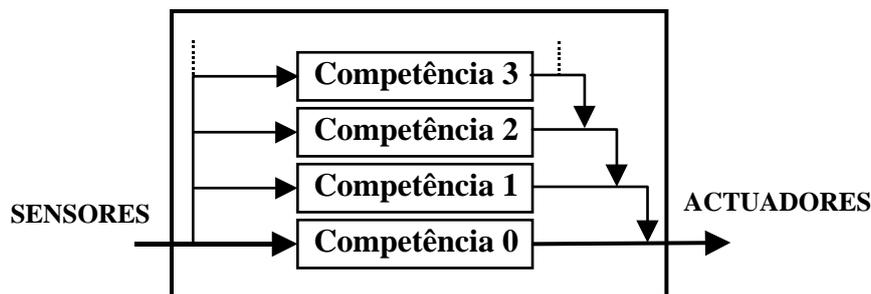


Figura 3.5 A *subsumption architecture*.

Os níveis de competência estão usualmente conectados directamente aos sensores e aos actuadores. Esta decomposição horizontal permite a distribuição da informação a processar por unidades especializadas, ligando cada nível aos sensores e actuadores mais adaptados às suas competências (enquanto que numa arquitectura funcional toda a informação proveniente dos sensores é centralizada e posteriormente processada).

Internamente, cada nível de competência é composto por um conjunto de módulos ligados entre si por canais de comunicação. Cada módulo dispõe de registos internos e de portas de entrada e saída para a recepção e envio de mensagens. As entradas podem estar ligadas aos sensores ou às saídas de outros módulos. As saídas podem estar ligadas aos actuadores ou às entradas de outros módulos. De um ponto de vista informático, um módulo é um processo paralelo assíncrono podendo ser descrito por uma máquina de estados finita.

Controlo por fusão: *Anderson e Donath*

Consiste em calcular uma solução que constitua um compromisso das respostas resultantes de N comportamentos, desencadeando uma única acção. O cálculo é normalmente realizado por um sistema central tendo em consideração todas as respostas dos comportamentos

candidatos. Esta técnica permite observar a influência dos diferentes comportamentos sob a resposta final do sistema.

Do ponto de vista do observador, Anderson e Donath modelam o comportamento como uma resposta reflexiva a um estímulo particular. Do ponto de vista interno decorre um processo uniforme no qual o sistema avalia a fidelidade da representação do mundo externo no seu estado interno, respondendo a alterações que ocorram. Os comportamentos não têm memória, são unicamente reacções instantâneas às situações observadas, podendo estas ser de dois tipos: **atração** ou **repulsão**.

Uma originalidade desta aproximação tem a ver com o estudo dos mecanismos que permitem a integração por fusão (em detrimento da sucessão de comportamentos independentes) proporcionando um comportamento mais complexo. Para implementar esta ideia, os autores utilizaram “campos potenciais” (representação uniforme inspirada na Física) que permite ao projectista juntar os comportamentos obtidos sobre um plano único. Os campos potenciais permitem desta forma uma integração conjunta das tendências comportamentais numa resposta única que é uma soma ponderada de todas as respostas (Figura 3.6).

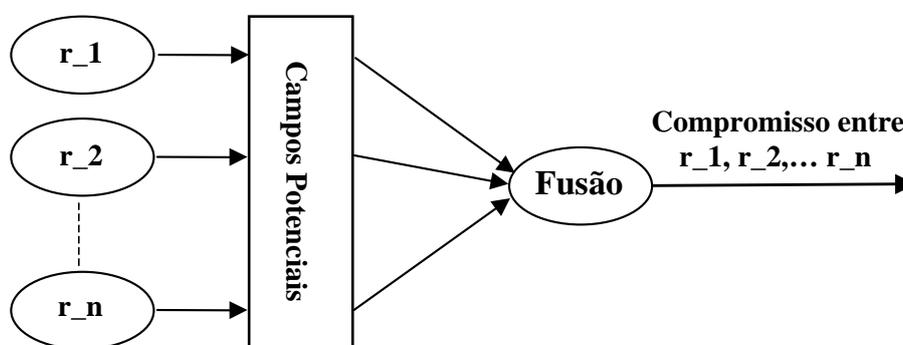


Figura 3.6 Arquitectura de fusão proposta por Anderson e Donath.

No caso da robótica móvel, o campo potencial é aplicado sobre uma carta polar centrada sobre o robô, permitindo representar todas as respostas dos comportamentos sobre a forma de vectores de movimento. A resposta do sistema após a fusão é um novo vector igual à soma ponderada das respostas obtidas a partir dos comportamentos activos.

Na Figura 3.7 apresentam-se os onze comportamentos básicos tratados neste trabalho. Cada um destes comportamentos é constituído por módulos que interpretam dados dos sensores, processam esses dados de alguma forma e produzem um sinal que é usado como comando de referência para os actuadores do robô. Deste modo, cada comportamento é independente dos outros e separado do sistema de controlo do robô. Esta base comportamental permite construir e observar 6 comportamentos emergentes correspondentes a combinações particulares dos comportamentos de base. Por exemplo, o comportamento emergente *Generalized wandering* emerge quando o comportamento básico *Forward-attraction* é activado concorrentemente com os comportamentos básicos *Passive-avoidance* e *Active-avoidance*; o comportamento *Perimeter following* emerge quando os comportamentos básicos *Passive-avoidance*, *Active-avoidance*, *Object-attraction*, mais um dos *Object following* (*Object-following CW* ou *Object-following CCW*) são activados concorrentemente.

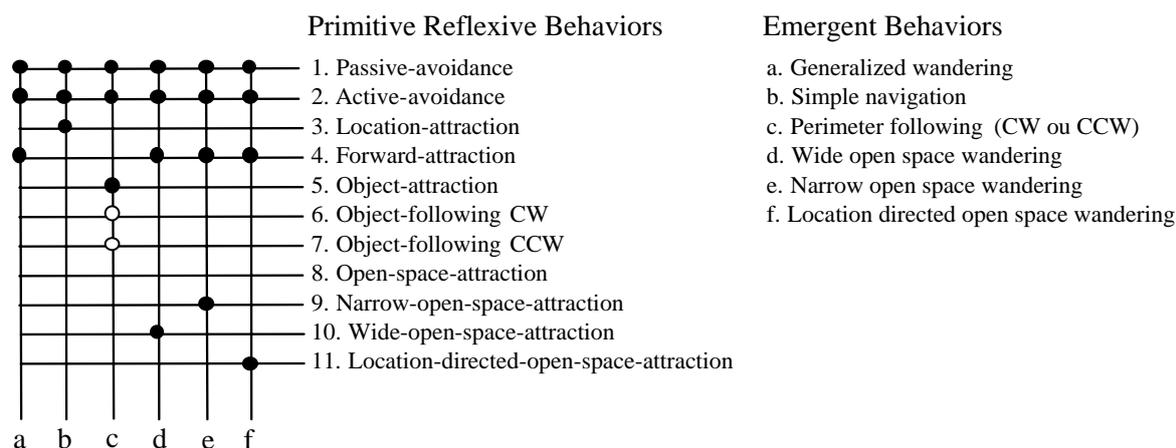


Figura 3.7 Onze comportamentos primitivos e seis comportamentos emergentes para fusão de diferentes combinações [Anderson and Donath, 90].

A dificuldade de uma abordagem deste tipo reside no estabelecimento de uma representação que permita uma resposta uniforme ao conjunto de respostas dadas pelos diferentes comportamentos. Por outro lado, os testes realizados por Anderson e Donath permitiram extrair dois problemas ligados à arquitectura de fusão:

- Dois comportamentos integrados podem em determinado momento e em situações concretas fornecer respostas que se anulam, resultando do processo de fusão uma resposta nula. Este problema denominado de estagnação, foi observado algumas vezes e está directamente ligado ao método da soma vectorial. Muitas soluções foram propostas, desde a utilização de um vector suplementar até à melhoria das funções de ponderação ou a existência de um sistema supervisor que detecte essas situações e forneça uma resposta.
- Os comportamentos integrados entram em ciclos infinitos. Esta situação pode resultar dos seguintes factores: limitações do robô para adequadamente adquirir a informação do ambiente; incapacidade do robô em responder a eventos que ocorram em pequenos períodos de tempo; tipo de resposta pretendida - resposta imediata a um estímulo. Estas situações fazem com que o robô fique dependente das contingências do ambiente. Contudo, nas aplicações em condições reais o ambiente geralmente introduz alterações suficientes que permitem a saída destes ciclos.

3.2.3 Aproximação híbrida

O paradigma cognitivista recorde-se, baseia-se numa metáfora computacional que propõe mecanismos de geração de raciocínios (geração de planos de acção em particular) baseado em modelos (simbólicos) do mundo. Esta aproximação desinteressa-se da natureza da actividade. Em oposição, o paradigma comportamentalista propõe mecanismos para o controlo de uma actividade localizada sem recorrer a qualquer representação interna, desinteressando-se pelo raciocínio a longo prazo.

A aproximação híbrida aspira a integrar uma acção localizada e um raciocínio a longo prazo num mesmo sistema de controlo. De um ponto de vista arquitectural duas linhas directrizes surgiram:

1. Integrar num só sistema uma parte comportamental e uma parte cognitiva. Esta solução é característica de uma aproximação híbrida. Ela, insere-se na continuidade de uma aproximação “descendente” do paradigma cognitivista [Arkin, 90].

As duas partes permanecem clássicas. A parte cognitiva ocupa-se de gerir o modelo do mundo e decidir quais os comportamentos a empreender numa problemática de coordenação (próximo da geração de planos de acção) e a parte comportamental encarrega-se da reacção (componente reactiva) em resposta a um acontecimento inesperado no ambiente.

Num quadro deste tipo, as dificuldades residem na falta de um ponto de interface que possibilite a correspondência entre a representação simbólica utilizada no nível cognitivo e as acções localizadas do nível comportamental.

2. Elaborar os mecanismos de raciocínio a partir da actividade. Esta solução se insere na continuidade da aproximação “ascendente” do paradigma comportamentalista [Mataric, 97]. O raciocínio e a representação são então fundamentados sobre a acção.

A aproximação descendente de Ron Arkin

Ron Arkin em [Arkin, 90] descreve uma arquitectura (AuRA - *Autonomous Robot Architecture*) composta por uma parte comportamental (reactiva) comandada por uma parte cognitiva (deliberativa). Arkin propôs a técnica dos campos potenciais para representação da interacção do robô com o seu ambiente de forma a coordenar os diversos comportamentos (componente reactiva de um robô móvel).

Nesta arquitectura, o conhecimento acerca do ambiente é representado por dois modelos. Um dos modelos, é constituído pela informação já conhecida acerca do ambiente em que o robô móvel se insere, podendo ser considerada relativamente estática ao longo de toda a missão do robô. Esta informação reside em memórias denominadas de “longo-prazo” (LTM - *Long-term*), indicando a inalterabilidade desses dados. Ao segundo modelo corresponde um conhecimento transitório/efémero, dinamicamente adquirido pelo robô no seu movimento. Esta informação reside em memórias denominadas de “curto-prazo” (STM - *Short-term*).

O AuRA é constituído por 5 subsistemas básicos: *Perception*, *Cartographic*, *Planning*, *Motor*, *Homeostatic Control*. A Figura 3.8 mostra as relações existentes entre os diferentes subsistemas.

A parte comportamental desta arquitectura é composta por rotinas de acção (*Motor*) e percepção (*Perception*) que permitem o tratamento localizado de actividades. As primeiras permitem o movimento de uma forma aleatória ou segundo uma dada direcção. Quanto às rotinas de percepção, elas são responsáveis pela extracção de informação ou características específicas do ambiente para realizar uma determinada tarefa.

A parte cognitiva é típica da corrente cognitivista, decompondo-se em todos os módulos da arquitectura funcional: percepção, modelação, planificação e acção. A função do módulo de percepção (*Perception*) consiste na integração da informação proveniente das rotinas de percepção. O módulo de modelação gera uma representação do ambiente (*Cartographic*). O módulo de planificação (*Planning*) é constituído por um gerador de planos de acção hierárquico que permite extrair os planos globais para as missões de navegação que em seguida são refinados pelo módulo de acção (*Motor*). Um módulo suplementar (*Homeostatic*

Control) encarrega-se de “restringir” os módulos de planificação e acção em função do estado de determinadas variáveis internas do sistema.

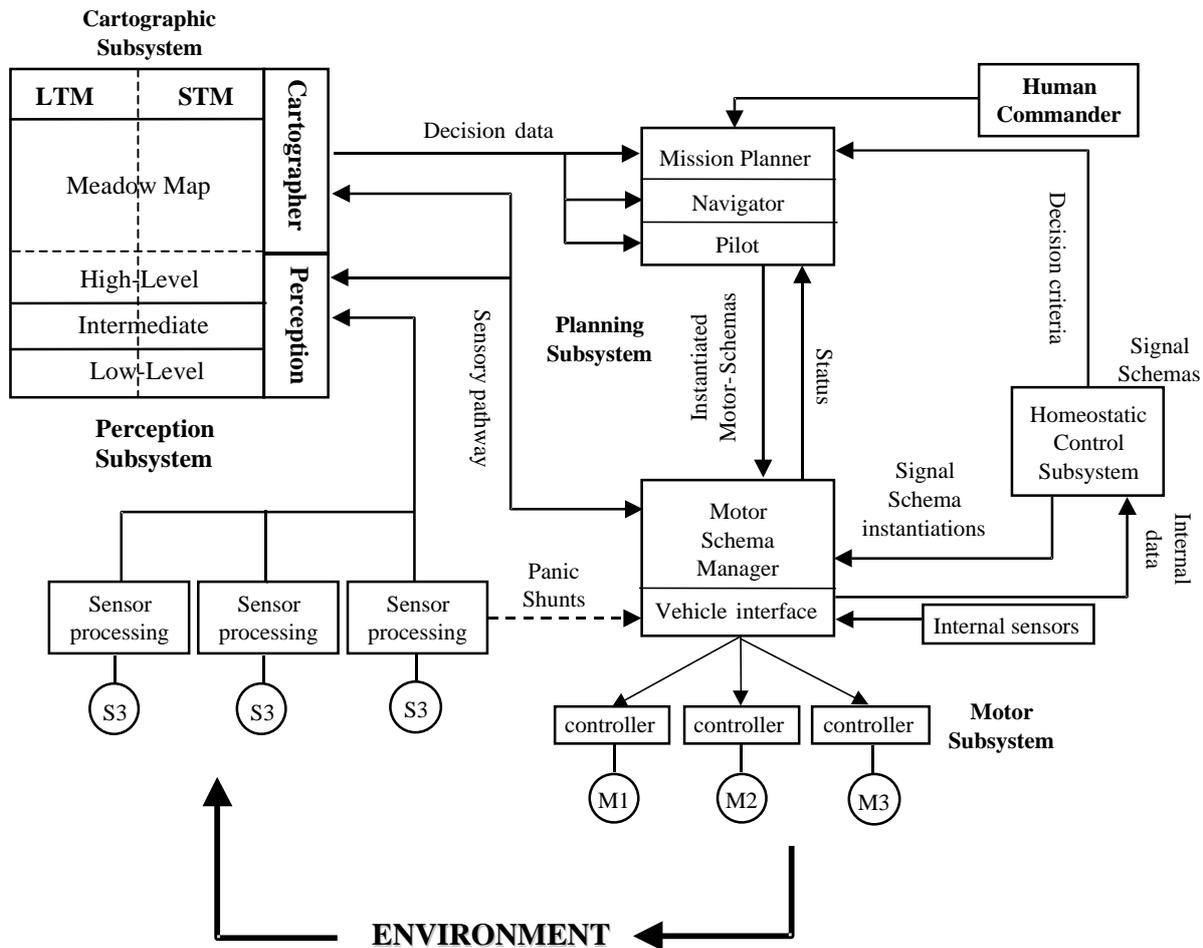


Figura 3.8 Arquitectura AURA [Arkin, 90].

A aproximação ascendente de Maja Mataric

Maja Mataric fez parte do laboratório de IA dirigido por Rodney Brooks no MIT. Devido aos seus antecedentes, Mataric não se revê numa aproximação híbrida do ponto de vista de arquitectura, preferindo separar os sistemas puramente reactivos dos sistemas baseados em comportamentos. Segundo ele, os sistemas baseados em comportamentos incluem algumas das propriedades dos sistemas reactivos, podendo conter componentes reactivos mas não sendo unicamente limitados a este tipo de elementos [Mataric, 97]. Estes sistemas podem usar diferentes formas de representação interna e realizar cálculos internamente com o objectivo de decidir qual a acção a efectuar.

Dentro deste quadro, ele sobrepõe níveis de competência que designaremos de cognitivos (tendo em consideração o que anteriormente foi referido) a níveis de competência ditos comportamentais. A Figura 3.9 representa esta arquitectura, onde os módulos comportamentais *safe wondering* e *boundary following* e os módulos cognitivos *map-learning* e *path-finding* se encontram separados por um módulo intermédio denominado *landmark detection*.

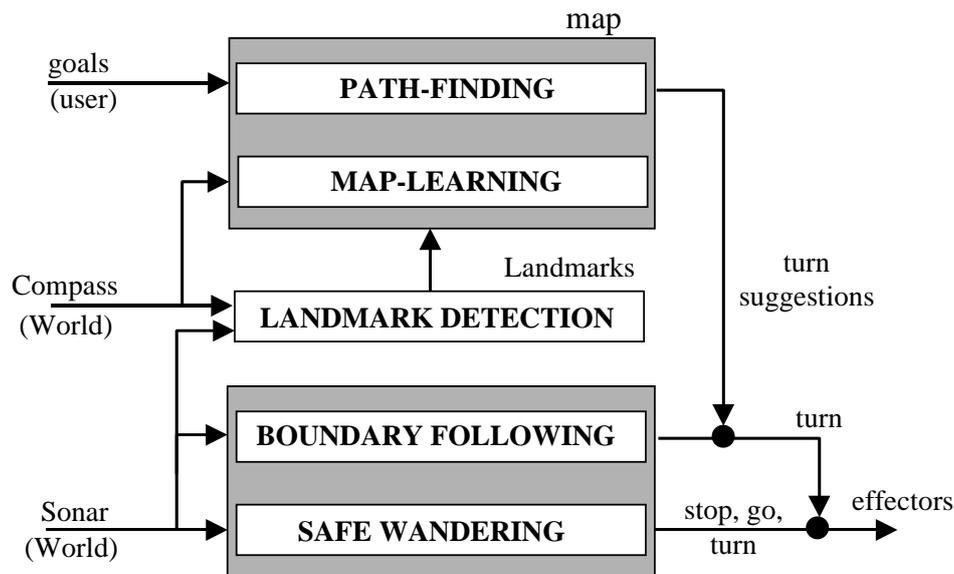


Figura 3.9 Arquitectura do sistema de Maja Mataric [Mataric, 97].

Os módulos comportamentais (*safe wondering* e *boundary following*) permitem a navegação evitando os obstáculos. O módulo intermédio (*landmark detection*) tem como função detectar as características regulares do ambiente (obstáculos à esquerda, obstáculos à direita ou obstáculos de ambos os lados). Estas características são transmitidas ao módulo de aprendizagem. O módulo de aprendizagem (*map-learning*) tem como função a construção de uma carta topológica do meio. O módulo de planificação (*path-finding*) permite intervir sobre o controlo enviando comandos ao sistema em função do objectivo introduzido pelo utilizador.

3.3 O Modelo de Referência RCS

A arquitectura de controlo RCS consiste num modelo de grande generalidade e, por isso, também de alguma complexidade de implementação, que tem por base um modelo teórico daquilo que se considera ser um sistema inteligente discutido em [Albus, 91].

O RCS evoluiu através de uma variedade de versões ao longo dos anos, à medida que o conhecimento acerca da complexidade e sofisticação de comportamentos inteligentes era cada vez maior. A primeira implementação (RCS-1) foi projectada por Barbera em meados dos anos 70 para uma aplicação de robótica em que era necessário a interacção com sensores. No RCS-1, é dada ênfase à combinação de comandos com a realimentação sensorial sendo em seguida calculada a resposta para cada relação entre objectivos e estados. A aplicação consistia no controlo de um manipulador com um sistema de visão de luz estruturada, para tarefas de procura visual. O RCS-1 foi profundamente influenciado por modelos biológicos, tais como o modelo do cerebelo de Marr-Albus, e o *Cerebellar Model Arithmetic Computer* (CMAC). Neste último é possível implementar a seguinte função

$$\text{Comando}_{t+1}(i-1) = H_i(\text{Comando}_t(i), \text{Estado}_t(i), \text{Realimentação}_t(i)) \quad (3.1)$$

onde, H_i é uma função de muitas variáveis; i é o índice indicando o nível hierárquico; t é o índice do tempo.

O CMAC torna-se uma máquina de estados quando algumas das suas saídas são ligadas directamente a entradas. Em consequência disso, o RCS-1 foi implementado igualmente como uma máquina de estados, adaptada a diferentes níveis de controlo hierárquico. Em cada nível, o comando de entrada selecciona o comportamento a ser transformado através de

realimentação na forma de estímulo-resposta. O CMAC torna-se então o modelo de referência do RCS-1 (Figura 3.10).

O RCS-1 é semelhante em muitos aspectos à *subsumption architecture* de Brooks, com uma excepção: o RCS selecciona os comportamentos antes de um evento acontecer, partindo dos objectivos expressos por comandos; no caso do Brooks é após acontecer o evento que é seleccionado o comportamento [Albus, 94].

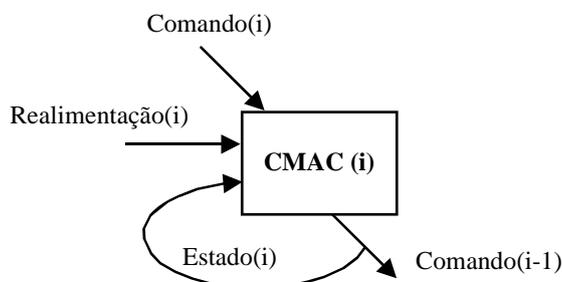


Figura 3.10 Bloco básico do RCS-1.

A geração seguinte do RCS - o RCS-2 (Figura 3.11) foi desenvolvida por Barbera, Fitzgerald, Kent e outros para o projecto AMRF (NIST *Automated Manufacturing Research Facility*), no princípio dos anos 80. No RCS-2 a função H é uma máquina de estados finitos executada a partir de um tabela de estados. A nova característica desta arquitectura foi a inclusão de uma função G, constituída por alguns algoritmos de processamento sensorial, de que são exemplo os algoritmos para técnicas de luz estruturada e análise de manchas. O RCS-2 foi definido em 8 níveis hierárquicos que consistem em: *Servo, Coordinate Transform, E-Move, Task, Workstation, Cell, Shop e Facility levels of control* [Albus, 94].

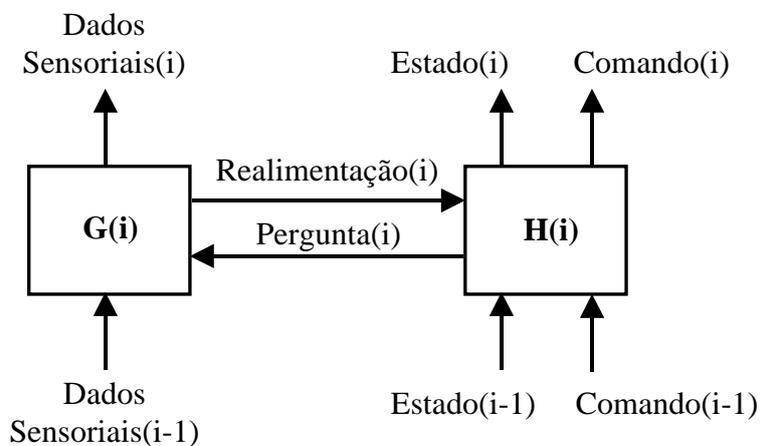


Figura 3.11 Bloco básico do RCS-2.

O RCS-3 (Figura 3.12) foi inicialmente desenvolvido para o projecto MAUV (NBS/DARPA *Multiple Autonomous Undersea Vehicle*) e posteriormente adoptado para o NASREM (NASA/NBS *Standard Reference Model Telerobot Control System Architecture*). A mais importante característica introduzida pelo RCS-3 é o Modelo do Ambiente (MA). Este serve de base ao planeamento de tarefas e ao processamento sensorial (PS). O refinamento posterior é uma função do módulo decomposição de tarefas (DT), que atribui apenas uma única tarefa ao executor de cada subsistema [Albus, 94].

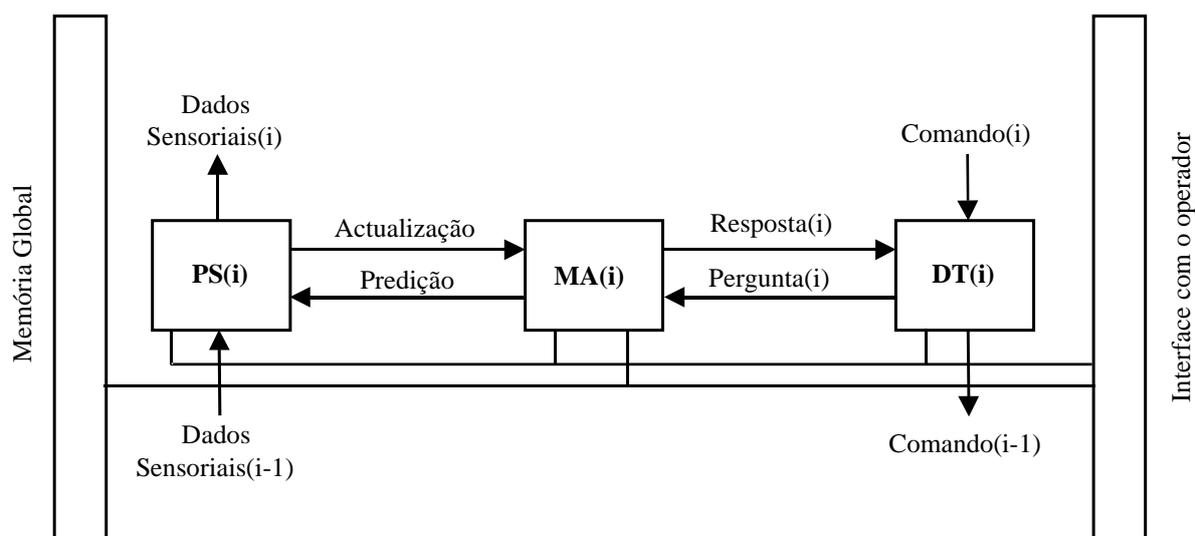


Figura 3.12 Bloco básico do RCS-3.

Finalmente o RCS-4 (Figura 3.13) é constituído por 4 módulos/elementos funcionais, denominados: Gerador de Comportamentos (GC), Processamento/Percepção Sensorial (PS), Julgamento de Valores (JV) e Modelo do Ambiente (MA). A inovação principal deste modelo é a existência do módulo Julgamento de Valores. Para Albus, estes 4 módulos funcionais são a base de um sistema inteligente em que as entradas e as saídas deste sistema são efectuadas respectivamente pelos sensores e actuadores [Albus, 91]. A partir deste momento o RCS-4 passa a ser denominado unicamente por RCS.

3.3.1 Elementos de um sistema inteligente

Para Albus um sistema inteligente é constituído por os 4 módulos já referidos (GC, PS, JV e MA). As entradas e as saídas de um sistema deste tipo são da responsabilidade respectivamente dos sensores e actuadores.

Actuadores

A saída de um sistema inteligente é sempre processada por actuadores, podendo expressar-se de diferentes formas: movimento, força, posicionamento de braços, pernas mãos e olhos. Os actuadores podem ainda gerar forças para posicionar sensores, excitar transdutores, manipular ferramentas e mover manipuladores. Um sistema inteligente poderá ter dezenas, centenas, milhares e até milhões de actuadores, podendo todos eles ser coordenados de forma a realizar tarefas de forma a satisfazer objectivos. Os actuadores biológicos são os músculos e as glândulas. Os actuadores máquina são os motores, os pistões, as válvulas, os solenóides entre outros transdutores de saída.

Sensores

As entradas de um sistema inteligente são produzidas por os sensores. Estes podem ser de muitos tipos, como por exemplo: sensores de brilho e cor; tácteis, de força, de binário; detectores de posição e/ou de velocidade, pressão e dispositivos de medida e temperatura. Os sensores podem ser utilizados para monitorizar o estado do ambiente externo e o estado interno dos próprios sistemas inteligentes, fornecendo as entradas a um módulo de processamento sensorial.

Processamento sensorial

Um módulo de processamento sensorial (PS) corresponde à transformação dos dados provenientes dos sensores em representações úteis e com sentido do ambiente. O PS recebe os dados provenientes dos sensores correspondentes a características do ambiente exterior e do próprio sistema interno. A correlação entre as grandezas medidas e as estimadas são usadas para detectar eventos e reconhecer entidades e situações. As variações entre estas mesmas grandezas são usadas para actualizar os modelos internos. Este módulo segundo Huang [Huang, 96] inclui as seguintes funções:

Aquisição de dados. Esta função pode incluir técnicas de amostragem, filtragem e validação dos dados provenientes dos sensores.

Integração de dados. Esta função integra os dados sensoriais provenientes de múltiplas fontes no espaço e no tempo.

A integração temporal dos dados produz uma média ao longo de uma janela temporal. Os limites desta janela podem ser obtidos a partir dos valores previstos para a duração dos eventos, ou através dos parâmetros definidos pelo módulo GC. A integração temporal dos valores de semelhanças ou diferenças, obtidos a partir da comparação do ambiente real com os valores previstos, produzem funções de correlação temporal entre o modelo e os dados observados.

A integração espacial sintetiza a informação sensorial vinda de diversas fontes num único ponto no tempo, determinando se as propriedades geométricas da entidade modelada do mundo correspondem às da entidade real.

Assimilação de dados. Esta função permite reconhecer padrões e/ou detectar eventos.

Modelo do ambiente

Ao módulo modelo do ambiente (MA) corresponde a construção, manutenção e uso das representações internas do ambiente.

O MA contém uma memória dinâmica denominada de “curto-prazo” para respostas rápidas e uma memória estática denominada de “longo-prazo” para respostas planeadas, que englobam relações pragmáticas e semânticas entre entidades, e ligações entre representações simbólicas. Este módulo tem como função responder às questões efectuadas pelo GC relacionadas com o estado do ambiente, simular resultados de possíveis futuros planos e estimar observações sensoriais com base no conhecimento inerente à base de dados. Na interacção entre o PS e o MA são calculadas transformações entre a representação descritiva e gráfica, sendo mantidos apontadores que ligam sinais a entidades simbólicas e vice-versa. Desta forma, é possível que os atributos das entidades simbólicas sejam actualizados a partir dos atributos dos sinais observados. Igualmente, permite prever os sinais a serem gerados a partir dos quadros de entidades simbólicas. Os sinais previstos podem ser usados pelo módulo de processamento sensorial para configurar filtros, máscaras, e janelas para a correlação e modelos de correspondência, assim como, estabelecer a correspondência entre as características do sinal. Segundo Huang [Huang, 96] o MA inclui as seguintes funções:

Gestão da base de dados. Esta função permite, partindo dos dados fornecidos pelo PS actualizar a base de dados em tempo real, reforçando/actualizando o conhecimento já existente.

Estimativa e previsão do estado. Os resultados desta função podem ser usados pelo GC para planear o próximo movimento ou pelo PS para esperar determinados dados provenientes dos sensores.

Gestão de perguntas. O GC pode questionar o MA durante o planeamento “Como é que faço o movimento” ou durante a execução “qual é o valor actual de X”. Esta função é responsável por extrair a informação da base de dados e gerar as respostas.

Julgamento de valores

O módulo de julgamento de valores (JV) calcula: os custos, riscos e benefícios das acções e planos; a estimativa da importância e valor dos objectos, eventos e situações; a confiança na informação; os efeitos compensadores e punitivos de eventos e estados perceptíveis.

Tendo em consideração a avaliação efectuada o JV selecciona objectivos e estabelece prioridades.

Segundo Huang [Huang, 96] este módulo inclui as seguintes funções:

Crítério de cálculo e actualização. A importância dos critérios pode ser determinada em função das tarefas a atribuir. Segurança, tempo, ou precisão podem ser os valores mais importantes em determinadas situações. Albus também descreve aspectos emocionais como um critério importante em [Albus, 91]. Esta função pode necessitar de actualização ou de novo critério de cálculo de forma a ser possível responder a novas solicitações.

Julgamento de valor. Calcula o custo e benefício dos planos colocados como hipótese baseando-se no critério estabelecido. Os resultados são enviados para o GC para selecção de um plano.

Gerador de comportamentos

O módulo gerador de comportamentos (GC) tem a responsabilidade de planear e controlar as acções de forma a atingir objectivos pretendidos. Recebe “ordens” para realizar tarefas com determinados objectivos e prioridades, formula e/ou selecciona planos e efectua acções de controlo. O desenvolvimento ou selecção de planos é efectuada a partir do conhecimento antecipado das tarefas e da avaliação efectuada pelo JV, em combinação com a informação em tempo real fornecida pelo MA. Desta forma é-lhe possível distribuir com maior eficiência as “ferramentas” e recursos a agentes e estabelecer um melhor escalonamento das acções. O GC controla as acções através da comparação do estado actual com o estado desejado previamente especificado por um plano, usando uma lei de controlo para calcular a melhor acção de forma a obter uma diferença nula. O GC pode ainda aprender modelos de sistemas e otimizar leis de controlo.

Segundo Huang [Huang, 96] o módulo GC inclui as seguintes funções:

Atribuição de tarefas. Decomposição espacial dos comandos que chegam ao nó pelos “subordinados”. O resultado poderá ser usado para um planeamento temporal.

O Planeador. O planeador usualmente tem a função de construir, avaliar e seleccionar a sequência de tarefas a executar a partir de um conjunto de hipóteses.

O Executor. Executa os planos seleccionados. A execução é determinada pela comparação entre os valores de comando enviados pelo “Planeador” e os valores observados fornecidos pelo MA.

O modelo para sistemas inteligentes proposto organiza os módulos que a constituem segundo as relações funcionais e o fluxo de informação apresentados na Figura 3.13.

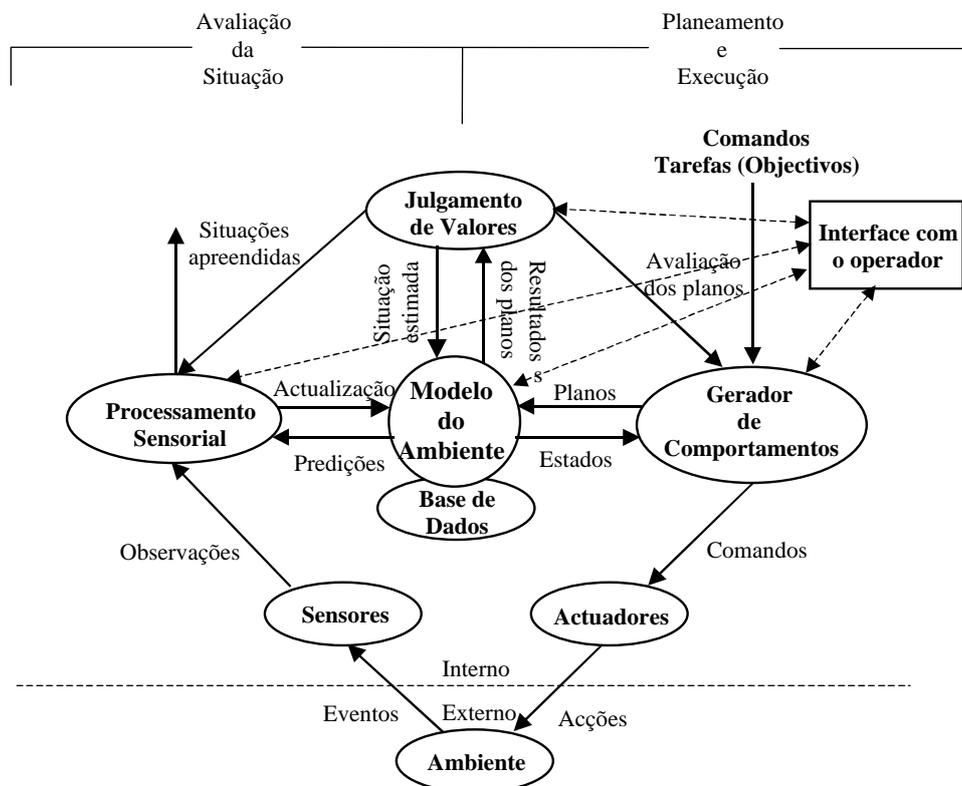


Figura 3.13 Modelo de referência RCS do Albus.

A interpretação da Figura 3.13 pode ser efectuada de uma forma genérica usando os argumentos do próprio Albus:

- Em todos os sistemas inteligentes existe um sistema que processa informação sensorial, a fim de adquirir e manter o modelo respeitante ao ambiente externo.
- Em todos os sistemas inteligentes existe um sistema de geração e planeamento de acções que controla os actuadores, visando objectivos comportamentais globais, no contexto do modelo apreendido.
- Em sistemas com níveis mais elevados de inteligência o módulo de geração de comportamentos pode interagir com o modelo do ambiente e com o sistema de apreciação (JV) para ponderar sobre parâmetros relativos a espaço e tempo, geometria e dinâmicas, formular ou seleccionar planos com base em dados de custo, risco, utilidade e prioridade de objectivos. Nestes sistemas a componente sensorial pode interagir com o modelo do ambiente e com o sistema de apreciação a fim de atribuir valores a entidades e eventos a situações observadas e apreendidas.

3.3.2 Arquitectura para sistemas inteligentes

Com base no modelo de sistemas inteligentes referido, Albus propôs uma arquitectura de controlo para sistemas inteligentes onde essas relações são repetidas e distribuídas através de uma estrutura computacional mista, hierárquica e horizontal, como se ilustra na Figura 3.14. A arquitectura é hierárquica porque os comandos fluem hierarquicamente de cima para baixo na cadeia dos módulos GC. O processamento de informação sensorial também é hierárquico, fazendo-se segundo o sentido ascendente na hierarquia de módulos de PS. A arquitectura é

ainda hierárquica, uma vez que o processamento sensorial e as funções de representação do ambiente têm níveis hierárquicos no que respeita à resolução temporal e espacial. A arquitectura exhibe estruturas horizontais, uma vez que os dados são partilhados entre módulos heterogéneos de um mesmo nível hierárquico. Cada nível consiste num controlador inteligente capaz de planear, controlar, representar o conhecimento, fazer julgamento de valores, bem como efectuar processamento sensorial.

As funções e características de cada nível podem ser resumidas nos seguintes pontos:

- As variáveis de estado, entidades, eventos e mapas, são mantidos sendo a resolução espacial e temporal apropriados para cada nível.
- As malhas de realimentação são fechadas, fornecendo comportamentos reactivos com grande largura de banda e de resposta rápida nos níveis mais baixo, e de resposta lenta com reacções deliberadas nos níveis mais altos [Albus, 96a]. O RCS fornece o que Brooks chama de “comportamento coerente de vários processos adaptativos” [Brooks, 94].
- Os objectivos globais elaborados pelos níveis mais altos são refinados e concentrados em direcção a uma decomposição mais refinada em sub objectivos.
- A atenção é focada na obtenção de uma resolução mais refinada do modelo do ambiente.

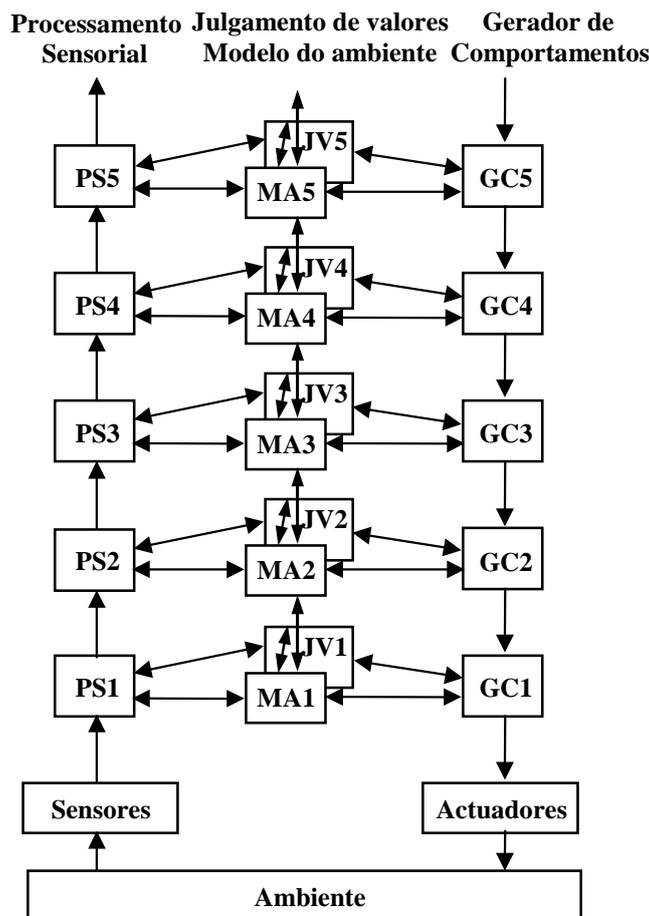


Figura 3.14 Relações funcionais no sistema de controlo misto hierárquico-horizantal do Albus.

Em resumo, o objectivo de cada nível hierárquico é o refinamento com um maior detalhe das tarefas e compreensão do ambiente, fortalecendo a carga computacional de todos os

níveis dentro de um limite que pode ser gerido por agentes inteligentes de modesta capacidade.

No topo da hierarquia são definidos objectivos estratégicos, definidas a influência das prioridades na selecção dos objectivos e das tarefas em toda a hierarquia.

Nos níveis mais altos são produzidos planos, é efectuado o reconhecimento de entidades e são realizadas simulações de modelos do ambiente mais complexos, dispondo por esse motivo de maiores intervalos de tempo para executar estas tarefas. Contudo, os detalhes da execução são deixados para os subordinados.

Nos níveis intermédios, tarefas com objectivos e prioridades são recebidas dos níveis superiores, e subtarefas com sub objectivos e com menor prioridade são enviadas para os níveis inferiores. O detalhe da execução é deixado novamente aos subordinados.

Aos níveis hierárquicos inferiores corresponde um maior detalhe espacial e maior resolução temporal, o que implica planos mais detalhados e horizontes de planeamento mais retraídos. Estas características dão origem a um menor alcance dos objectivos a atingir.

3.4 Arquitectura de controlo adoptada

A implementação de uma arquitectura de controlo envolve diversos passos, num processo de refinamento constante do conhecimento. Envolve compreensão, assimilação, ordenação e integração do conhecimento no domínio do problema e dos sistemas. Envolve também conhecimentos de computação e de controlo, recorrendo a uma aproximação sistemática. A identificação da informação relevante, função e desempenho pretendidos, interfaces, restrições existentes e critérios de validação, são factores importantes na implementação de qualquer arquitectura.

Tendo em consideração o anteriormente referido, foram utilizados diagramas de fluxo de dados (DFD) para ilustrar a relação entre os processos no respeitante ao fluxo de informação existente entre os mesmos. Os DFD são sucessivamente refinados até as suas transformações exibirem uma larga coesão, ou seja, descreverem uma função que possa ser implementada por um módulo. Com o objectivo de se iniciar a análise das tarefas é estabelecido o primeiro nível de abstracção (DFD nível zero), conforme se apresenta na Figura 3.15.

A Figura 3.16 apresenta o DFD nível 1 resultante do refinamento efectuado no DFD de nível 0. Este DFD já exhibe um nível de resolução considerável, servindo de ponto de partida para a obtenção de um esboço da arquitectura de controlo a implementar (Figura 3.17).

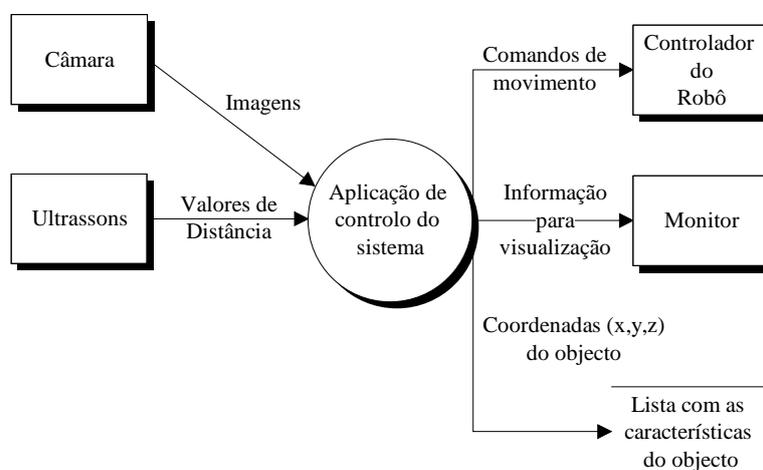


Figura 3.15 DFD nível zero.

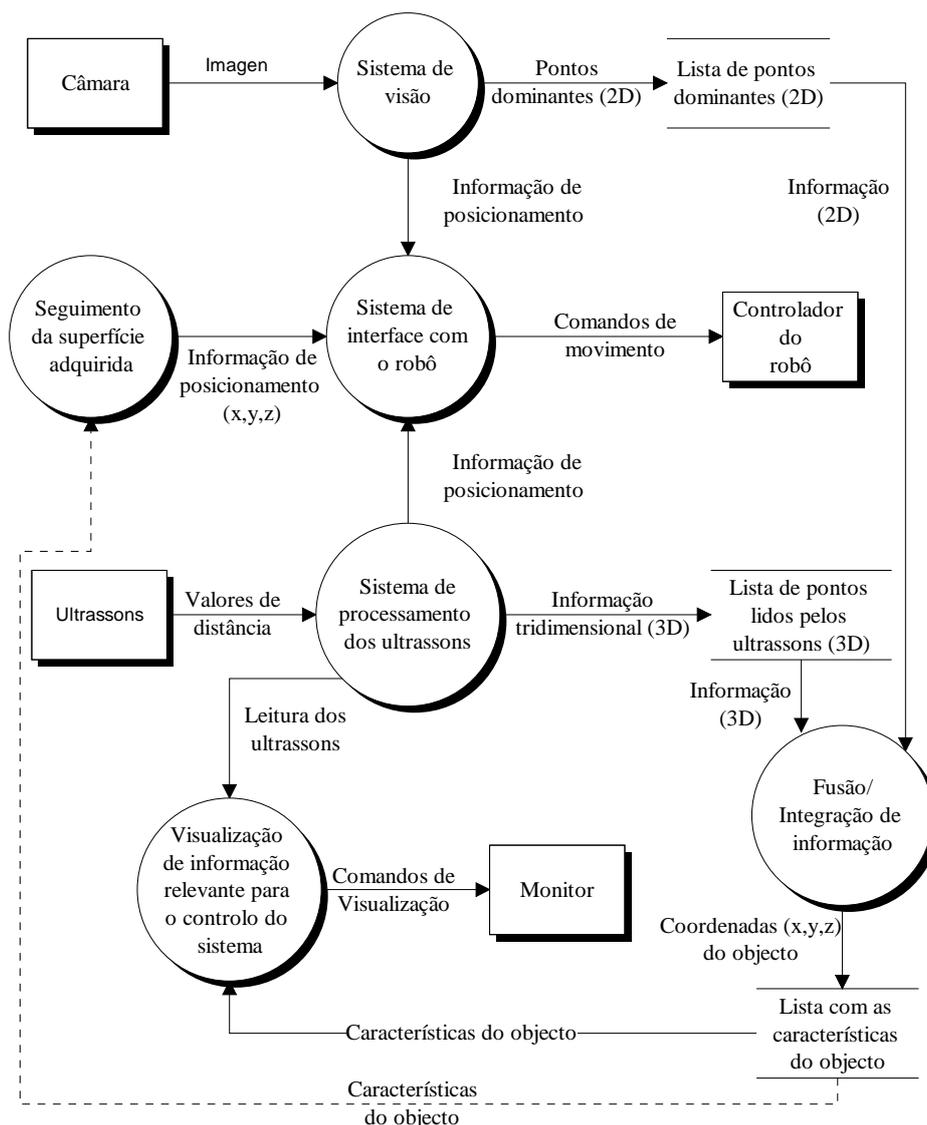


Figura 3.16 DFD nível um.

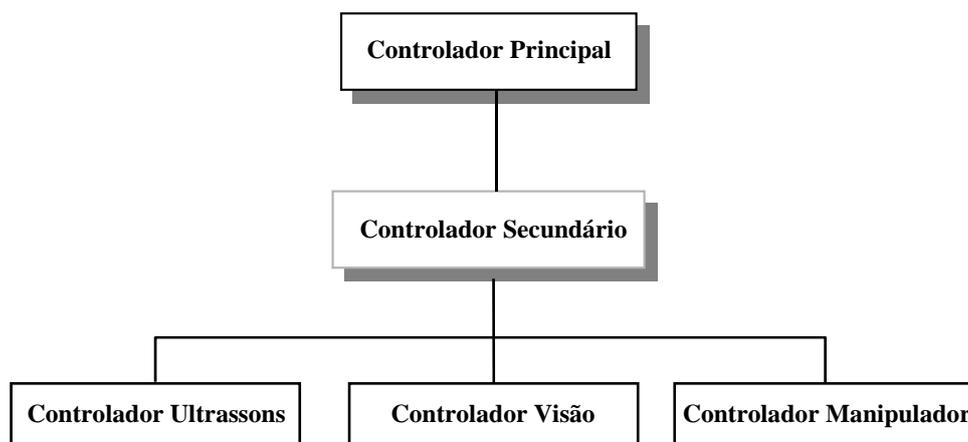


Figura 3.17 Esboço da arquitectura de controlo a implementar.

O controlador principal tem a função de coordenar as seguintes tarefas: procura/deteção do objecto na área de trabalho; aquisição dos contornos e do relevo da superfície do objecto

encontrado; seguimento da superfície adquirida (não implementado nesta fase). Estas tarefas são posteriormente entregues aos seus subordinados.

O controlador secundário decompõe as tarefas que lhe são atribuídas pelo seu superior em subtarefas, com um menor nível de abstracção mas com um maior detalhe e resolução. Essas subtarefas são: procura do objecto no espaço de trabalho, alinhamento da câmara relativamente ao centro de massa do objecto, extracção dos pontos dominantes que permitem descrever o contorno do objecto, controlo dos ultrassons, controlo da sequência de rastreio para obtenção da profundidade e geração das trajectórias de seguimento da superfície. Os seus comandos são distribuídos por três controladores subordinados.

O controlador do manipulador decompõe os comandos de movimento do robô em trajectórias que levam o manipulador até à posição pretendida.

O controlador de visão recebe comandos de pedido de aquisição de imagem que decompõe nas subtarefas necessárias à obtenção da informação bidimensional (2D) da imagem.

O controlador dos ultrassons recebe pedidos de leitura da distância que decompõe nas subtarefas necessárias à obtenção de imagens tridimensionais (3D).

A diversidade da natureza dos dados de entrada, o reconhecimento e planeamento das tarefas envolvidas, as ligações requeridas entre actuação e sensorização fazem do desenho da estrutura de controlo deste projecto um problema não trivial, necessitando de uma base conceptual com alguma complexidade. A arquitectura RCS foi a adoptada em detrimento das arquitecturas cognitivista, comportamental e híbrida, pelos seguintes motivos:

1. Os problemas da arquitectura cognitivista quando utilizada em aplicações reais, como por exemplo, tempos de resposta proibitivos, falta de robustez e flexibilidade quando surgem imprevistos.
2. A arquitectura comportamental está mais adaptada a sistemas que interagem com ambientes em constante mutação, o que não se verificava nesta aplicação. Pela razão apontada esta arquitectura é normalmente utilizada em robótica móvel, como se demonstra pelas muitas publicações existentes.
3. A arquitectura híbrida composta por uma parte cognitivista e uma parte comportamental também não foi a opção escolhida, porque a parte comportamental não era necessária pelos motivos anteriormente referidos. Por outro lado, o RCS é uma arquitectura mais abrangente sendo possível implementar em que cada nó um sistema de controlo híbrido.

O mapeamento da arquitectura de controlo anteriormente apresentada num modelo RCS, foi distribuído por quatro níveis e três hierarquias diferentes (Figura 3.18): visão, ultrassons e manipulador.

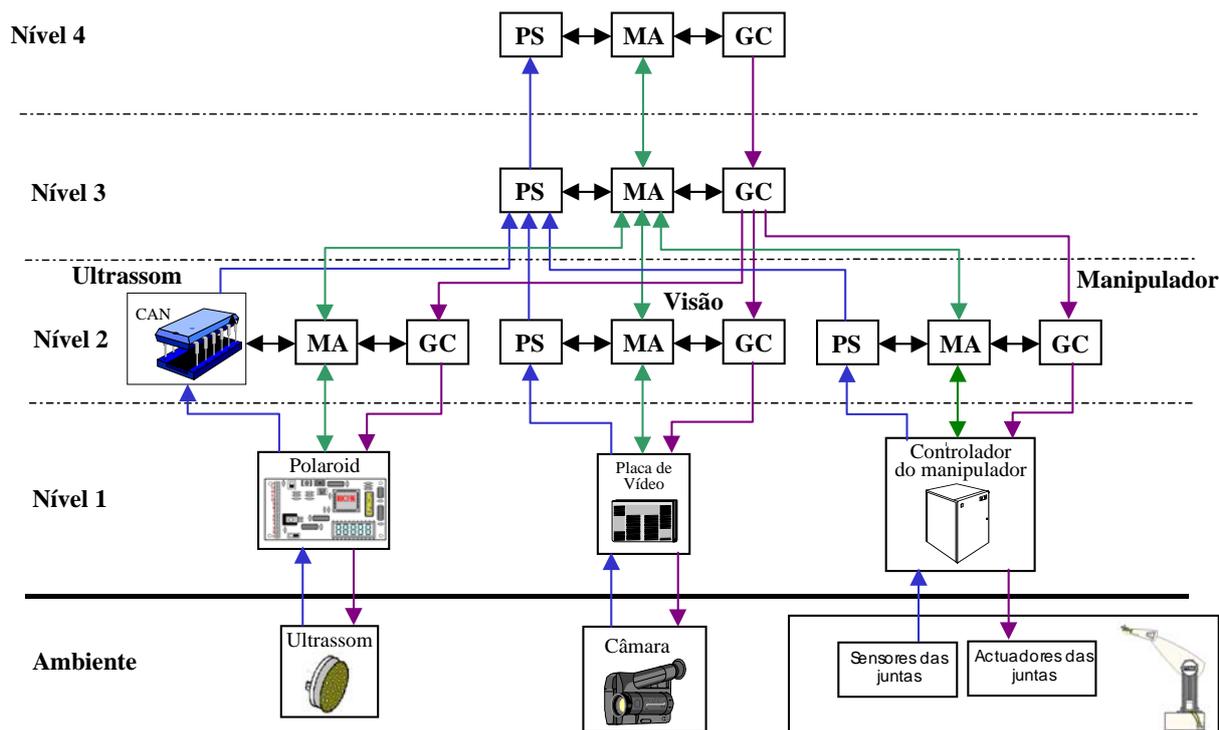


Figura 3.18 Arquitectura RCS.

O controlador de qualquer nível apenas conhece a linguagem descritiva dos objectivos que ele tem de atingir e consequentemente dos comandos e subtarefas nos quais são decompostos.

A descrição de cada um dos níveis utilizados e a enumeração dos seus objectivos funcionais são apresentados em seguida.

Nível 4 – Nível de Equipamento ou da tarefa

Neste nível, todo o ambiente de trabalho é referenciado correspondendo ao controlador principal da arquitectura de controlo. As tarefas de controlo são coordenadas e entregues aos controladores do nível inferior.

As tarefas deste nível são as seguintes:

- **Procura/Deteccção do objecto na área de trabalho do robô.**
- **Aquisição bidimensional do objecto.** O objectivo é a restrição da área de trabalho do robô.
- **Posicionamento do manipulador para iniciar o rastreio da superfície do objecto.** O objectivo é a actualização da área de trabalho do robô de forma a que o seu posicionamento inicial seja o mais adequado para uma melhor aquisição da 3D.
- **Adquirir a informação 3D (relevo) da superfície do objecto visível para o robô.**

Nível 3 – Nível Elementar

O PS deste nível tem a função de efectuar a fusão e a integração dos dados fornecidos pelos módulos subordinados de forma a que se possa construir um mapa da superfície do objecto. Estes dados constituem o MA global deste nível. O GC em interacção com o PS e MA é responsável pelo alinhamento da câmara e pelo controlo da sequência de rastreio da

superfície. Após a construção do mapa da superfície o GC será o responsável pela geração de trajectórias de seguimento da superfície.

Este nível apresenta um maior detalhe nos seus objectivos funcionais e a decomposição em subtarefas é processada da seguinte forma:

Procura/Deteção objecto na área de trabalho do robô. Decompõe-se nas seguintes subtarefas:

- Mover manipulador (entregue à hierarquia do manipulador).
- Aquisição/Processamento da imagem (entregue à hierarquia de visão).

Aquisição bidimensional do objecto. Consiste essencialmente na seguinte tarefa:

- Aquisição/Processamento da imagem (entregue à hierarquia de visão).

Posicionamento do manipulador para iniciar o rastreio da superfície do objecto. Decompõe-se nas seguintes subtarefas:

- Mover manipulador de forma a que os sensores ultrassons fiquem a apontar para o objecto (entregue à hierarquia do manipulador).
- Pedido de leitura aos sensores de ultrassons posicionados na parte superior da garra para detecção/validação dos limites reais da peça (entregue à hierarquia ultrassons).
- Mover o manipulador para a posição inicial de rastreio (entregue à hierarquia manipulador).

Adquirir informação da superfície do objecto (relevo). Consiste nas seguintes subtarefas:

- Pedido de leitura aos 4 sensores de ultrassons localizados na garra (entregue à hierarquia ultrassons).
- Mover o manipulador (entregue à hierarquia manipulador).

Nível 2 – Nível primitivo

Este nível assegura a suavidade da cinemática e das actividades de controlo, assim como do sucesso da dinâmica requerida. O modelo do ambiente pode conter nomes e atributos de características lineares tais como linhas, segmentos de trajectórias e vértices. No caso desta aplicação é constituído por três controladores individuais que se ocupam de partes distintas do projecto.

- **Hierarquia Manipulador.** O módulo PS recebe os ângulos descritivos de cada junta, que processa e transforma na posição corrente do mundo (em coordenadas cartesianas X-Y-Z) e na respectiva orientação (em graus O-A-T), correspondentes à localização do robô (cinemática inversa). O módulo GC recebe a posição pretendida para o robô em coordenadas do mundo (X-Y-Z) e da orientação desejada (O-A-T), que processa e transforma nos ângulos descritivos de cada junta (cinemática directa), gerando a trajectória necessária para atingir a nova posição. O MA gere a base de dados que contém a posição final do manipulador em coordenadas (X-Y-Z O-A-T) sendo actualizada com os dados fornecidos pelo PS, respondendo ao GC se a nova posição é igual ou diferente da que se encontra armazenada.

- **Hierarquia Visão.** Recebe pedidos de aquisição da imagem do objecto. A função do PS é efectuar o processamento da imagem (2D), agrupando pontos de forma a deduzir características do objecto (ex: contorno e pontos dominantes). A função do GC é gerar comandos que indiquem o momento em que a aquisição da imagem deve ser efectuada. O MA gere a base de dados que contém as características do objecto adquirido, fazendo a sua actualização com os dados fornecidos pelo PS. O GC pergunta ao MA se é necessário a aquisição de uma nova imagem ou se pode terminar o processo.
- **Hierarquia Ultrassons.** Recebe pedidos de leitura para os 4 ultrassons localizados na garra. O módulo PS deste nível corresponde à placa Universal CAN I/O, a qual recebe um conjunto de valores de distância efectuando a sua validação. O GC gera comandos que indicam qual o sensor que fica activo e o número de leituras a efectuar pelo mesmo. O MA faz a gestão de uma base de dados que contém os valores de distância lidos pelos 4 sensores. A sua actualização é efectuada com os dados fornecidos pelo PS. O GC pergunta ao MA qual o sensor que esteve activo (só um sensor fica activo num determinado momento).

Nível 1 – Nível do actuador ou servo

Os módulos GC iniciam acções que produzem como saídas os sinais eléctricos, mecânicos, hidráulicos e pneumáticos, necessários para activar os actuadores. Os módulos de PS recebem e processam os sinais provenientes dos sensores. Os módulos MA neste nível contêm valores de variáveis de estado tais como, posição das juntas, velocidade, força, leituras de sensores de proximidade, posição de comutadores, assim como atributos das imagens (cor e brilho associados a pontos) obtidas por câmaras.

- **Hierarquia Manipulador.** Na presente aplicação, é o controlador do manipulador quem executa os módulos PS, GC e MA. O GC converte os ângulos fornecidos para cada junta nos sinais eléctricos requeridos pelos actuadores das mesmas. O PS recebe dos sensores de junta os sinais eléctricos que processa e transforma até obter o ângulo respectivo de cada junta. O MA deste nível contém o valor da posição das juntas e o valor da velocidade, sendo actualizado a partir dos dados fornecidos pelo PS.
- **Hierarquia Visão.** Os módulos PS, GC e MA neste nível são da responsabilidade da placa de digitalização. O GC produz os sinais eléctricos enviados à câmara para aquisição da imagem. O PS processa os sinais eléctricos provenientes da câmara transformando-os em pontos. O MA contém os atributos dos pontos, como por exemplo o brilho e a cor, sendo actualizado com os dados fornecidos pelo PS.
- **Hierarquia Ultrassons.** Os módulos PS, GC e MA são da responsabilidade da placa *Polaroid*. O GC envia aos sensores de ultrassons os sinais eléctricos necessários para execução das leituras e o sinal eléctrico para o relé que liga o ultrassom que fica activo. O PS recebe os sinais eléctricos provenientes dos ultrassons resultante do eco recebido por estes e calcula o valor da distância. O MA contém o valor da distância sendo actualizado a partir dos dados fornecidos pelo PS e os parâmetros de funcionamento dos ultrassons como por exemplo a frequência, número de pulsos a transmitir, ou o tempo de vazio.

3.5 Sumário

Neste capítulo foram abordadas as arquitecturas de controlo mais utilizadas em sistemas multi-sensoriais, nomeadamente as arquitecturas cognitivista, comportamental e híbrida, tendo sido ainda dado realce à arquitectura RCS proposta por Barbera em 70 e adoptada posteriormente por Albus.

A aproximação cognitivista na modelação e concepção de agentes autónomos faz a mesma abordagem da IA tradicional. Ela reduz a autonomia à capacidade de raciocínio. Dentro desta abordagem foi proposta a arquitectura funcional como modelo do raciocínio. Neste, a preceder uma acção, existe sempre uma sequência de raciocínio que conduz a uma planificação *off-line*, o que implica uma complexidade exponencial dos algoritmos. Por conseguinte, em sistemas reais, este tipo de arquitectura apresenta respostas proibitivas e pouca flexibilidade face ao imprevisto.

Como resposta às limitações apresentadas pela arquitectura anterior surgiu a arquitectura comportamental que se baseia num raciocínio local onde o processamento se situa a um nível subsimbólico do género “estímulo-resposta” ou “condição-acção”. Contudo a originalidade desta aproximação constitui igualmente a sua maior limitação, pois grande parte do tempo é dispendido a reagir ao ambiente, não realizando a tarefa pretendida. Por outro lado a falta da componente deliberativa (cognitiva) impede a realização de tarefas com determinada complexidade.

Finalmente, como alternativa às dificuldades apresentadas pelas anteriores arquitecturas surgiu uma arquitectura híbrida que oferece um compromisso entre a componente cognitiva e a comportamental. Em arquitecturas deste tipo é normalmente usado um sistema reactivo para controlo de baixo nível e um sistema de planeamento para controlo de mais alto nível.

Em alternativa às anteriores arquitecturas surgiu a arquitectura RCS. Esta consiste num modelo de grande abrangência (o que implica alguma complexidade de implementação), que tem por base um modelo teórico do que se considera um sistema inteligente. Por outro lado a RCS apresenta uma grande flexibilidade permitindo aumentar o número de nós facilmente em resposta quer a um aumento do número de sensores ou actuadores ou da complexidade das tarefas a efectuar pelo sistema. Esta arquitectura foi a adoptada (Figura 3.18) ao longo do trabalho devido às limitações apresentadas para as arquitecturas cognitivista, comportamental e híbrida.

Capítulo 4

4. A célula de trabalho

4.1 Introdução

O objectivo deste capítulo é fazer uma apresentação geral da infra-estrutura experimental montada para o desenvolvimento deste trabalho e justificar algumas das opções tomadas ao nível do *hardware*.

O capítulo começa por apresentar o termo robô, dando alguma ênfase à definição dos grandes estágios de evolução porque tem passado a robótica. Em seguida, é apresentado o manipulador PUMA 560 utilizado neste trabalho, suas limitações e modificações efectuadas.

Na secção seguinte, é feita uma apresentação e classificação dos sensores mais utilizados na robótica. Posteriormente, é dado realce aos sensores utilizados, nomeadamente, a câmara de vídeo (câmara CCD), sensores de infravermelhos e de ultrassons. Na apresentação destes é dado ênfase à sua função, características, limitações e soluções adoptadas.

4.2 Robôs do tipo manipulador

A palavra *robot* surge da tradução para inglês das palavras checas ‘*robota*’ e ‘*robotnik*’ que significam em checo respectivamente trabalho pesado/forçado e servo. A popularidade deste termo tem a ver com uma peça literária do checoslovaco *Karel Capek* datada de 1921, onde as máquinas substituíam os homens no trabalho, acabando por eliminar os seus patrões e dominar o mundo.

Actualmente, existe uma distinção entre os domínios da robótica móvel e da robótica fixa. São exemplos dos primeiros, as plataformas móveis ou ainda os robôs com “pernas” desenvolvidos em diversos centros de investigação do mundo. No outro campo da robótica predominam os braços articulados, encontrando-se actualmente numa fase bastante avançada e muito divulgados a nível industrial, nomeadamente na indústria automóvel, sendo normalmente designados por robôs industriais ou manipuladores. Estes dois grandes campos da robótica distinguem-se sobretudo pela forma como os robôs interagem com o meio que os rodeia, mais concretamente, nas questões de percepção e interpretação da informação necessária para levar a cabo uma determinada tarefa.

O robô do tipo manipulador é um dispositivo mecânico cuja base é fixa mas com mobilidade num conjunto de articulações, que constituem o braço do sistema, a cuja extremidade - o pulso - se liga frequentemente um actuador mecânico com graus de

mobilidade adicionais. Existem diversas classificações para este tipo de robôs. A primeira classificação que usualmente é realizada está associada a uma perspectiva histórica, classificando os robôs por gerações. Quatro ou cinco gerações são normalmente enumeradas, distinguindo-se essencialmente pelo grau de autonomia e pela sua flexibilidade ou adaptabilidade a novas situações.

A *Japanese Industrial Robot Association* (JIRA) adoptou outra classificação para os robôs, tendo definido seis classes com base no seu grau de “inteligência”. Estas vão desde a classe que agrupa os robôs menos flexíveis, denominados manipuladores manuais, até à classe que agrupa os robôs de maior flexibilidade que percebem e interagem com as modificações do ambiente, denominados de robôs inteligentes. Com base ainda no grau de “inteligência” e sendo esta entendida como a capacidade de adquirir e aplicar conhecimento, podemos ser mais restritivos e agrupar os robôs em três grandes categorias, concretamente, em robôs de primeira, segunda e terceira geração. Os robôs ditos de primeira geração não possuem capacidade de interagir de uma forma adaptativa com o ambiente. As suas capacidades limitam-se à execução determinística e repetitiva de sequências de acções previamente programadas. Os robôs deste tipo comportam-se basicamente como um sistema versátil de posicionamento. Os robôs de segunda geração distinguem-se dos anteriores por possuírem a capacidade de interagir de uma forma adaptativa, ainda que limitada, com o meio ambiente. Esta capacidade torna-se possível pelo uso de sistemas sensoriais que adquirem informação a respeito do mesmo. Os robôs da terceira geração são objecto de investigação e desenvolvimento em numerosos centros de investigação do mundo. Destes, espera-se que venham a possuir aptidão para interagir com o meio ambiente de uma forma inteligente, exibindo uma grande versatilidade na execução de tarefas diversificadas, traduzido nas capacidades de percepção, decisão e planeamento.

Um outro tipo de classificação adoptado tem como critério o nível da linguagem de programação usada. Três categorias são normalmente aceites: 1—sistemas guiados, onde o utilizador define quais os movimentos a ser efectuados; 2—sistemas programados ao nível do robô, onde o utilizador escreve um programa que determinará o movimento e a percepção; 3—sistemas de programação ao nível da tarefa ou objectivos, onde o utilizador especifica apenas as operações que o robô deverá efectuar nos objectos e/ou ambiente com que interage. Trata-se de programação com mais alto nível de abstracção, onde os detalhes para executar uma dada tarefa são perfeitamente transparentes para o utilizador.

4.2.1 O manipulador PUMA 560

O robô usado ao longo deste trabalho é um manipulador industrial PUMA 560 com o controlador MARK II. Trata-se de um robô não redundante, com seis juntas revolutas, o que lhe confere seis graus de liberdade. Ao controlador encontra-se conectado um terminal de vídeo com uma unidade de disco e um *teach-pendant* [Unimation, 83]. A linguagem usada é o VAL.

O *teach-pendant* permite o comando manual do robô, facilitando o ensino de uma determinada tarefa através da memorização de alguns pontos da trajectória pretendida para a execução da mesma.

O terminal vídeo possibilita a interacção com o utilizador e o controlo do robô através de comandos VAL. A unidade de disco permite o armazenamento da informação necessária para nova sessão de trabalho do robô, como por exemplo programas e pontos de localização.

A Figura 4.1 apresenta o manipulador, dá indicações sobre o eixo de cada junta e o ângulo varrido por cada uma delas. A posição do manipulador no espaço tridimensional é controlado pelas juntas 1, 2 e 3 em coordenadas cartesianas (X-Y-Z) especificadas em milímetros,

enquanto que as juntas 4, 5 e 6 são especificadas em graus e estabelecem a orientação desejada (O-A-T).

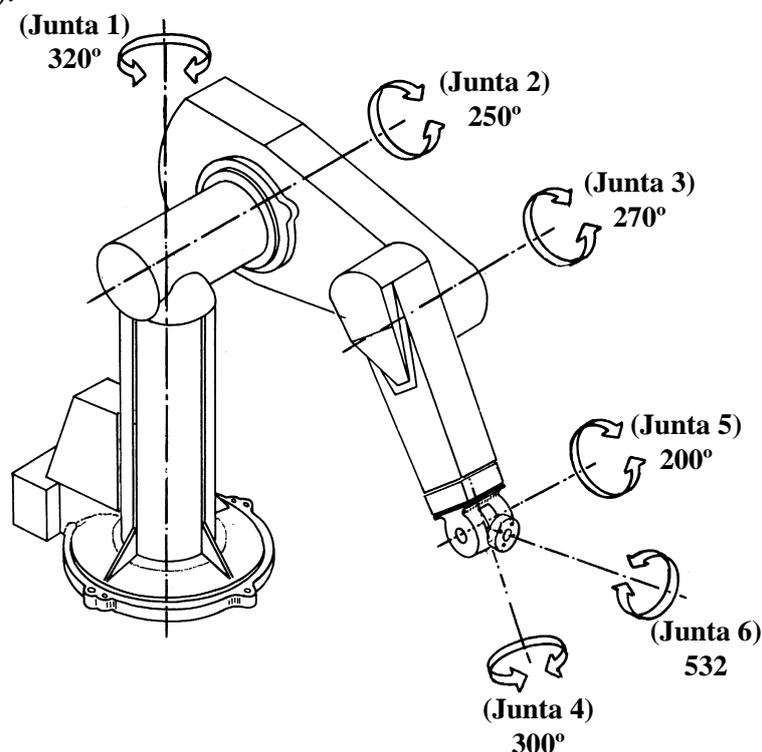


Figura 4.1 O manipulador PUMA 560.

O funcionamento do PUMA 560 requer controlo de posição e velocidade, como é usual em robôs industriais. Para conseguir este objectivo, cada junta é tratada como um servomecanismo, controlado de uma forma independente. Para conhecer a posição e velocidade de cada uma das juntas, existem potenciômetros e codificadores de posição ópticos [Moreira et al., 96]. A arquitectura de controlo original do PUMA 560 é mostrada na Figura 4.2. Devido ao facto de a um pequeno movimento de qualquer junta corresponder algumas revoluções do respectivo codificador de posição, torna-se necessário realizar a calibração sempre que o sistema é reinicializado. Para tal, primeiramente o manipulador começa por movimentar ligeiramente cada uma das juntas de forma a detectar a marca zero de cada um dos codificadores de posição. Quando esta marca é detectada, utiliza-se o valor indicado pelo potenciómetro para estabelecer a posição absoluta. Após efectuada a inicialização, o valor indicado pelo potenciómetro só volta a ser usado numa próxima calibração. O controlador inclui um microcomputador DEC LSI/11 que corre o sistema operativo VAL-II residente em EPROM e RAM onde o programa do utilizador é armazenado. A interacção com o utilizador é possível através do sistema operativo VAL-II. O LSI/11 interpreta as instruções de VAL-II e envia o valor de posição de cada junta para os respectivos controladores. Estes são baseados no microprocessador 6503 da *Rockwell*, com um intervalo de amostragem de 0.875 ms. As referências de posição provenientes do microcomputador, alimentam os controladores em cada 28 ms. Em cada um deles, existe uma malha de realimentação da posição e velocidade proveniente do codificador de posição, que, combinado com as respectivas referências calcula os sinais a fornecer ao amplificador de potência, previamente convertidos em sinais analógico por um conversor D/A. O algoritmo de controlo implementado em cada junta é do tipo proporcional+integral+derivativo (PID).

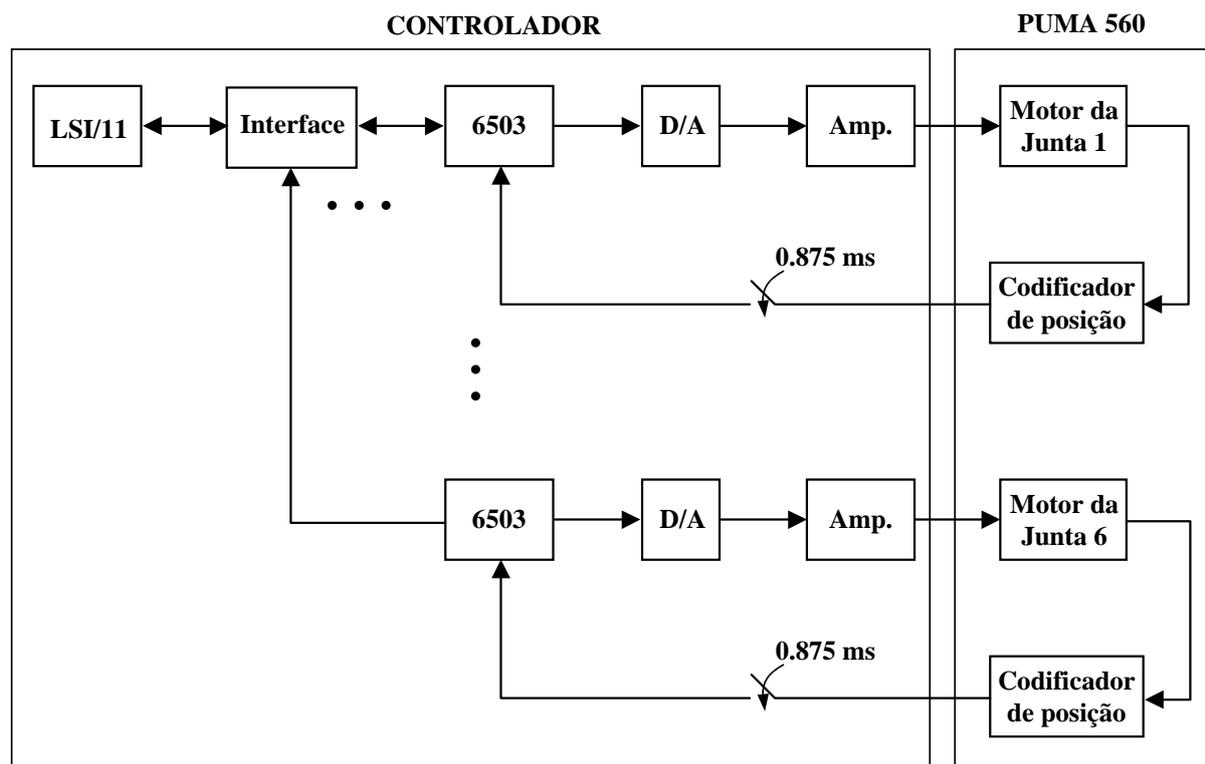


Figura 4.2 Arquitectura de controlo original do PUMA 560.

Esta arquitectura, gerida por um sistema operativo fechado (VAL-II) apresenta algumas dificuldades quando usada em tarefas de controlo de alto nível, devido essencialmente a:

- Os controladores das juntas apresentam ganho fixo tornando impossível a sua alteração ou a alteração do algoritmo de geração de trajectórias, fazendo com que o desempenho não seja idêntico para diferentes velocidades e/ou cargas e/ou configurações do braço.
- Determinados recursos para sensorização não são suportados como por exemplo para visão e força.
- A comunicação entre um PC e o controlador MARK II era somente possível através da porta série a uma velocidade máxima de 9600 bps. Na versão seguinte (MARK III) deste controlador já existe um porto ALTER, que permite a realização de controlo em tempo real.

A solução adoptada para evitar as limitações anteriormente referidas, foi a substituição integral do sistema operativo VAL-II e de parte considerável do *hardware* de controlo do manipulador (Figura 4.3). O novo *hardware* instalado permite o acesso directo à posição das juntas substituindo o anterior controlador. Baseia-se nas placas TRC041, TRC004 e TRC100 desenvolvidas pela *Trident Robotics*. As TRC041 são inseridas no barramento do controlador do PUMA de forma a extrair todos os sinais necessários a um controlo de baixo nível do mesmo. A TRC004 é uma placa genérica para interface em aplicações com servo-controladores [Trident Robotics, 94b] (anexo II). Esta placa recebe os sinais provenientes da TRC041 fazendo o respectivo interface com a TRC100. A TRC100 opera inserida no barramento ISA de um computador pessoal (PC), mas tem ainda possibilidade de funcionar isolada ou num barramento EISA de um PC. É uma placa genérica com base num processador RISC para aplicações com servo-controladores e aquisição de dados [Trident Robotics, 94a] (anexo II). Na Figura 4.16 é possível observar as ligações entre estas placas assim como a sua instalação.

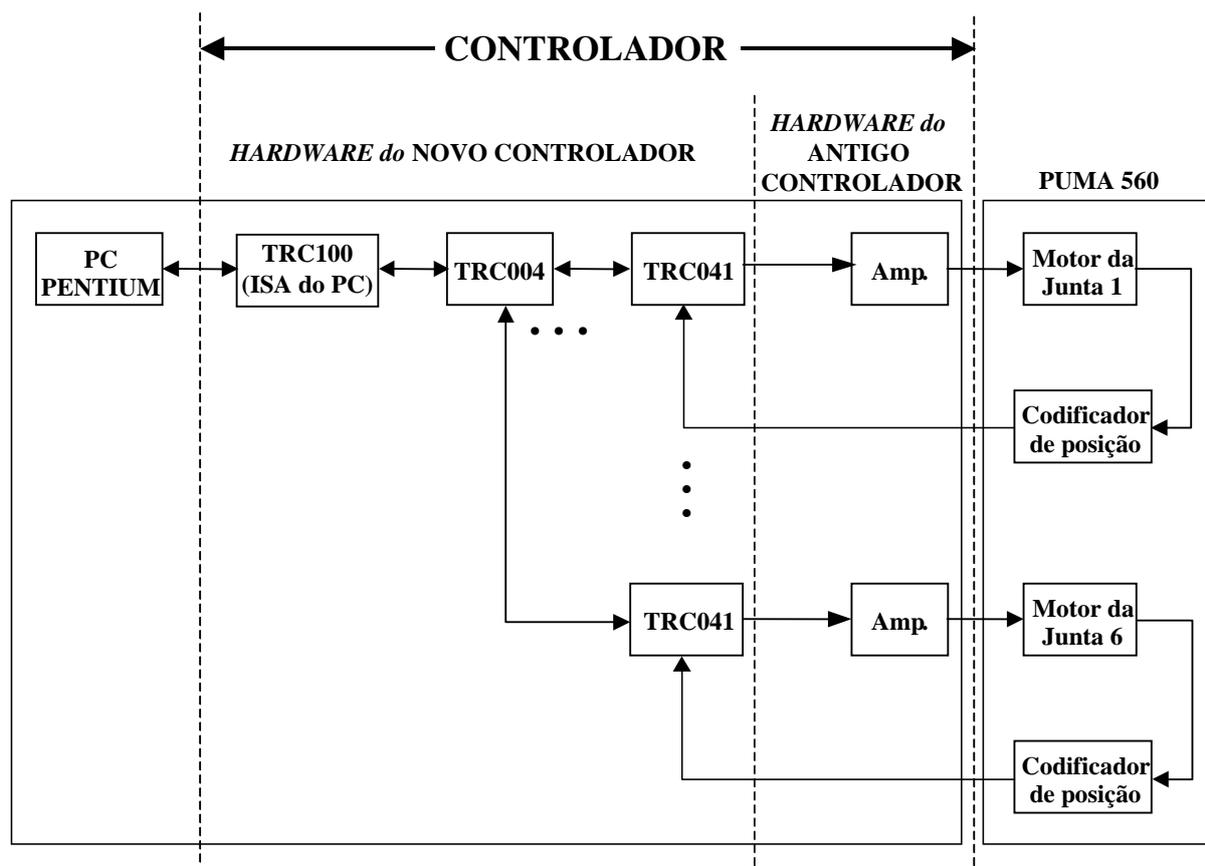


Figura 4.3 Nova arquitectura do controlador do PUMA 560.

A interface entre o TRC100 e o TRC004 é realizada por *software* desenvolvido por a Trident Robotics: uma DLL (*dynamic link library*) contendo funções que permitem a calibração (Potcal e Calibrate), funções de localização (Here e Where), funções de movimento (Speed, Ready, Move, Moves, dMove e Teach) e funções de garra (openi, closei e relax); programa existente na EPROM da placa TRC100. Contudo, devido a alguns problemas no código existente na DLL, nomeadamente, nos cálculos da cinemática directa e inversa, foi necessário o desenvolvimento/correção de rotinas que efectuassem essas tarefas. Como apoio a esta implementação foram utilizados os textos de Craig [Craig, 89], Fu [K. S. Fu et al., 87], Corke [Corke, 94] e Moreira [Moreira et al., 96]. Foi ainda utilizada a *Robotic Toolbox for use with MATLAB* [Corke, 96] para realização dos testes.

Para facilitar a execução de determinadas tarefas ao robô foram adicionadas algumas funções, como por exemplo o Draw e o Base com características idênticas às das suas congêneres do VAL II.

4.3 Sensores

Os sensores são dispositivos que permitem ao robô a “percepção” do ambiente que o rodeia. Esta percepção é feita através da observação de grandezas físicas, que podem ir da simples medição de distâncias à interpretação relativa dos níveis de iluminação ou de cor numa ou mais imagens de vídeo ou de infravermelhos.

Os sensores mais utilizados em robótica podem ser divididos em duas categorias principais: de estado interno e de estado externo. Os sensores de estado interno estão associados à quantificação de variáveis internas, sendo utilizados no controlo do robô. Os potenciômetros, os *resolvers* e os *encoders* são exemplos desse tipo de sensores. Os sensores

externos são os dispositivos que percebem o ambiente exterior. Alguns exemplos destes sensores são: os sensores de binário, de proximidade, de distância, de força, de visão e de temperatura.

Os sensores externos podem ainda classificar-se em sensores com e sem contacto. Na primeira categoria estão incluídos sensores como os de tacto, de deslizamento e de binário; na segunda categoria incluem-se os sensores de visão, acústicos e electromagnéticos. É de salientar que usualmente em robótica os sensores sem contacto são normalmente utilizados para obter uma primeira informação acerca do ambiente e os com contacto estão normalmente associados a estágios terminais (no caso de um robô manipulador estão associados a tarefas de manipulação ou processamento de objectos).

Neste trabalho foram utilizados sensores sem contacto nomeadamente, sensor de visão (câmara CCD), sensores de ultrassons e de infravermelhos, que serão descritos com mais pormenor nos pontos seguintes.

4.3.1 Sensor de visão

O sistema de visão é composto por uma câmara de vídeo CCD (*Charge Coupled Devices*) monocromática, modelo CC-510C da *National Electronics* com uma resolução de 500x582 pontos e uma placa digitalizadora *Video Blaster SE 100* da *Creative Labs* colocada no barramento EISA do PC. A placa possibilita a apresentação de imagens em janelas até ao tamanho total do monitor, recorrendo a modificações de escala do sinal de entrada. Dispõe de três entradas de vídeo comutáveis e um *frame buffer* de 768KB mapeados num espaço de endereçamento de 1MB acessível pelo CPU num formato linear, que deverá estar acima da memória do sistema. O *frame buffer* desta placa utiliza uma configuração YUV [Monteiro, 97] [Creative Labs, 92]. Este formato define o Y como sendo a luminância e UV os valores da crominância azul e vermelha, respectivamente. A relação com o espaço de cor RGB é obtida por:

$$\begin{aligned} Y &= 0.301 \times R + 0.586 \times G + 0.113 \times B \\ U &= -0.301 \times R - 0.586 \times G + 0.887 \times B \\ V &= 0.299 \times R - 0.586 \times G - 0.113 \times B \end{aligned} \tag{4.1}$$

Nestas expressões todas as quantidades têm uma resolução de 7 bits.

A câmara de vídeo foi fixada sobre o ombro do robô. A sua função é procurar o objecto através da comparação das imagens adquiridas na região de trabalho definida por uma mesa semi-circular (Figura 4.4). Após a localização do objecto, é adquirida a informação bidimensional necessária para se delimitar uma região de interesse - região ocupada pela superfície visível do objecto. Esta informação permite um primeiro posicionamento do robô para iniciar a obtenção da terceira dimensão através da utilização de sensores de distância.

Devido à distância entre a peça e a câmara ser variável, tornou-se impossível retirar a real dimensão da mesma. Para resolver esta limitação, a informação da câmara foi complementada com dados provenientes dos sensores de distância, de forma a delimitar com maior exactidão a região de trabalho do robô.



Figura 4.4 Visualização da área de trabalho com todos os sensores montados no PUMA 560.

4.3.2 Sensores de ultrassons

O princípio básico da medição de distâncias com sensores de ultrassons baseia-se no registo da diferença de tempo que ocorre entre a transmissão de uma onda sonora emitida e a recepção do seu eco. A utilização deste tipo de sensores em aplicações de robótica está largamente difundida, devido essencialmente ao seu baixo custo quando comparado com medidores de distância à base de laser.

Contudo, muitos trabalhos prévios foram desencorajadores como por exemplo os de Flynn [Flynn, 88], Beckerman [Beckerman and Oblow, 90], Leonard [Leonard and Durrant-White, 92] e Manyika [Manyika and Durrant-White, 94] que concluíram que o trabalho com ultrassons encerra grandes dificuldades devido essencialmente a estes possuírem um feixe com um grande ângulo de abertura, o que implica uma pobre resolução angular. Esta característica provoca a obtenção de medidas erradas se a onda emitida não for normal à superfície do objecto reflector. A Figura 4.5 mostra que a distância entre o sensor e a parede deveria ser \overline{OA} . Contudo, a distância detectada será $\overline{OA'}$, porque o primeiro eco devolvido detecta este [Song and Tang, 96]. A posição medida neste caso é então dada por A' .

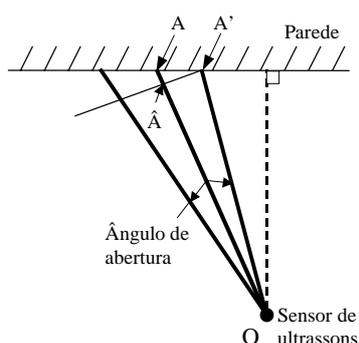


Figura 4.5 Influência do ângulo de abertura do feixe quando é efectuado o varrimento de uma parede com um sensor de ultrassons.

Por este motivo, existe uma diferença entre a localização da parede percebida e a sua localização real. Este tipo de erros provocados por um ângulo de abertura grande afecta a exactidão com que o ambiente é percebido. Por outro lado, se o ângulo de incidência for superior a um valor máximo (aproximadamente 25° para o transdutor electrostático da Polaroid), poderá nunca ser recebido qualquer eco ou ser recebido após reflexões sucessivas, o que leva à obtenção de medidas erradas (Figura 4.6).

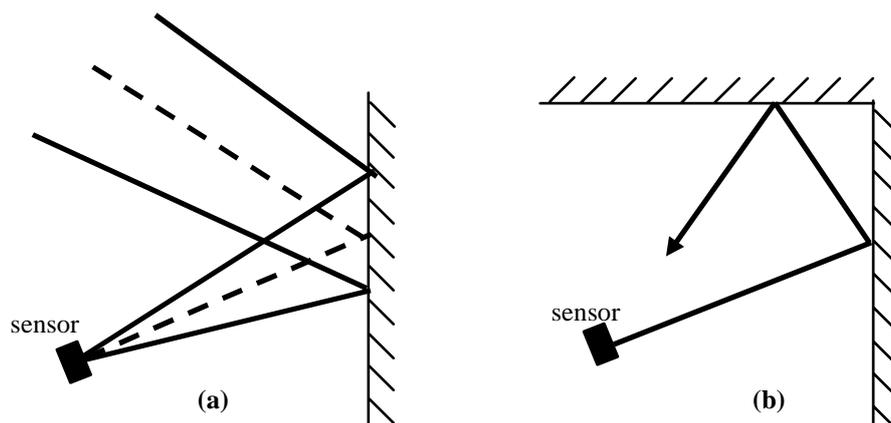


Figura 4.6 Reflexão resultante de um ângulo de incidência grande: (a) nenhum eco é recebido pelo sensor; (b) o eco é recebido após reflexões sucessivas.

Acerca deste assunto diversos autores fizeram os mais diversos comentários, como por exemplo:

- Os sensores de ultrassons apresentam muitas deficiências...: 1. pobre direccionalidade o que limita a exactidão na determinação da posição espacial de uma fronteira na gama de 10-50 cm, dependendo da distância ao obstáculo e do ângulo entre a superfície do obstáculo e o feixe acústico; 2. Frequentes leituras erradas...; 3. Reflexões especulares que ocorrem quando o ângulo entre a frente da onda e a normal a uma superfície plana é grande [Borenstein and Koren, 89].
- Os dados provenientes de ultrassons são seriamente corrompidos por reflexões sucessivas [Crowley, 89].
- ... o uso de ultrassons representam de certa forma a pior opção para efectuar localizações a partir de dados [Drumheller, 87].

Comentários semelhantes a estes abundam na literatura, sendo os ultrassons caracterizados como problemáticos, imprevisíveis e não confiáveis. Contudo, apesar destas conclusões, os sensores de ultrassons apresentam algumas vantagens para além do seu baixo custo que tornam aliciante a sua utilização:

- A tecnologia dos ultrassons é conhecida e dominada há bastante tempo;
- Não apresenta qualquer risco de segurança para os seres humanos;
- Em circunstâncias adequadas (superfície reflectora normal ao feixe e dentro da gama) a informação proveniente deste tipo de sensores é bastante precisa;
- São relativamente fáceis de controlar;
- O sinal refletido é um sinal rico em informação contendo, designadamente, informação relativa à orientação da superfície reflectora.

Modelo físico dos ultrassons

O método de medição de distâncias por ultrassons mais comum é baseado na contagem de tempo que um sinal sonoro demora a percorrer a distância entre o emissor e o receptor. Este intervalo de tempo é designado por “tempo de voo” (TOF – *Time-of-Flight*). No caso

estudado o emissor funciona também como receptor, existindo um tempo de vazio entre as duas situações em que não é possível a detecção do eco (Figura 4.7).

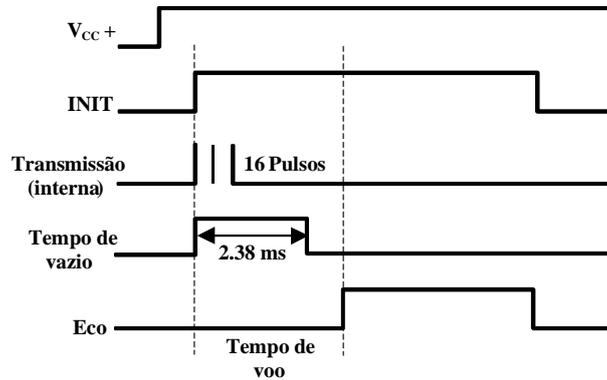


Figura 4.7 Diagrama de funcionamento dos ultrassons.

O tempo de voo medido é relativo ao dobro da distância entre o transmissor e a superfície reflectora, determinado pela expressão [Maslin, 83]:

$$d = \frac{1}{2} \times c \times TOF \tag{4.2}$$

onde,

$$c = 331.4 \sqrt{\frac{T + 273}{273}} \text{ m/s} \tag{4.3}$$

T – temperatura ambiente em graus Centígrados c – velocidade do som no ar.

É fácil constatar a partir da relação dada pela equação 4.3, que a velocidade do som no ar é afectada com relevância pela temperatura. Se pretendermos valores mais exactos da distância então a temperatura terá que ser conhecida de forma a que a sua influência possa ser compensada. Os factores principais que afectam a eficiência de um sistema baseado em ultrassons estão relacionados com o desempenho dos próprios sensores, a frequência de operação, e a gama de distância desejada. Sem ter em consideração a electrónica de processamento, existem ainda dois aspectos que não se podem descuidar quando se trabalha com sistemas de ultrassons: o primeiro tem a ver com a relação entre o tamanho do transdutor, o ângulo de abertura do feixe e a frequência de operação; o segundo com a relação entre a gama de distância desejada e a frequência de operação.

Para obtermos a primeira relação, é comum usar um modelo do tipo pistão para analisar este tipo de sensores. Este modelo considera que o emissor vibra harmoniosamente e de forma uniforme ao longo de toda a superfície do sensor, sendo o padrão da radiação descrito em termos de funções de Bessel pela seguinte expressão [Maslin, 83]:

$$P(\theta) = \frac{2 J_1(ka \sin(\theta))}{ka \sin(\theta)} \tag{4.4}$$

onde, $k = \frac{2\pi}{\lambda} = 2\pi \frac{f}{c}$

a = raio do sensor (pistão)

θ = ângulo de azimute

J_1 = Função de Bessel da primeira espécie de 1ª ordem.

Com o objectivo de se obter o ângulo θ em função da frequência e do tamanho do sensor, é comum definir este como o ângulo que intercepta os pontos de ambos os lados do eixo principal onde o padrão da radiação é $-3dB$ relativamente ao seu valor no eixo principal ($\theta = 0$).

Estabelecendo $x = ka \sin(\theta)$ e substituindo na equação 4.4 obtemos,

$$\frac{2J_1(x)}{x} = -3dB = \frac{1}{\sqrt{2}} \quad (4.5)$$

Para resolver a equação 4.5 é necessário utilizar um procedimento iterativo, sendo uma boa aproximação a obtida através da expansão da função de Bessel até 3 termos:

$$x = 1.62 \quad \text{como} \quad x = ka \sin(\theta) \quad (4.6)$$

então,

$$\theta = \sin^{-1}\left(\frac{x}{ka}\right) = \sin^{-1}\left(\frac{1.62}{ka}\right) = \sin^{-1}\left(\frac{1.62\lambda}{2\pi a}\right) = \sin^{-1}\left(\frac{1.62c}{2\pi fa}\right). \quad (4.7)$$

A largura de banda é igual a 2θ . O padrão do campo de radiação em função de θ é apresentado na Figura 4.8 [Polaroid, 95] para o transdutor electrostático série 600 da *Polaroid Corp.* do tipo pistão usado neste trabalho.

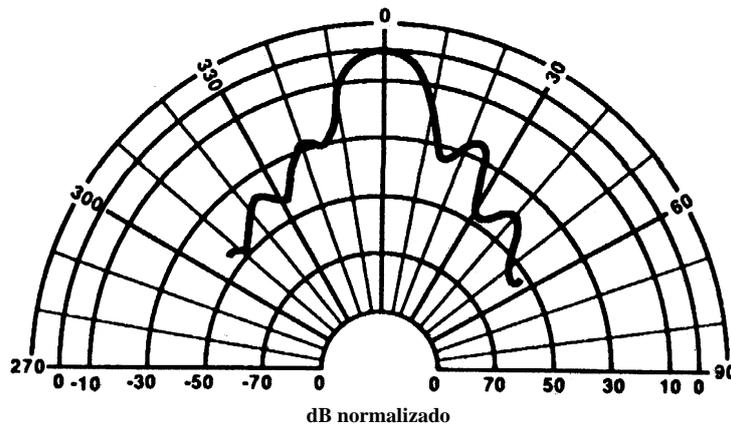


Figura 4.8 Padrão do campo de radiação para o transdutor electrostático série 600 da *Polaroid Corp.* com os parâmetros: $c = 343.2$ m/s, $a = 19$ mm e $f = 50$ kHz.

Do diagrama de radiação pode confirmar-se que a potência de emissão decai de $3dB$ para um azimute de cerca de 5.4° , definindo-se assim um ângulo de abertura (largura de banda) de cerca de 11° dentro do qual a energia emitida será eventualmente suficiente para, após uma reflexão favorável, provocar uma detecção. Contudo, para este sensor é usual estabelecer-se um θ_{max} aproximadamente igual a 22° , ou seja o ângulo de abertura inclui não só o lóbulo central como também parte dos lóbulos secundários.

Segundo Barshan e Kuc [Barshan and Kuc, 90] a distribuição do campo acústico produzido por um emissor apresenta um padrão constituído por duas regiões distintas: a região de campo próximo ou zona de *Fresnel* e a região do campo afastado ou zona de *Fraunhofer*, como se apresenta na Figura 4.9. Na região do campo próximo, o feixe está contido num cilindro de diâmetro $2a$, sendo a o raio do sensor. Esta zona estende-se desde a

face do sensor até uma distância aproximadamente igual a $\frac{a^2}{\lambda}$. Na zona de campo afastado e de acordo com a teoria da difração, o diâmetro exterior do feixe diverge com um ângulo $\theta = 2\theta_0$, com

$$\theta_0 = \sin^{-1}\left(\frac{0.61 \lambda}{a}\right) \tag{4.8}$$

O ângulo θ_0 corresponde ao primeiro mínimo da função de Bessel na expressão (4.4) [Nunes, 95].

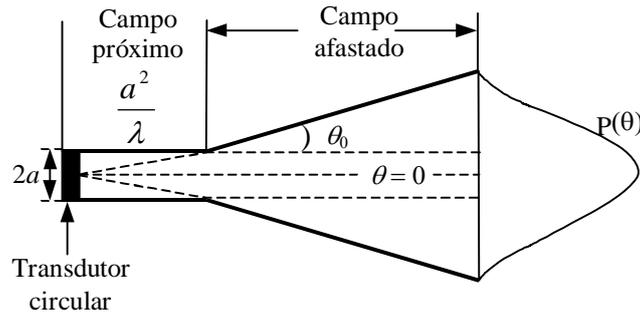


Figura 4.9 Distribuição do campo acústico produzido por um modelo do tipo pistão. O ângulo θ_0 é designado ângulo de divergência.

Na segunda relação, a gama máxima mensurável para um transdutor de ultrassons depende da frequência de operação devido à sua relação com a atenuação do som no ar. A Figura 4.10 [Polaroid, 95] mostra que, à medida que a frequência de operação aumenta também aumenta a atenuação da onda acústica no meio de propagação e, por conseguinte, se a energia do sinal emitido se mantiver o alcance do sensor diminui significativamente.

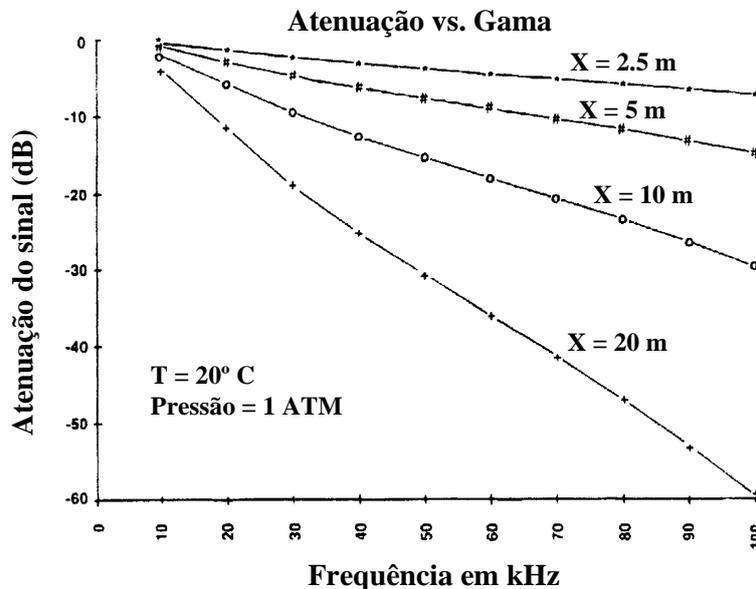


Figura 4.10 Relação entre a frequência e a atenuação no ar.

Soluções adoptadas

Para aquisição do relevo foi usado um *kit* da *Polaroid Corp. (Ultrasonic Ranging Developer's Kit)*. Os principais componentes deste *kit* são um transdutor electrostático da série 600 (Figura 4.11), que funciona como emissor e receptor, e um módulo electrónico para processamento com facilidades de comunicação através do protocolo RS-232. Este último subdivide-se na parte digital e na parte analógica [Polaroid, 95].

A parte digital baseia-se no microprocessador Intel 80C196. Este corre *software* existente em EPROM que permite por exemplo: a definição do número de pulsos a transmitir (por defeito são 16), a frequência de operação (por defeito é 52 kHz) e o tempo de vazio (por defeito 2.38 ms) (anexo III).

A parte analógica é quase totalmente controlada por o circuito integrado *Polaroid* (com a referência 614909). Este permite: um ajuste global do ganho a partir de uma resistência externa (potenciómetro); a selecção de um ganho exponencial controlado digitalmente por 4 saídas digitais do 80C196 num máximo de 12 patamares; trabalhar com frequências entre os 20 kHz e os 90 kHz. Na parte analógica ainda é possível seleccionar um limite mínimo (entre 1.2 V e 2 V) através de uma resistência variável, a partir do qual a detecção do eco é possível. Reduzindo esse limite ao mínimo incrementa a sensibilidade na detecção do eco, mas torna-o igualmente mais susceptível ao ruído acústico e eléctrico.

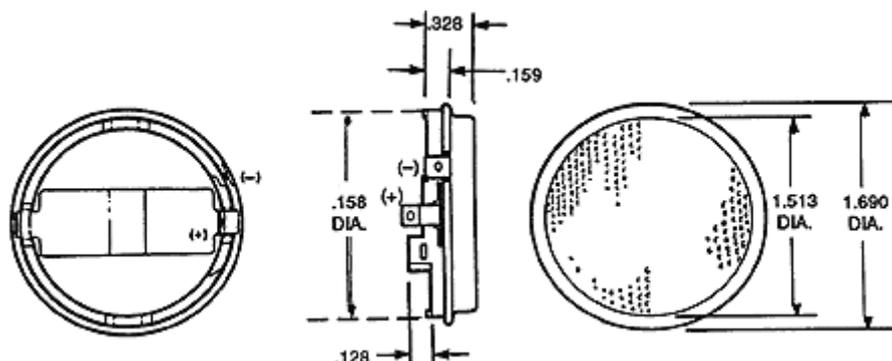


Figura 4.11 Transdutor electrostático.

O princípio de funcionamento deste sistema baseia-se na transmissão de pulsos ultrassónicos até ao alvo. Após o último impulso, o circuito comuta de função e entra na fase de espera do eco (Figura 4.12). A partir deste momento, o circuito vai acumulando a energia devolvida pelos pulsos transmitidos, até ao momento em que regressou a energia suficiente para se declarar uma detecção. O tempo decorrido entre o início do pulso transmitido e a recepção do eco (TOF) é medido. Conhecendo a velocidade do som no ar, o sistema converte o tempo decorrido na distância medida, como já anteriormente referido.

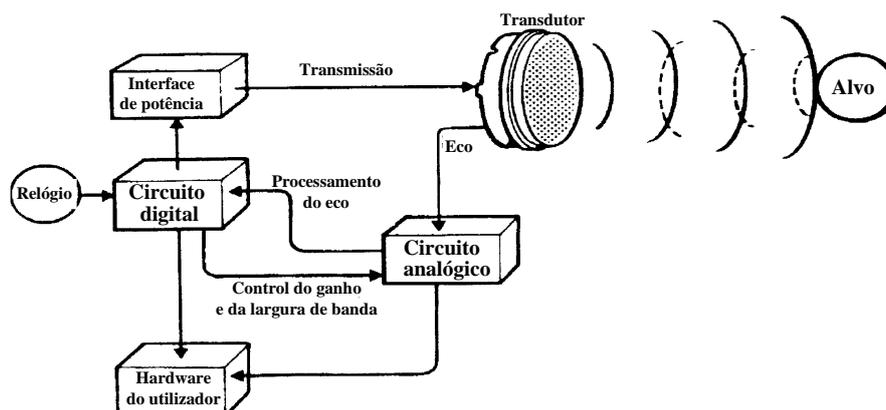


Figura 4.12 Diagrama de blocos do circuito da Polaroid para o sensor de ultrassom.

Com o objectivo de minimizar os problemas colocados pelos ultrassons anteriormente referidos tomaram-se as seguintes opções:

1. Colocação de um tubo com cerca de 20 cm em frente de cada um dos sensores (Figura 4.13).
2. Aumentar a frequência de operação de 50 kHz para 63 kHz.
3. Utilização de 8 pulsos em vez de 16 e diminuição do tempo de vazio de 2.38 ms para 1.38 ms.
4. Ajuste do ganho global e exponencial assim como do limite mínimo para a detecção do eco no módulo electrónico.

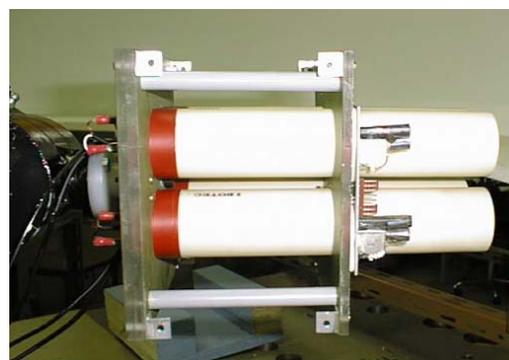


Figura 4.13 Posição dos tubos relativamente aos sensores e a sua colocação na parte terminal do robô.

As primeiras duas opções tiveram como objectivo aumentar a zona de *Fresnel* e por conseguinte obter uma diminuição do ângulo de abertura na gama de operação pretendida. O aumento da frequência implica ainda que o sistema fique a trabalhar com uma melhor resolução angular. Contudo, esse aumento provoca uma maior atenuação da onda no ar diminuindo desta forma o valor de distância máxima detectável. Neste caso particular, a atenuação verificada não originou qualquer problema porque o valor máximo pretendido nunca necessita ultrapassar os 80 cm, enquanto que o valor máximo medível com esta configuração pode ir até aproximadamente 2 m (valor obtido na prática com o alvo normal ao feixe).

A terceira opção teve como objectivo diminuir o valor mínimo detectável de 40 cm para 25 cm. Esta solução possibilitou: uma maior aproximação entre os ultrassons e a superfície da peça a ser varrida, tendo como consequência uma redução do ângulo de divergência (maior

direccionalidade); e aumentar o número de medidas efectuadas por segundo tornando o processo de aquisição do relevo mais rápido.

Finalmente, os ajustes efectuados na última opção tiveram como objectivo fazer com que o eco seja detectado na gama de medida pretendida e que o aumento do ruído (acústico e eléctrico) seja reduzido ao mínimo.

Para além dos problemas já descritos, há um outro problema associado ao uso simultâneo de múltiplos sensores, que é a interferência mútua (*cross-talk*). Este problema foi resolvido por multiplexagem, ou seja, unicamente é permitido o disparo de um dos ultrassons num determinado instante. Este processo aliado à baixa velocidade do som tornou a aquisição do relevo um processo relativamente demorado.

4.3.3 Sensores de infravermelhos

Os sensores de infravermelhos são usados para percepção de pequenas distâncias (até cerca de 1 metro). As primeiras versões deste tipo de sensores geravam informação de natureza binária, do tipo, existe um obstáculo a uma distância inferior a um dado limite (especificação do sensor) ou não existe obstáculo algum. Não eram por isso verdadeiros medidores de distâncias, mas sim detectores de proximidade. As últimas evoluções introduziram a medição de distâncias, mas sempre dentro de uma gama de medida restrita. Com a perspectiva de poderem vir a ser utilizados neste trabalho, foi testado um díodo emissor de luz (*Light emitting diode-LED*) comum como fonte de luz e um receptor da SHARP série GP1U58X, tendo sido desenvolvida a electrónica de processamento.

O GP1U58X funciona usualmente como detector, tendo por conseguinte uma saída digital. Esse sinal é obtido a partir do *Schmitt trigger* que está em série com um filtro passa banda de 40 kHz e um amplificador de sinal (Figura 4.14). Com o objectivo de efectuar a tarefa pretendida, o GP1U58X foi modificado de acordo com Doty [Doty and Iglesia, 96], de forma a ser possível a medição de distâncias numa gama entre os 10 cm e os 40/50 cm, o que corresponderia a uma variação em tensão entre os 1.8 V e os 2.6 V.

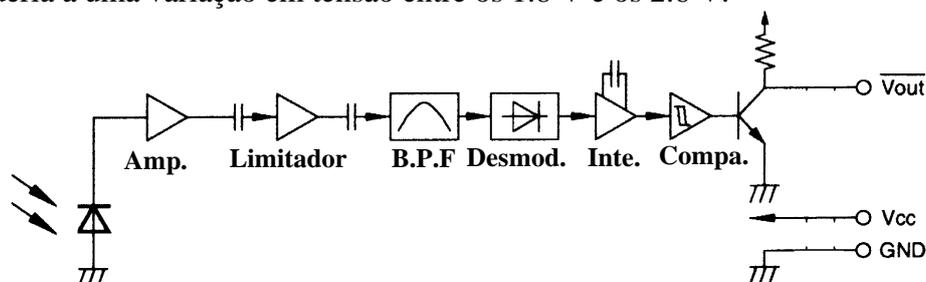


Figura 4.14 Diagrama de blocos interno do GP1U58X.

A electrónica de processamento baseia-se no 555 que gera um sinal com uma frequência de 40 kHz para atacar o díodo emissor com pulsos de corrente (Figura 4.15). A utilização destes, deve-se ao facto que o alcance dos LEDs depende mais da amplitude dos pulsos do que do seu valor médio. Um amplificador diferencial foi usado para processar o sinal proveniente do GP1U58X, possibilitando uma saída entre os 0 V e os 4.5 V. A saída do amplificador diferencial foi conectada à entrada de um conversor analógico-digital (A/D) para posterior processamento.

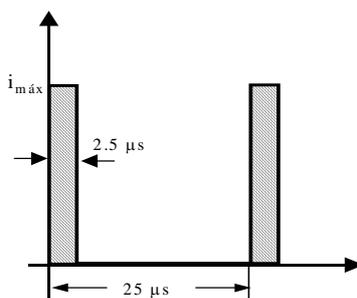


Figura 4.15 Forma de onda aplicada ao diodo emissor.

Aproveitando a normalmente boa resolução angular dos infravermelhos, a sua função seria a detecção com mais precisão do ponto onde aconteciam as variações de profundidade. O mesmo é difícil de conseguir com ultrassons devido às limitações anteriormente referidas. Contudo, a utilização desta solução apresentou alguns problemas que motivaram o abandono da utilização dos infravermelhos:

- Fraca precisão de medida, apresentando frequentemente erros de 3 e 4 cm, tendo ainda valores superiores se a superfície reflectora não fosse perpendicular ao feixe.
- Grande sensibilidade à propriedade reflectora dos materiais (cor, brilho), tornando difícil a sua calibração uma vez que se pretende a utilização de peças com diferentes características.
- A implementação de um processamento mais cuidado não originou melhorias significativas no seu desempenho.
- A melhoria do *hardware* e a sofisticação do processamento para os ultrassons, traduziu-se, em que a utilização dos infravermelhos não reflectisse nenhuma melhoria nos resultados obtidos por estes, agravando-os até em determinadas situações.

4.4 Rede CAN

Dadas as limitações dos actuais robôs industriais em integrarem informação sensorial como já anteriormente referido, tornou-se necessário proceder à utilização de estruturas de *hardware* e *software*, com características adequadas à integração de sensores num ambiente robotizado. Para este fim adoptou-se a utilização do CAN, que é um protocolo de comunicação série que suporta eficientemente controlo distribuído em tempo real. O CAN foi originalmente desenvolvido na Alemanha pela companhia *Bosch* para a indústria automóvel. O seu domínio de aplicação é inserido nas redes de alta velocidade com um baixo custo de multiplexagem de cabos. Com o objectivo de se obter compatibilidades entre quaisquer implementações CAN, a *Bosch* escreveu um conjunto de especificações. Contudo, a compatibilidade, tem pontos de vista diferentes no que diz respeito por exemplo às características eléctricas e interpretação dos dados a serem transferidos.

Com a finalidade de garantir transparência e flexibilidade na implementação dividiu-se o CAN em três camadas, segundo o modelo de referência ISO/OSI [Bosch, 91]:

1. Camada física.
2. Camada de ligação ou transferência. Esta camada é normalmente ainda dividida em duas subcamadas: MAC (*Media Access Control*) e LLC (*Logical Link Control*).
3. Camada de aplicação.

É ainda de salientar que a especificação CAN ISO 11898 refere-se somente às primeiras duas camadas.

Camada Física

Específica as características eléctricas e físicas do barramento, o *hardware* que efectua a conversão dos caracteres de uma mensagem em sinais eléctricos nas mensagens a transmitir e sinais eléctricos em caracteres nas mensagens recebidas. Devido aos requerimentos de estabilidade o comprimento máximo da rede está directamente relacionado com a taxa de transmissão. Alguns valores usuais são:

Taxa de transmissão	Comprimento máximo da rede
1 Mbit/s	30 m
800 kbit/s	50 m
500 kbit/s	100 m
250 kbit/s	250 m
125 kbit/s	500 m
62.5 kbit/s	1000 m
20 kbit/s	2500 m
10 kbit/s	5000 m

Camada de Ligação

É ao nível desta camada que são reconhecidos e interpretados os formatos das mensagens. Esta camada “constrói” as mensagens que são transmitidas pela camada física e descodifica as mensagens recebidas a partir da mesma.

Nos controladores CAN, a camada de ligação é usualmente implementada em *hardware* devido à sua complexidade. Como é comum com outras redes a camada de ligação é subdividida em 2 subcamadas: a subcamada LLC e a subcamada MAC. Os serviços fornecidos pela LLC são os seguintes: filtragem para aceitação de mensagens, notificação de sobrecarga no barramento e gestão da recuperação de erros. Os serviços fornecidos pela subcamada MAC são os seguintes: codificação e envio das mensagens, descodificação das mensagens recebidas, gestão das mensagens prioritárias, envia confirmações (*acknowledgement*), detecção e sinalização de erros e estabelece a interface de comunicação com a camada física.

Camada de Aplicação

Muitas aplicações de CAN requerem serviços que estão fora do alcance da camada de ligação definida pela especificação ISO 11898, mas que podem ser implementadas na camada de aplicação. Algumas opções existem para a camada de aplicação: CAL (*CAN Application Layer*), CANOpen, PCAL (*Portable CAN Application Layer*), DeviceNet, SDS (*Smart Distributed System*) e CAN Kingdom.

Na implementação física da rede usada no trabalho foram utilizadas 2 placas da STZP (*Steinbeis Transferzentrum Prozessautomatisierung*):

1. PC/CAN-Interface PCI 02 – Trabalha no barramento ISA do PC. Baseia-se no controlador CAN da *Philips* PCA82C200, podendo operar separadamente ou em simultâneo duas redes, uma vez que possui dois controladores deste tipo. A frequência de

operação do controlador CAN é de 16 MHz [STZP, 95] (anexo IV). O interface do barramento CAN obedece à especificação ISO 11898.

2. Universal CAN I/O Board – Esta placa baseia-se na utilização do microcontrolador da *Philips* 80C592 da família 80C51, mas com controlador CAN incorporado e trabalhando a uma frequência de 16 MHz. A memória endereçável é de 64 K podendo ser ocupada por uma EPROM (64 K), RAM (64 K) ou ambas tendo neste caso 32 K cada. Esta placa disponibiliza ainda 16 entradas e saídas digitais, 8 entradas analógicas e ainda duas saídas PWM [STZP, 93a] (anexo IV).

O meio físico da rede obedece à norma da CiA (*CAN in Automation*), sendo composto por uma linha de 2 fios com um retorno comum e uma terminação em ambas as extremidades constituído por resistências compatíveis com a impedância característica da linha. A transmissão é efectuada de uma forma diferencial.

Para implementação da comunicação foi fornecido pela STZP uma biblioteca (DLL - *dynamic link library*) de rotinas que possibilita a transferência de dados numa rede CAN que inclua a placa PCI 02 [STZP, 93b]. O *software* de comunicação sobre a rede CAN para a Universal CAN I/O foi desenvolvido de raiz.

É ainda de referir que a comunicação é idêntica para todas as aplicações CAN. Contudo, existem duas configurações básicas no que diz respeito ao *hardware*. O BasicCAN e o FullCAN. No BasicCAN existe uma estreita ligação entre o controlador CAN e o microcontrolador associado. Nesta configuração o microcontrolador poderá ser interrompido para atender as mensagens que chegam via rede. No FullCAN existe *hardware* adicional que gere automaticamente as mensagens a transmitir e a receber sem interromper o microcontrolador associado. A utilização de *hardware* baseado num microcontrolador com controlador CAN incorporado impôs a configuração BasicCAN. Por outro lado os serviços que a rede teria que fornecer não justificavam uma configuração FullCAN.

4.5 Sumário

A Figura 4.16 resume o equipamento utilizado ao longo deste trabalho e as suas conexões.

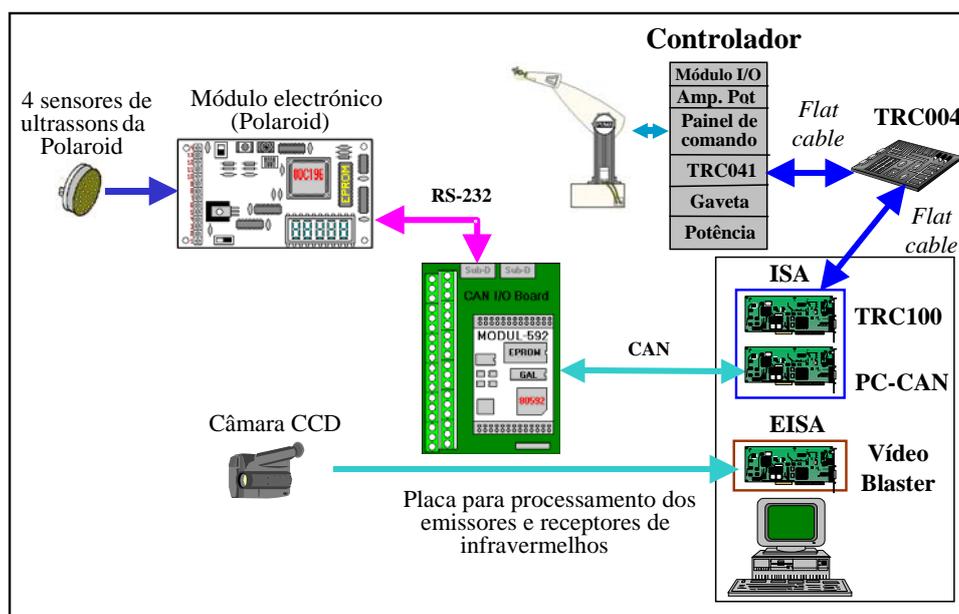


Figura 4.16 Célula de trabalho e suas conexões.

Neste capítulo, foram apresentados com detalhe os diversos elementos que constituem a célula de trabalho e as soluções adoptadas para minimizar algumas limitações ou problemas colocados pelos diversos elementos.

No ponto 4.2 foi efectuado um resumo das diferentes classificações utilizadas em robótica tendo sido apresentado em seguida o manipulador PUMA 560 e enumerados os problemas levantados por a parte de controlo que dificultariam/impossibilitariam o desenvolvimento deste trabalho, tendo sido apresentada igualmente a solução adoptada.

No ponto 4.3 foram apresentados todos os sensores utilizados, suas características, limitações e soluções adoptadas para minimização dessas limitações. Neste ponto foi dado um maior realce aos sensores de ultrassons, apesar dos muitos trabalhos já existentes, devido a que os mesmos se constituíram num dos principais elementos para o desenvolvimento do trabalho.

Finalmente, no ponto 4.4 foi apresentado o protocolo de comunicação CAN, os elementos físicos utilizados para implementação deste protocolo e suas principais características.

Capítulo 5

5. Sistema de aquisição e comunicação de dados

5.1 Introdução

Dadas as limitações do PUMA 560 em integrar informação sensorial, foi necessário a utilização de uma estrutura que possibilitasse a integração de sensores. Optou-se por um modelo distribuído multiprocessador, onde a transferência de dados é efectuada sobre uma rede CAN (*Controller Area Network*). Como já foi referido, o CAN é constituído por duas camadas especificadas pela norma ISO 11898, camada física e a camada de ligação ou de transferência. Para a camada de aplicação existem diversas propostas de especificação, como são por exemplo o CAL (*CAN Application Layer*), CANOpen, PCAL (*Portable CAN Application Layer*), DeviceNet, SDS (*Smart Distributed System*) e CAN Kingdom. A parte física da rede CAN foi implementada com base em duas placas da STZP: a PCI 02 que tem como função a gestão de todas as comunicações via CAN com o PC e a Universal CAN I/O que tem como função a interface com o sistema sensorial enviando posteriormente a informação adquirida para o PC.

Para a camada de aplicação não foi adoptada nenhuma das especificações acima referidas. Estas são tradicionalmente utilizadas em redes com vários nós e onde o volume de mensagens a processar é significativo, o que implica uma estrutura de gestão de mensagens mais complexa. Atendendo a que neste trabalho apenas um único nó é necessário, optou-se pelo desenvolvimento de um protocolo mais simples para esta camada.

Em virtude da sua velocidade de tempo real e tolerância a falhas, mesmo em ambientes ruidosos, o CAN teve uma larga aceitação numa grande variedade de áreas de aplicação, como por exemplo, a indústria automóvel, a agricultura, a robótica e a automação industrial.

Neste capítulo começa-se por descrever as características fundamentais do CAN, tais como o formato das mensagens, mecanismos de detecção de erros e o protocolo básico do CAN. Em seguida é descrito o protocolo implementado para a camada de aplicação.

5.2 O CAN

5.2.1 Formato das mensagens

As mensagens transportam dados a partir dum nó transmissor para um ou mais nós receptores. É ainda de referir que existem duas especificações CAN:

- a especificação padrão (versão 2.0A) que suporta mensagens com 11 bits identificadores no campo de avaliação.
- a especificação estendida (versão 2.0B) que suporta mensagens com 11 ou 29 bits identificadores no campo de avaliação.

Com base nas especificações suportadas distinguem-se três tipos de controladores: os controladores 2.0A que só transmitem e recebem mensagens no formato padrão; os controladores 2.0B passivos que podem receber mensagens no formato estendido, mas ignoram-nas; os controladores 2.0B que podem emitir e receber mensagens em ambos os formatos.

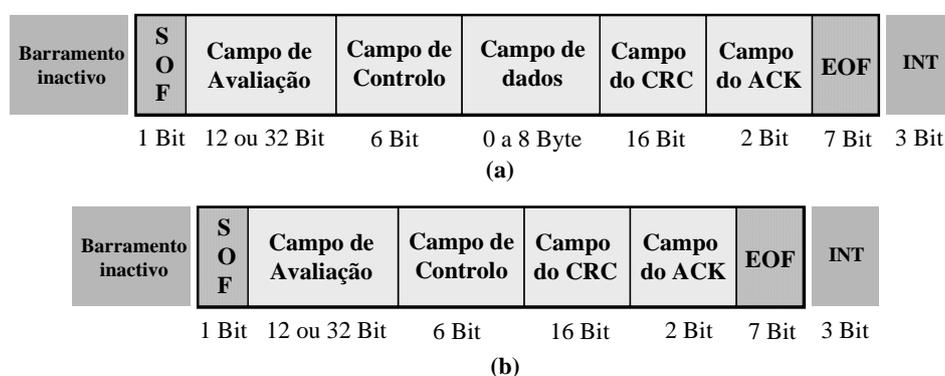


Figura 5.1 Formato dos quadro de mensagens CAN. (a) quadro de dados; (b) quadro remoto.

No CAN é possível utilizar dois tipos de quadros de mensagem (Figura 5.4), o quadro de dados (*data frame*) e o quadro remoto (*remote frame*) [CAN in Automation, 95]. Os quadros remotos são utilizados para um determinado nó solicitar a um outro nó determinado tipo de informação. Esta é distinguida pelo nó solicitado através do valor do identificador (ID) existente no campo de avaliação.

SOF (*Start of Frame*) – É um bit “dominante” (valor lógico 0) que indica o início da transmissão de uma mensagem sincronizando todos os nós da rede.

Campo de avaliação (*Arbitration field*) – A Figura 5.2 apresenta o formato do campo de avaliação para as mensagens CAN nas especificações existentes (quadro padrão e quadro estendido). No quadro padrão o comprimento do identificador (ID) é de 11 *bits* correspondendo ao ID base no quadro estendido. A seguir ao campo ID surge um *bit* RTR (*Remote transmission request*). O RTR é “dominante” quando é transmitido um quadro de dados e é “recessivo” (valor lógico 1) quando é transmitido um quadro remoto. A seguir ao RTR surge um outro *bit* já no campo de controlo denominado IDE (*Identifier extension*). Este *bit* é transmitido como “dominante” no quadro padrão e como “recessivo” no quadro estendido. No caso de colisão o quadro padrão prevalecerá relativamente ao estendido.

No quadro estendido o ID é constituído por 29 *bits* divididos em duas secções: o ID base constituído por 11 *bits* e o ID estendido constituído por 18 *bits*. O *bit* SRR (*Substitute remote request*) substitui o *bit* RTR e é sempre transmitido como “recessivo” para garantir que um

quadro padrão tenha sempre prioridade sobre um quadro estendido. No caso deste *bit* ser corrompido e transmitido como dominante é ignorado pelos receptores.

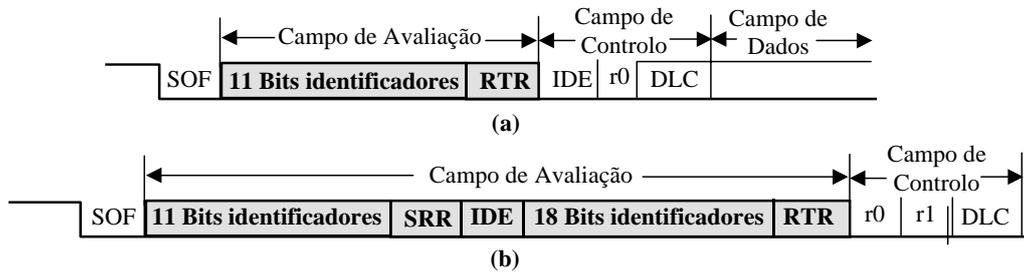


Figura 5.2 Campo de avaliação das mensagens CAN. (a) quadro padrão; (b) quadro estendido.

Campo de controlo (Control field) – O formato do campo de controlo é similar para as duas especificações (Figura 5.3). Para a especificação padrão o formato inclui o comprimento dos dados (DLC – *Data length code*), o *bit* IDE já referido e o *bit* reservado r0, sendo ambos transmitidos como “dominantes”. Na especificação estendida o formato é idêntico ao anterior com mais um *bit* reservado r1, também transmitido como “dominante”. A Tabela 5-1 apresenta a relação entre o número de *bytes* de dados a transmitir e o valor atribuído a cada *bit* que constitui o DLC.



Figura 5.3 Campo de controlo.

Nº de Bytes	DLC3	DLC2	DLC1	DLC0
0	d	d	d	d
1	d	d	d	r
2	d	d	r	d
3	d	d	r	r
4	d	r	d	d
5	d	r	d	r
6	d	r	r	d
7	d	r	r	r
8	r	d	d	d

Tabela 5-1 Relação entre o número de bytes de dados e o valor a atribuir a cada *bit* do DLC (d-dominante, r-recessivo).

Campo de dados (Data field) – O campo de dados é constituído por oito *bytes* como já foi referido (Figura 5.4). É ainda de salientar que este campo não existe no caso dos quadros remotos.



Figura 5.4 Campo de dados.

Campo do CRC (CRC field) – O campo CRC é constituído por 15 *bits* mais um *bit* limitador que é sempre transmitido como “recessivo”.



Figura 5.5 Campo CRC.

Campo do ACK (ACK field) – O campo ACK é constituído por 2 *bits*. O primeiro é um *bit* de espaçamento que é transmitido como “recessivo”, mas é posteriormente reescrito como

Verificação confirmação

É da responsabilidade do transmissor verificar se o *bit* ACK foi reescrito como “dominante” após ter sido inicialmente transmitido como “recessivo”. Se esta situação ocorrer significa que a mensagem foi recebida com sucesso por algum dos nós do barramento, caso contrário terá ocorrido um erro tendo início a transmissão do respectivo quadro de erro.

Bit de monitorização

Todos os nós efectuam uma monitorização ao nível do *bit*. Um erro ocorre se o transmissor envia um *bit* “dominante” mas detecta um *bit* “recessivo” no barramento ou na situação inversa. Se esta situação ocorrer um quadro de erro será transmitido.

Bit de enchimento

O CAN usa a técnica do bit de enchimento para verificar a integridade da comunicação. Após a transmissão de 5 bits consecutivos idênticos, o transmissor automaticamente insere um *bit* denominado de enchimento com polaridade oposta aos *bits* transmitidos. Se o receptor não confirmar esta situação transmitirá um quadro de erro.

5.2.3 O protocolo CAN

O protocolo CAN permite que diferentes nós tenham acesso simultâneo ao barramento, se mais que um nó tentar o acesso é necessário um “decisor” para resolver o conflito.

A técnica de acesso ao meio físico utilizada no CAN é o CSMA/CD (*Carrier Sense Multiple Access with Collision Detect*) que é uma extensão do tradicional CSMA (*Carrier Sense Multiple Access*). As modificações efectuadas tiveram como objectivo o reforço da resolução determinística de colisões num meio partilhado, através da introdução de identificadores prioritários na troca de mensagens. Desta forma, o CAN resolve de uma forma determinística qualquer colisão que ocorra no barramento [Cena and Valenzano, 97]. As mensagens de prioridade mais alta ganham o acesso ao barramento em detrimento de mensagens com prioridade mais baixa, sendo estas automaticamente retransmitidas no próximo ciclo do barramento ou nos ciclos seguintes, se não houver em espera mensagens de prioridade superior. Qualquer potencial conflito no acesso ao barramento é resolvido por um *bit* de sinalização (decisor) que pode tomar um de dois valores lógicos complementares: dominante ou recessivo.

O início da transmissão de uma mensagem é função da subcamada MAC. Sempre que esta detecta que o barramento está livre, isto é, quando nenhuma mensagem foi transmitida e foram detectados os 3 *bits* que constituem o campo INT (*INTermission*), inicia a transmissão de uma mensagem. Durante a fase de avaliação inicial, cada transmissor compara o *bit* que foi transmitido com o nível observado no barramento. Quando um *bit* recessivo é transmitido e um nível dominante é detectado no barramento, é necessário abortar a transmissão imediatamente, desde que se perca a “disputa”. Uma outra tentativa para enviar a mensagem poderá ser efectuada após a transmissão da mensagem vencedora ter terminado. Um mecanismo “decisor” assegura que quando uma colisão ocorre, todos os nós emissores parem a sua transmissão excepto um. Desta forma, a “contenda” é deterministicamente resolvida, e nenhuma informação é perdida.

Todos os nós da rede recebem as mensagens, fazendo em seguida um teste de aceitação para determinar se a mensagem e consequentemente o seu conteúdo é relevante para o

respectivo nó. Se a mensagem for relevante poderá ser processada, caso contrário é ignorada. Esta função é executada pela subcamada LLC. Esta gere também a retransmissão das mensagens que perderam a “contenda” ou que se viram envolvidas em erros de transmissão.

5.3 Software da camada de aplicação

O *software* que efectua a gestão das comunicações no PC baseia-se na biblioteca de rotinas fornecida pela STZP. Na comunicação entre o PC e a Universal CAN I/O, o PC funciona como nó mestre e a Universal CAN I/O como nó escravo, isto é, o início de toda e qualquer comunicação é da exclusiva responsabilidade do PC. A velocidade de transmissão usada é de 1 Mbit/s.

O formato das mensagens trocadas na rede obedece ao formato da especificação padrão descrita anteriormente. Todas as mensagens enviadas pelo PC são constituídas por um único *byte* no campo de dados (quadro de dados) que indica o número de medidas a efectuar pelo sensor que fica activo. A excepção é a mensagem de término da comunicação e de paragem da actividade de qualquer ultrassom que esteja activo. Esta não tem nenhum *byte* no campo de dados (quadro remoto), pois o próprio identificador contém a informação necessária. Como são utilizados 4 sensores de ultrassons, foram definidos quatro identificadores diferentes e mais um identificador que indica o fim da comunicação e coloca todos os ultrassons no estado inactivo (Tabela 5-2).

ID – Identificador (hex) (bin)	Sensor de Ultrassom Activo
0x08 (1000)	Sensor 1
0x10 (10000)	Sensor 2
0x18 (11000)	Sensor 3
0x20 (100000)	Sensor 4
0x48 (1001000)	Todos inactivos e término da comunicação

Tabela 5-2 Identificadores usados pelo sistema de aquisição de dados implementado.

Os valores dos identificadores não são sequenciais para que seja necessário apenas testar o valor de um *byte* em vez de dois, porque os três *bits* menos significativos dos 11 que constituem o identificador pertencem a outro *byte*, tornando assim mais rápida e fácil a implementação. No caso do microcontrolador utilizado (*Philips 80C592*) os 11 *bits* encontram-se divididos por 2 registos, DSCR1 (8 *bits* mais significativos) e DSCR2 (3 *bits* menos significativos), como se pode observar na Figura 5.8. Quando chega uma mensagem à Universal CAN I/O o *software* nesta apenas utiliza o valor do registo DSCR1 para comparação e posterior desencadeamento da respectiva acção, isto é, se o valor do DSCR1 for 0x01 ficará o ultrassom 1 activo, se for 0x02 será o ultrassom 2, se for 0x03 será o ultrassom 3, se for 0x04 será o ultrassom 4 e se for 0x09 coloca inactivo todos os ultrassons terminando a comunicação.

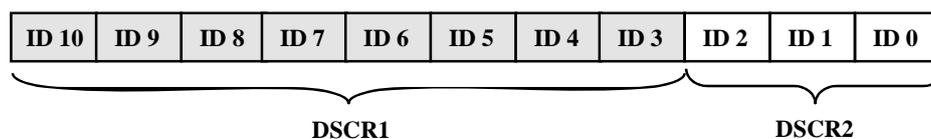


Figura 5.8 Distribuição dos 11 bits do identificador no microcontrolador *Philips 80C592*.

O *software* implementado na Universal CAN I/O tem que efectuar 3 tarefas:

1. Activar o ultrassom solicitado pelo PC por activação da linha digital correspondente.
2. Ligação entre a Universal CAN I/O e a placa da *Polaroid* via comunicação série a uma velocidade de 9600 bit/s para obtenção da medida efectuada pelos ultrassons (Figura 4.16). A opção por uma comunicação série em detrimento da saída analógica da placa *Polaroid* (que poderia ter sido ligada a uma das entradas A/D da Universal CAN I/O), resultou na obtenção de dados com maior precisão e rapidez.
3. Enviar a informação solicitada pelo PC (medidas efectuadas pelos ultrassons) via rede CAN. Para executar esta tarefa foram desenvolvidas entre outras, as rotinas necessárias à inicialização, transmissão e recepção de mensagens, que permitem controlar a transferência de dados sobre uma rede deste tipo. É de salientar que a recepção das mensagens foi implementada por interrupção, tendo prioridade máxima sobre qualquer outra tarefa.

A activação de um determinado ultrassom corresponde a actuar uma das saídas digitais (0 V-24 V) disponíveis na Universal CAN I/O, correspondente ao ultrassom pretendido. Este procedimento faz actuar o contacto de um relé (são 4 - um para cada ultrassom) fechando o circuito entre o ultrassom e a placa da *Polaroid* que o controla. Desta forma só um dos ultrassons será activado num determinado instante evitando o problema de interferência mútua (*cross talk*), possibilitando também a utilização de uma única placa para controlar vários ultrassons.

O protocolo de comunicação série foi imposto pelo *software* existente na placa *polaroid*, que ao envio de um *carriage return* por parte da Universal CAN I/O responde devolvendo o valor medido pelo ultrassom.

As mensagens CAN enviadas pela Universal CAN I/O têm um campo de dados com dois comprimentos diferentes, 7 *bytes* e 1 *byte*. As primeiras transportam a informação respeitante à medida da distância em centímetros efectuada pelo ultrassom. As segundas indicam qual o sensor de ultrassons que esteve activo terminando a comunicação.

A Figura 5.9 apresenta a estrutura do *software* implementado nesta placa. Após as inicializações o programa fica num ciclo infinito que só é interrompido quando chega uma mensagem do PC, acontecendo então uma interrupção para o seu atendimento. Em seguida inicia-se o processo de aquisição das medidas pelo sensor identificado pelo ID da mensagem, sendo o número de medidas a efectuar determinado pelo valor do campo de dados. O atraso de 6 ms em cada aquisição garante que a placa *Polaroid* já adquiriu um novo valor. Este tempo foi estimado tendo em consideração o tempo de propagação do som no ar (343.2 m/s a 20°C), a distância máxima a ser medida (80 cm) assim como algum processamento efectuado pela placa *polaroid*.

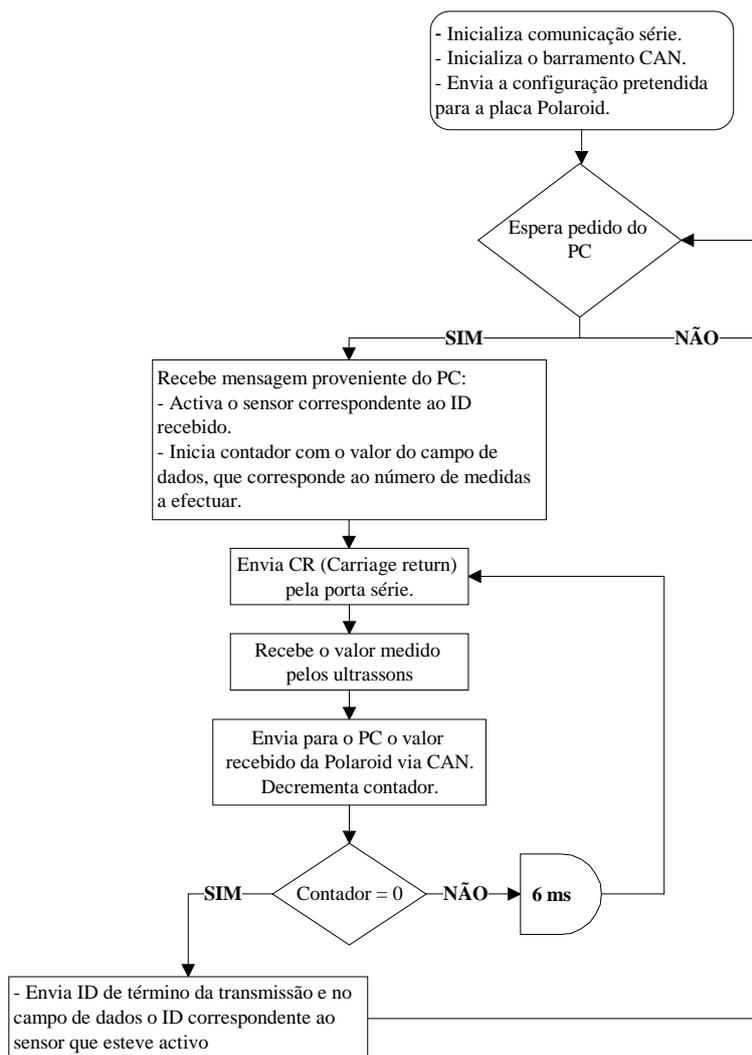


Figura 5.9 Fluxograma do *software* implementado na Universal CAN I/O.

5.4 Sumário

Neste capítulo foram apresentadas as características fundamentais de um barramento do tipo CAN. Este suporta arquitecturas multi-mestre, sendo adequado para ambientes distribuídos em pequena escala.

Devido à simplicidade do *hardware* necessário e a facilidade com que funções básicas de transferência de dados são implementadas (sem qualquer preocupação com a implementação de códigos de detecção/correção de erros) e levando ainda em consideração a flexibilidade que resulta da possibilidade de em qualquer instante aumentar o número de nós sem modificações significativas ao nível do *software*, este tipo de rede provou ser uma boa solução.

Capítulo 6

6. Sistema de aquisição de imagem

6.1 Introdução

Neste capítulo apresenta-se com detalhe a estrutura de *software* que implementa a hierarquia de visão, cuja finalidade é restringir a área de trabalho do robô à superfície visível do objecto. Foram utilizadas técnicas de visão por computador cujo “produto final” é a obtenção dos pontos dominantes que caracterizam a silhueta do objecto.

São três as tarefas a desempenhar pelo sistema de visão:

1. Procura/detecção do objecto na área de trabalho;
2. Alinhamento da câmara com o centro de massa do objecto;
3. Obtenção dos pontos dominantes.

O capítulo divide-se em duas partes distintas: na primeira parte são discutidas questões relacionadas com a iluminação, sendo em seguida apresentadas com algum detalhe as técnicas de processamento de imagem implementadas; na segunda parte esclarece-se a sequência de acções necessárias ao desempenho das tarefas atribuídas ao sistema de visão. O *software* necessário para o sistema de visão foi desenvolvido pelo aluno Nuno Monteiro [Monteiro, 97] no seu estágio de final de curso sob orientação do doutorando.

6.2 Visão por computador

A denominação “visão por computador” é utilizada em aplicações de sistemas computadorizados que recebem e processam informação visual.

A nível funcional, e de uma forma genérica, o processo de visão por computador pode dividir-se nos seguintes componentes: aquisição, pré-processamento, segmentação, representação e descrição, reconhecimento, interpretação. Aquisição é o processo do qual resulta numa imagem digital. O pré-processamento está relacionado com técnicas de filtragem básica da imagem, como a remoção de ruído, ou o realce de detalhes. A segmentação divide uma imagem em regiões de interesse. A descrição está relacionada com a definição das características dos objectos (por exemplo o tamanho, e a forma), tornando possível a sua distinção. O reconhecimento é o processo pelo qual se identifica os objectos.

Finalmente, a interpretação atribui significado ao objecto reconhecido, isto é, indica o que fazer com ele [Gonzalez and Woods, 92] [K. S. Fu et al., 87].

É usual agrupar estes métodos em três níveis de visão: baixo, médio e alto. Do nível baixo de visão fazem parte os métodos que melhoram a qualidade de uma imagem com a finalidade de aumentar as probabilidades de sucesso dos processos seguintes. A fase pré-processamento insere-se neste nível. Do nível intermédio de visão, fazem parte a segmentação, a descrição e o reconhecimento. Nestes processos recebe-se como entrada a imagem pré-processada, sendo posteriormente gerada uma estrutura de dados que representa algumas das características da mesma, como por exemplo a descrição do contorno do objecto. A interpretação constitui o nível mais alto de visão, uma vez que processa as estruturas de dados, gerando outras estruturas ou tomando decisões com base nas entrada. À medida que se avança do nível mais baixo para o mais alto, diminui a quantidade de dados a processar, mas aumenta a complexidade do algoritmo de processamento [K. S. Fu et al., 87].

A divisão descrita constitui um modelo genérico e uma visão prática dos sistemas de visão actualmente desenvolvidos. Com base neste modelo foram implementadas as fases de aquisição, pré-processamento, segmentação e descrição, de acordo com o diagrama funcional apresentado na Figura 6.1.

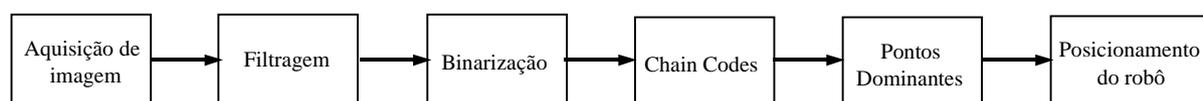


Figura 6.1 Diagrama funcional do sistema de visão implementado.

Neste diagrama, a filtragem corresponde ao pré-processamento de imagem, a binarização ao processo de segmentação, e os *chain codes*, juntamente com os pontos dominantes, ao processo de descrição (no sistema intervêm apenas os níveis baixo e intermédio de visão).

6.2.1 Iluminação

A iluminação é um factor essencial quando se trabalha em visão por computador, na medida em que a sua eficiência se traduz numa maior ou menor complexidade de alguns dos algoritmos de visão a implementar. Uma boa iluminação possibilita um bom contraste, minimiza reflexos e atenua ou elimina sombras, reduzindo a complexidade do processamento de sinal.

São várias as técnicas de iluminação utilizadas salientando-se no entanto quatro: a iluminação frontal, a retro-iluminação, a iluminação direccionada e a iluminação estruturada (Figura 6.2).

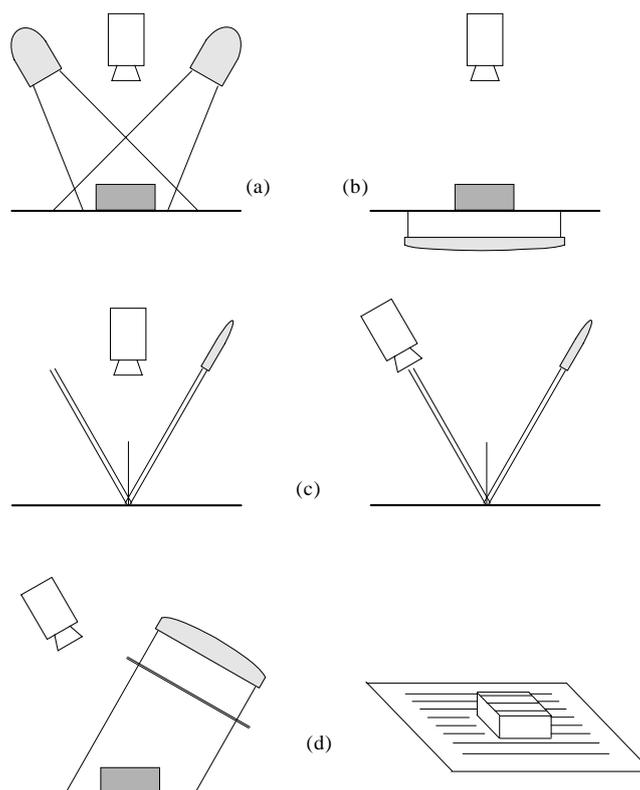


Figura 6.2 Técnicas básicas de iluminação: (a) Iluminação frontal. (b) Retro-iluminação com fonte de luz de raios paralelos. (c) Iluminação direccional. O tracejado representa a dispersão do raio de luz num defeito. (d) Iluminação estruturada. Neste caso um padrão que intercepta os raios de luz é usado para definir o padrão de luz escolhido: um conjunto de linhas paralelas.

A técnica de iluminação frontal ilumina a região de interesse, de forma a que sejam as propriedades da superfície do objecto as responsáveis pelas características de interesse na imagem. As fontes de luz estão localizadas do mesmo lado da câmara e direccionadas para o objecto. Estas podem ser em qualquer número, podendo ainda ser mais ou menos direccionais. O caso extremo de não direccionalidade designa-se “iluminação difusa”.

A retro-iluminação facilita a obtenção de imagens binarizadas. As luzes estão localizadas para além do objecto, estando direccionadas para uma superfície entreposta entre estas e o objecto. Esta técnica de iluminação é adequada a aplicações nas quais a silhueta do objecto é suficiente para a análise a efectuar. Um exemplo consiste nas mesas de luz (ou as mesas retro-iluminadas) sobre as quais se colocam os objectos a serem analisados, estando a câmara por cima da mesa e apontando para esta.

A iluminação direccional é usada principalmente na inspecção de superfícies reflectoras. Os defeitos numa superfície, como os riscos ou os buracos, podem ser detectados utilizando-se um raio de luz altamente direccional e analisando-se a dispersão do raio na superfície.

A técnica de iluminação estruturada utiliza um padrão de luz conhecido. Neste caso é habitual a utilização de um ou mais pontos, ou uma ou mais riscas ou grelhas. Estabelecido o padrão de luz, a presença do objecto é detectada pelos distúrbios que este produz no padrão projectado. Uma das vantagens desta técnica é a possibilidade de extrair informação tridimensional do objecto a partir da análise da perturbação no padrão.

No caso deste trabalho e tendo em consideração que o que se pretende extrair e descrever é a silhueta do objecto a partir de um pequeno número de pontos (pontos dominantes), a retro-iluminação seria a técnica mais apropriada. No entanto, dado que se procura uma

solução económica e o posicionamento relativo da mesa, câmara e objecto, esta técnica era de difícil implementação, pelo que foi preterida pela iluminação frontal. Como esta técnica dá importância à superfície do objecto surgiram alguns problemas com o contraste e as sombras.

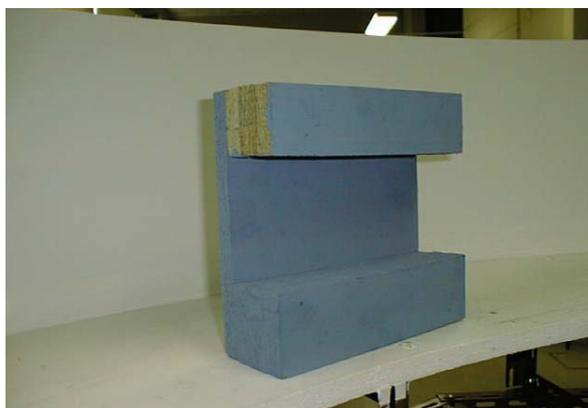


Figura 6.3 Aspecto da mesa e orla construída para possibilitar o contraste com objectos mais escuros.

Para resolver o problema do contraste optou-se por construir um fundo de cor uniforme (branco) para a imagem tal como se mostra na Figura 6.3. Assim a situação óptima ou de contraste máxima será obtida quando o objecto for preto. Em termos práticos, verificou-se que não é necessário um contraste muito elevado, bastando um grau que possibilite que na imagem de tons de cinzento, se distingam com clareza as regiões do objecto e do fundo.

O problema das sombras surge sempre que o contraste entre a sombra e o fundo é suficiente para que esta seja considerada parte integrante do objecto. A estratégia para resolver este problema consistiu na utilização de uma fonte de luz direccionada para a sombra. Assim, a região de sombra provocada pela incidência de uma fonte de luz será a região de iluminação de outra fonte de luz, conseguindo-se uma redução acentuada do contraste entre a sombra e o fundo, deixando estas de interferir no processo de aquisição.

6.2.2 Filtragem

Com o processo de filtragem pretendia-se reduzir efeitos indesejáveis que pudessem surgir na imagem adquirida como resultado da quantificação, da amostragem, transmissão ou distúrbios no ambiente, aquando da sua aquisição. Pretendia-se ainda que a perda de informação respeitante ao contorno fosse mínima ou não existisse. Foram implementados e testados dois filtros passa-baixo: filtragem de média e filtragem de mediana.

Filtro de média

Este método baseia-se na atribuição a cada ponto da imagem de um valor obtido através do cálculo da média dos pontos na sua vizinhança. Em termos formais, o processo pode ser descrito da seguinte forma: dada uma imagem $f(x, y)$, obtêm-se uma imagem suavizada $g(x, y)$ na qual cada ponto (x, y) é obtida calculando-se a média de intensidades dos valores dos pontos de f compreendidos numa vizinhança pré-definida desse ponto. A imagem suavizada é assim obtida pela relação

$$g(x, y) = \frac{1}{P} \sum_{(n,m) \in S} f(x, y) \quad (6.1)$$

para todo o x e y da imagem $f(x,y)$. O S é o conjunto das coordenadas dos pontos da vizinhança de (x,y) , incluindo o próprio. O P é o número total de pontos da vizinhança, incluindo também o ponto (x,y) .

A forma da vizinhança pode ser qualquer, mas a quadrada é de longe a mais utilizada nos sistemas de visão. A vizinhança pode ter qualquer dimensão, sendo contudo os tamanhos pequenos os mais comuns. O tamanho e a forma da vizinhança são definidos pelo padrão do filtro escolhido. No caso presente, e por questão de eficiência, o filtro efectua a média de 9 pontos recorrendo para tanto a uma máscara constituída pela matriz 3x3

$$\begin{bmatrix} \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \end{bmatrix}. \quad (6.2)$$

Filtros de 5x5, 7x7 ou mesmo 11x11 são também usuais. Na escolha do tamanho do filtro dois aspectos são influentes: o primeiro diz respeito ao número de cálculos necessários para o processamento completo da imagem, que é tanto maior quanto maior for a dimensão do filtro; o segundo aspecto a ter em conta é que filtros de maior dimensão atenuam as variações mais lentas de tom, o que resulta numa imagem demasiado suavizada, mais esbatida, com poucos detalhes, o que não é desejável.

Filtro de mediana

Neste tipo de filtro procede-se da seguinte forma: o valor de cada ponto da imagem é ordenado juntamente com o valor dos pontos seus vizinhos, determinando-se em seguida a mediana, e atribuindo-se-lhe esse valor. Assim, para uma vizinhança de 3x3 a mediana é o quinto valor mais baixo (ou mais alto) do conjunto de valores da vizinhança e do próprio ponto. Suponhamos por exemplo, que o valor de um ponto é 130 e os da sua vizinhança (99, 105, 97, 98, 110, 100, 109, 98). Ordenando de forma crescente estes valores (97, 98, 98, 99, 100, 105, 109, 110, 130), resulta uma mediana de 100. Este valor é então atribuído ao ponto, conseguindo-se desta forma eliminar o ruído.

A filtragem baseada na mediana tem como principal função, forçar os pontos com intensidades muito distintas a serem como os seus vizinhos, conseguindo-se assim eliminar picos de intensidade que apareçam isolados, conservando os detalhes da imagem.

6.2.3 Segmentação

A segmentação é o processo de divisão da imagem em regiões distintas. É normalmente o primeiro passo na procura automática, reconhecimento, ou contagem de objectos numa imagem. O resultado do algoritmo de segmentação muitas vezes determina o sucesso ou fracasso de todo o trabalho de análise seguinte, como a descrição ou o reconhecimento.

A segmentação é conseguida de uma forma contextual ou não contextual. Na segmentação contextual o processo de segmentação emprega relações de vizinhança entre as características presentes na imagem. De uma forma genérica, a classificação contextual é normalmente mais bem sucedida, uma vez que a informação local da imagem reforça a decisão de classificação. No entanto, é também teoricamente difícil de quantificar [Shaulkoff, 89] a detecção de variações bruscas nos tons de imagem (detecção dos contornos) e a formação de regiões com características semelhantes. A binarização é uma técnica de segmentação não contextual, sendo uma das mais usadas em visão industrial. Embora não utilizando qualquer informação

contextual ou estrutural, quando há necessidade de se extrair objectos de uma imagem, a binarização revela-se uma ferramenta importante, podendo ser facilmente implementada e executada a altas velocidades [Lima and Campilho, 94]. Dadas as características das imagens tratadas neste trabalho, foi esta a técnica de segmentação adoptada.

Binarização

Uma imagem $f(x, y)$ com 256 tons de cinzento é convertida numa imagem binária de acordo com a função

$$g(x, y) = \begin{cases} 0 & \text{se } f(x, y) \leq T \\ 255 & \text{se } f(x, y) > T \end{cases} \quad (6.3)$$

O critério de atribuição do tom preto (valor 0) ou branco (valor 255) reside na comparação do tom de $f(x, y)$ com um nível de binarização T escolhido. Uma abordagem mais generalizada desta técnica é a chamada binarização multi-nível, normalmente utilizada quando existe mais que um objecto de tons distintos sobre um fundo contrastante.

A dificuldade da binarização tem a ver com a escolha do nível correcto de binarização. Uma forma de determinar o nível de T é através da análise do histograma (Figura 6.4).

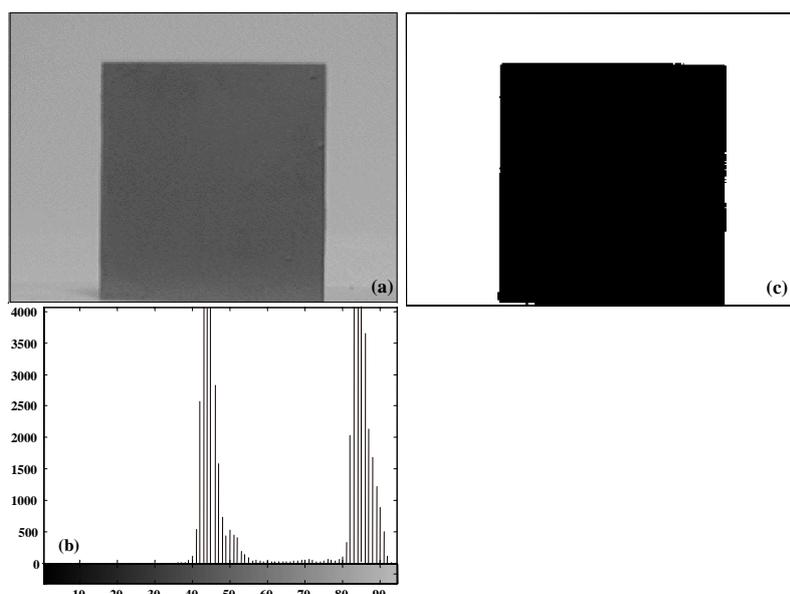


Figura 6.4 binarização: (a) imagem original; (b) histograma de (a); (c) binarização com um nível 64.

No histograma da Figura 6.4 sobressaem dois picos ou modas de intensidade. Uma vez que o objecto é mais escuro que o fundo, a primeira moda corresponderá a pontos do objecto e a segunda a pontos do fundo. Desta forma, um nível de binarização adequado à separação do objecto do fundo encontra-se entre os dois picos do histograma. Quando a binarização é efectuada em toda a imagem a partir de um nível escolhido, é classificada como global.

Uma binarização local é também possível. Este tipo de binarização tem em conta propriedades locais (por exemplo a média das intensidades na vizinhança), sendo normalmente utilizada em imagens onde seja necessário compensar efeitos de iluminações não uniformes, reflexos ou sombras.

No caso presente, e uma vez que as condições de iluminação são mantidas sob controlo e as propriedades físicas dos objectos (nomeadamente a sua reflexividade) pouco variam,

constatou-se que a utilização do método de binarização global era passível de ser utilizado. Contudo, embora a utilização de um nível fixo de binarização fosse satisfatório, optou-se por a utilização de um método automático de escolha do nível denominado método de Otsu [Azeredo et al., 96] [Lima and Campilho, 94]. Este utiliza o histograma para calcular o nível de binarização assumindo a existência de pelo menos dois picos, um correspondente ao objecto e outro ao fundo. A separação entre os dois picos é tanto maior quanto melhor for o contraste entre o objecto e o seu fundo. O método consiste na separação em duas classes dos pontos pertencentes ao fundo e ao objecto pela maximização da expressão

$$\omega_0 \omega_1 (\mu_1 - \mu_0)^2 \tag{6.4}$$

onde

$$\omega_0 = \sum_{i=0}^k p_i \quad \omega_1 = \sum_{i=k+1}^{255} p_i \quad \mu_0 = \frac{\sum_{i=0}^k i \cdot p_i}{\omega_0} \quad \mu_1 = \frac{\sum_{i=k+1}^{255} i \cdot p_i}{\omega_1}$$

e

$$p_i = \frac{n_i}{N}$$

ω_0 e ω_1 - são os momentos cumulativos de ordem zero.

μ_0 e μ_1 - são os momentos cumulativos de primeira ordem.

i - é o valor de cinzento.

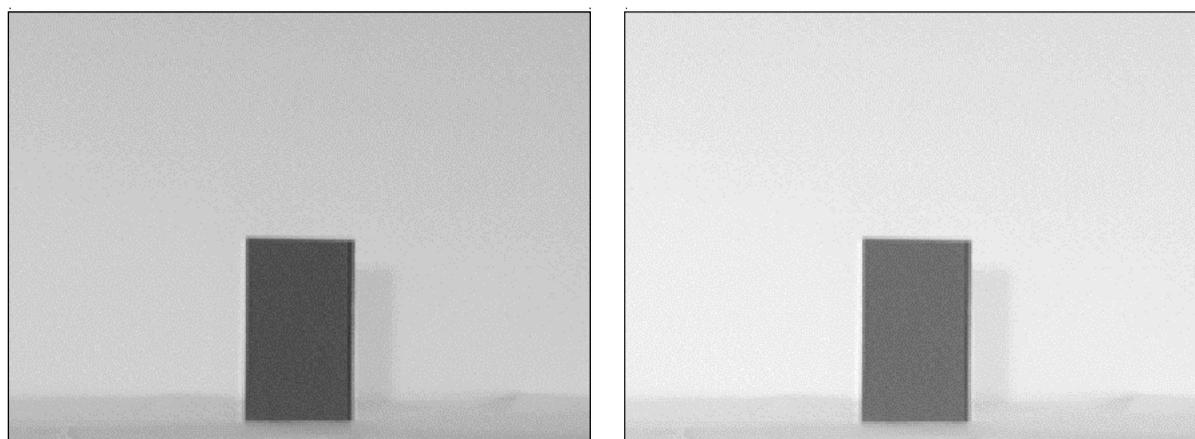
k - é o nível de binarização pretendido.

p_i - probabilidade do valor de cinzento i .

n_i - número de pontos com valor de cinzento i .

N - número total de pontos.

Esta opção dotou o sistema de visão de uma maior robustez e flexibilidade, quer às condições de iluminação, quer às características dos objectos a utilizar. A Figura 6.5 mostra o resultado da binarização de Otsu quando se trabalha em condições de iluminação distintas.



(a)

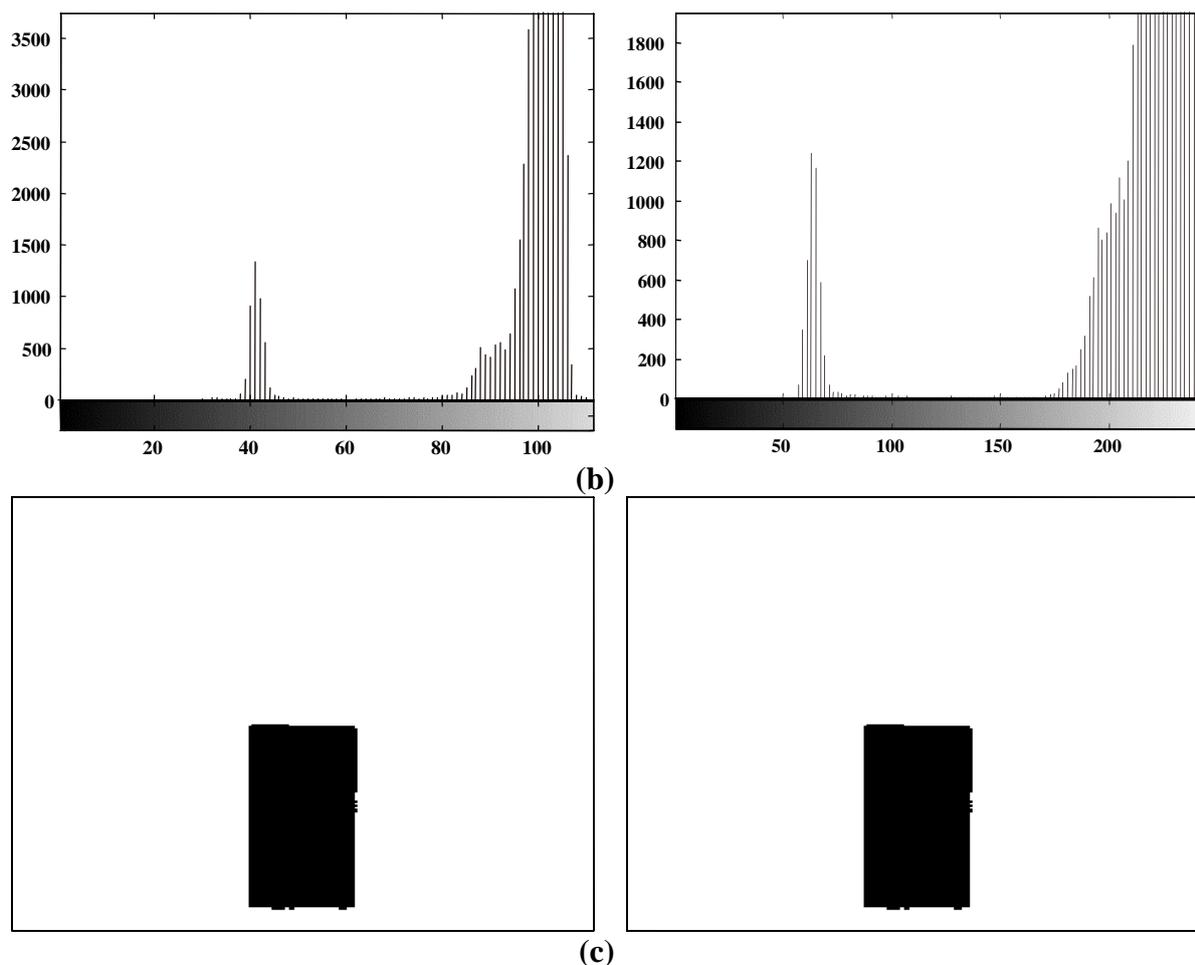


Figura 6.5 Resultados da binarização com duas intensidades de iluminação: (a) imagens originais; (b) histogramas; (c) imagens binarizadas com os níveis de 75 (à esquerda) e 149 (à direita) determinados automaticamente pelo método de Otsu.

6.2.4 Descrição

Esta fase corresponde à transformação da imagem num tipo de representação mais compacta, constituindo o primeiro passo na análise da imagem. Encontrando-se esta segmentada em duas regiões distintas (objecto e fundo), interessa então representar ou descrever a região de interesse – a região correspondente ao objecto. Existem preferencialmente duas formas de descrever essa região, uma baseada nas características externas (o contorno) e outra nas características internas (os pontos que formam a região) [Gonzalez and Wintz, 87]. Normalmente, a descrição externa é escolhida quando o interesse está na forma do objecto, enquanto que uma descrição interna se revela mais apropriada quando se pretende representar propriedades como a cor ou textura do objecto.

Pretende-se neste caso, uma representação externa adequada do objecto na posição em que estiver no momento da sua aquisição. A representação externa do objecto pode efectuar-se através do seu perímetro, comprimento máximo, número de concavidades e convexidades bem como da orientação dos segmentos de recta que unem pontos extremos do objecto. Existem, no entanto, um conjunto de técnicas mais usuais e genéricas, adequadas para a representação do contorno, como por exemplo, os *chain codes*, as assinaturas, as aproximações poligonais e os descritores de Fourier. A representação baseada nos *chain codes* foi o método adoptado neste trabalho, devido à sua simplicidade de implementação e fidelidade na descrição do contorno.

Detecção do contorno

Embora a detecção de pontos e linhas sejam objecto de alguma discussão no processo de segmentação, a detecção do contorno é com certeza a aproximação mais comum para detectar discontinuidades significativas nos níveis de cinzento. Esta situação resulta de que pontos isolados e linhas finas não ocorrem com frequência na maioria das aplicações práticas.

Existem diversos algoritmos para a detecção do contorno que podem ser utilizados para extrair os pontos que o representam. Os algoritmos mais comuns baseiam-se em operadores de gradiente e de Laplace. Um contorno é definido como uma transição entre uma região escura e uma região clara ou o contrário.

O algoritmo aqui utilizado na detecção do contorno é muito simples, baseando-se unicamente na noção de vizinhança. Partindo da imagem segmentada (através da binarização) é avaliado se os pontos com as coordenadas $(x+1, y)$ e $(x-1, y)$ pertencem a regiões distintas. Se esta situação se verificar então o ponto (x, y) é identificado como pertencente ao contorno. A representação deste é efectuada por o *chain codes*.

Representação do contorno por chain codes

O *chain code*, utilizado pela primeira vez por Freeman em 1961, baseia-se no uso de identificadores para a caracterização das direcções do contorno. O número de identificadores utilizados depende da conectividade considerada. No caso implementado, foram utilizados os identificadores apresentados na Figura 6.6.

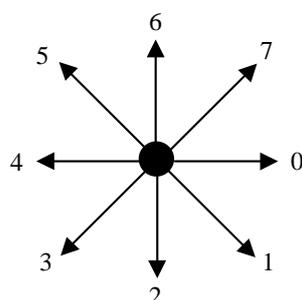
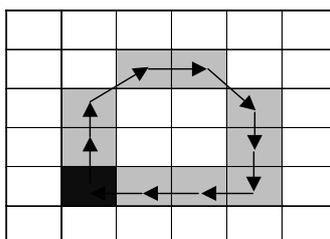


Figura 6.6 Identificadores utilizados na codificação das direcções.

Tomando como exemplo a imagem exibida na Figura 6.7, começa-se o processo de representação por *chain code* a partir de um ponto qualquer do objecto pertencente ao contorno (neste caso foi seleccionado o ponto inferior esquerdo), avançando-se de seguida ao longo deste. A direcção do movimento de um ponto para o outro é guardada, construindo-se a sequência denominada *chain code*. O procedimento é repetido até que se regresse ao ponto de partida. A Figura 6.7 mostra o resultado da aplicação desta técnica.



Chain code: 6670122444

Figura 6.7 Representação por *Chain code*. O ponto preto assinala o início do processo.

O algoritmo de procura da direcção correcta do contorno é um aspecto a considerar. A primeira direcção de procura é definida pela seguinte regra: adicionar 4 à que resultou na entrada do ponto actual usando aritmética módulo 8, e adicionar 1 ao resultado obtido. Caso não se encontre o ponto do contorno, a pesquisa continua na direcção imediatamente seguinte à que se tentou, repetindo-se até que um ponto do contorno seja encontrado [Kasturi and Trivedi, 90]. Concretizando num exemplo, supondo que a direcção de entrada no ponto foi a 4 então a primeira direcção a procurar será a 1 $((4+4) \text{ MOD } 8 + 1 = 1)$. Um outro aspecto a ter em consideração no processo de seguimento e descrição do contorno, é que este não poderá ser composto por linhas descontínuas, mas sim por uma curva fechada e contínua. Este aspecto é sempre garantido, uma vez que é descrita a mancha de pontos associada ao objecto e resultante da binarização.

Um dos problemas apresentados por este método é a descrição do ruído que interfere localmente com o contorno. Para o atenuar foram tomadas duas opções:

1. Na pesquisa do primeiro ponto do contorno procura-se um agrupamento de pontos (4 neste caso) pertencentes ao objecto em detrimento de um só ponto.
2. A utilização do método descrito em [K. S. Fu et al., 87], que se baseia na utilização de equações lógicas como complemento dos resultados obtidos pela aplicação do filtro da mediana.

A aplicação do filtro da mediana consegue corrigir pequenas irregularidades do contorno e da imagem, nomeadamente pequenas concavidades, convexidades e pontos isolados. No entanto este filtro não tem capacidade para reconstruir cantos. A Figura 6.8 apresenta a ampliação de uma imagem binarizada, sendo visível o canto inferior esquerdo de um objecto branco cujo contorno foi afectado por algum ruído e o resultado da aplicação do filtro da mediana.

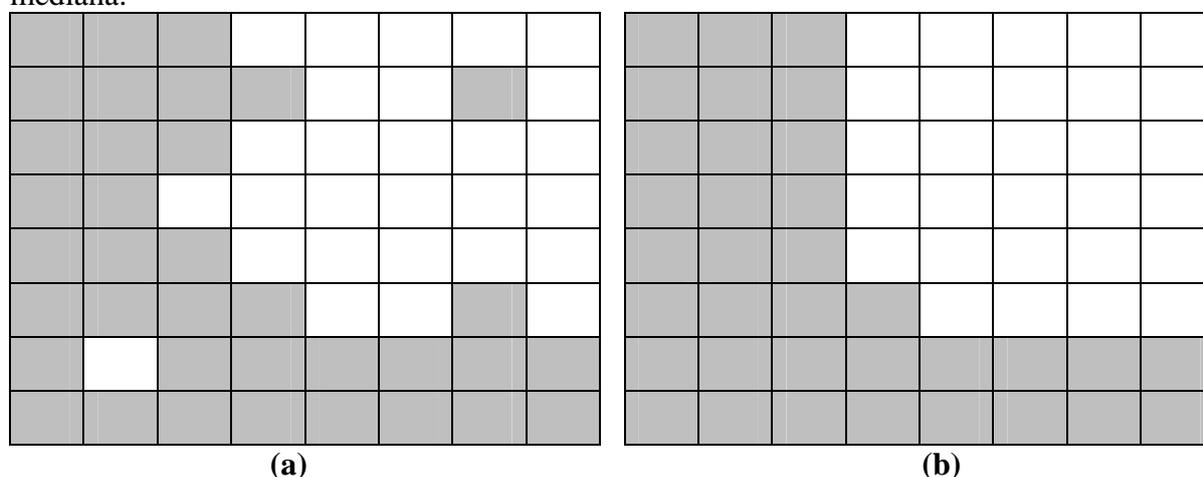


Figura 6.8 Resultado da aplicação do filtro da mediana sobre o detalhe de uma imagem binarizada. (a) imagem original; (b) imagem filtrada.

O método descrito em [K. S. Fu et al., 87] baseia-se num conjunto de equações lógicas que avaliam os pontos de uma vizinhança centrada em p , atribuindo a p um valor verdadeiro (ponto branco) ou falso (ponto preto), dependendo do arranjo espacial dos pontos nessa vizinhança. Nas equações referidas (Figura 6.9) a adição e a multiplicação são utilizadas, respectivamente, como operadores lógicos OR e AND.

a	b	c
d	p	e
f	g	h

$$E_1 = p + bg \times (d + e) + de \times (b + g)$$

$$E_2 = p \times [(a + b + d) \times (e + g + h) + (b + c + e) \times (d + f + g)]$$

$$E_3 = \bar{p} \times (dfg) \times (a + b + c + e + h) + p$$

$$E_4 = \bar{p} \times (abd) \times (c + e + f + g + h) + p$$

$$E_5 = \bar{p} \times (egh) \times (a + b + c + d + f) + p$$

$$E_6 = \bar{p} \times (bce) \times (a + d + f + g + h) + p$$

Figura 6.9 Definição da vizinhança e equações lógicas utilizadas na determinação do valor a atribuir a p.

Pela avaliação da primeira equação, consegue-se a transformação das pequenas concavidades em segmentos de recta e o preenchimento dos pequenos buracos (um ponto) existentes no interior do objecto. A segunda equação permite a eliminação dos pontos claros isolados e a transformação de pequenas convexidades em segmentos de recta. As últimas quatro equações possibilitam a reconstituição dos cantos, estando associadas respectivamente ao canto superior direito, inferior direito, superior esquerdo e inferior esquerdo.

A Figura 6.10 ilustra o resultado da aplicação destas equações.

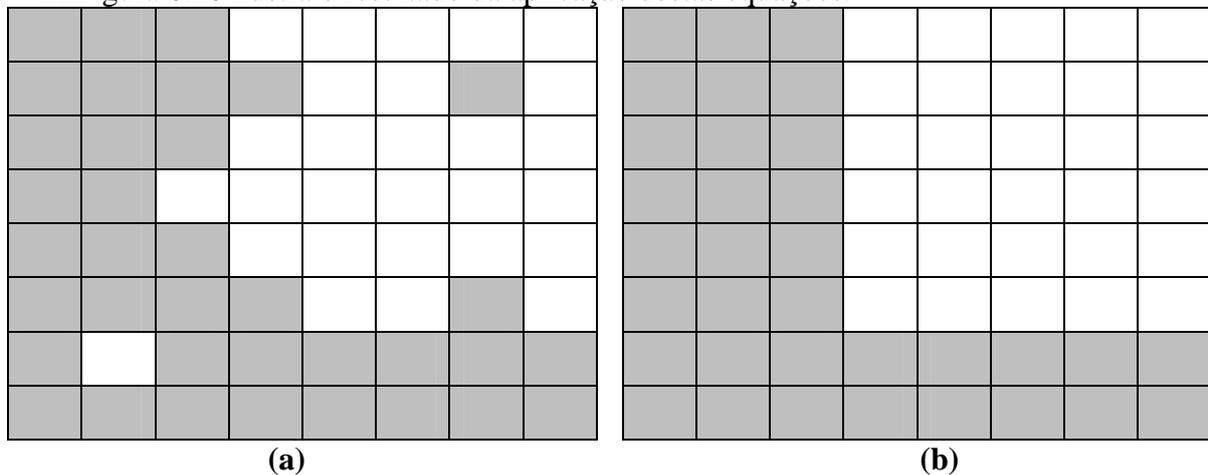


Figura 6.10 Resultado da aplicação das equações lógicas de Fu sobre um detalhe de uma imagem binarizada. (a) Imagem original; (b) Imagem resultante.

Pontos dominantes

Muitas vezes é possível descrever com fidelidade o contorno do objecto por um conjunto reduzido de pontos, designados por “pontos dominantes”. Estes estão relacionados com as zonas do contorno onde existe maior variação, como esquinas, concavidades e convexidades acentuadas (no caso de um objecto rectangular, por exemplo, bastam 4 pontos para identificar claramente o seu contorno).

A natureza discreta e ruidosa dos contornos digitais limita a qualidade de extracção das suas características. Pequenas concavidades e convexidades podem ser introduzidas devido a erros de quantificação, à representação discreta dos contornos e ao ruído. O método de extracção dos pontos dominantes deverá ser capaz de tolerar estas irregularidades e detectar as variações consideradas importantes para a definição dos objectos. O método implementado, actua no domínio espacial constituindo uma versão melhorada do algoritmo de *splitting* clássico apresentado por Duda e Hart, baseando-se na grandeza do deslocamento para decidir sobre a importância de um ponto.

A ideia base da versão clássica do método de *splitting* é a de examinar um conjunto de pontos e verificar se podem considerar-se colineares. Para tal, o contorno do objecto é

inicialmente decomposto em duas secções, sendo cada uma examinada para se verificar se satisfaz o critério de colinearidade. Se uma secção não satisfizer o referido critério, é detectado um ponto dominante e a secção é de novo decomposta em duas subsecções que posteriormente serão examinadas. Este processo é executado recursivamente até serem avaliadas todas as subsecções. A Figura 6.11 esclarece e exemplifica a utilização deste método, podendo o algoritmo utilizado ser descrito como se segue:

1. traça-se o segmento de recta que une os pontos p_1 e p_2 ;
2. encontra-se o ponto q - ponto da curva que se encontra mais distante do segmento de recta $\overline{p_1p_2}$;
3. se a distância for superior a um valor limite pré-estabelecido (grandeza do deslocamento) então q é ponto dominante e definem-se dois novos segmentos $\overline{p_1q}$ e $\overline{qp_2}$ sobre os quais se aplica novamente o algoritmo de *splitting*. Senão, aceita-se a curva entre os pontos p_1 e p_2 como sendo uma recta [Tavares, 93].

O valor limite pré-estabelecido usado para o teste de colinearidade deve ser escolhido de modo a que sejam seleccionados os pontos dominantes desejados, mas ao mesmo tempo tolerados os ruídos indesejados. Um valor muito pequeno deste limite provoca uma selecção de pontos instáveis (não dominantes). Por outro lado, um valor muito grande provoca uma selecção estável mas com um número insuficiente de pontos para uma descrição suficientemente fiel do contorno. Assim, um compromisso entre estes valores deve ser procurado, sendo usual esta pesquisa ser efectuada por experimentação.

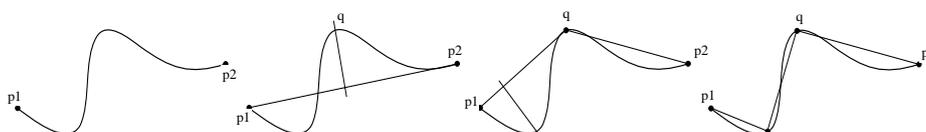


Figura 6.11 Ilustração do método de *splitting* [Tavares, 93].

O método implementado baseia-se neste algoritmo clássico de *splitting*, conservando na íntegra a qualidade de aproximação do algoritmo clássico, e simultaneamente, aumenta de forma significativa a velocidade de processamento (no algoritmo clássico, o tempo de CPU consumido é muito elevado, visto que cada ponto da curva é utilizado mais do que uma vez). A alteração efectuada reside no cálculo da distância, tendo-se substituído a distância euclidiana do algoritmo clássico pela distância incremental. Assim, a este método poder-se-á chamar método de *splitting* com distância incremental [Tavares, 93].

Método de *splitting* com distância incremental

abordagem consiste em escolher os dois pontos do contorno que se encontrem mais distantes um do outro de entre todos os possíveis pares de pontos do contorno. Outra ainda seria escolher-se, sobre o contorno, os pontos superior mais à esquerda e o inferior mais à direita. Seria ainda possível uma abordagem que resolveria a principal desvantagem da primeira solução apresentada, o elevado tempo computacional. O método prevê a escolha dos dois pontos do contorno que estejam mais afastados do centro de massa do objecto e cujos segmentos com o extremo comum do centro de massa estejam afastados no mínimo 90° [Tavares, 93] [E. Davies, 90].

No entanto, no caso presente uma vez que se pretende a descrição do contorno para definição do local de interesse e não uma descrição para um posterior reconhecimento do

objecto, o problema da escolha dos dois pontos iniciais não se revela tão importante. Assim, optou-se pelo método mais simples: o primeiro ponto é o ponto inferior mais à esquerda do contorno do objecto, que corresponde ao ponto encontrado pelo processo de descrição em *chain code* do contorno; o segundo ponto é o terceiro ponto do contorno. Este segundo ponto não é considerado dominante, servindo apenas para definir a direcção requerida para iniciar o processo.

A Figura 6.12 demonstra a eficiência do método sobre contornos ideais, isto é, sem os efeitos introduzidos pelo ruído. O objectivo do quadrado é o de avaliar a selecção dos pontos efectuada pelo método sobre cantos perfeitos. O objectivo do triângulo é o de mostrar o seu comportamento num segmento de recta cuja direcção não coincide com nenhum código de direcção do *chain code*. Finalmente, a circunferência é usada para se observar os pontos seleccionados num contorno curvo.

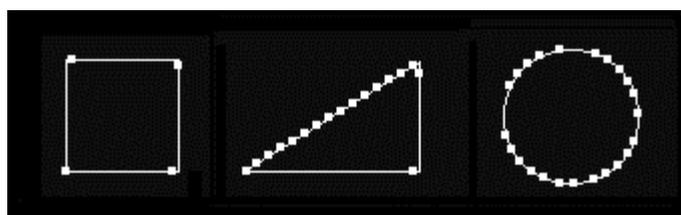


Figura 6.12 Resultado do método de *splitting* com distâncias incrementais sobre contornos ideais. O limite utilizado foi 3.5.

Como se pode constatar este método tem a desvantagem de falhar nos cantos perfeitos. A diferença entre o ponto desejado e o ponto assinalado é tanto maior quanto maior for o valor limite usado para o teste de colinearidade. Resulta também pontos dominantes indesejados em linhas rectas cuja direcção não coincida com nenhum dos códigos de direcção usados no processo de codificação, dando tantos mais pontos quanto menor for o limite utilizado. Esta característica é consequência da natureza discreta de uma imagem digital. Por outro lado, o método revela-se satisfatoriamente adequado nos contornos curvos.

Para colmatar essa limitação, recorreu-se a um processo de selecção dos pontos dominantes a partir dos pontos candidatos seleccionados através da aplicação do método dos desvios angulares.

Método dos desvios angulares

O processo de selecção utilizado para escolha dos candidatos a pontos dominantes baseia-se no princípio dos desvios angulares. Os pontos são seleccionados segundo o seguinte critério: desvios grandes seleccionam pontos, desvios pequenos não seleccionam pontos. A direcção é definida por três pontos candidatos consecutivos. Por outro lado o valor limite contrariamente ao que se passava no método de *splitting*, é uma grandeza relativa variando de acordo com a distância entre os dois pontos extremos dos três candidatos que em cada instante são analisado. A função escolhida para definir esta grandeza foi o quociente entre o valor do limite máximo permitido e a distância euclidiana dos dois pontos anteriormente citados (pontos extremos). Conseguir-se assim, tolerar desvios maiores quando a distância entre pontos é menor e desvios menores para distâncias maiores. A Figura 6.13 mostra o resultado da aplicação do método de *splitting* seguido do método de selecção.

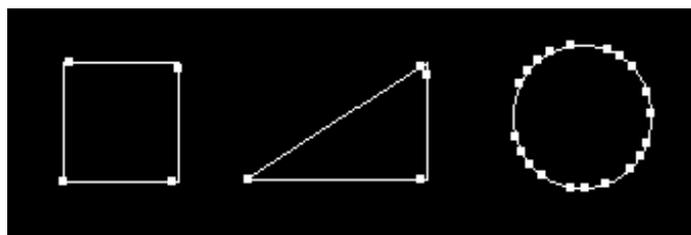


Figura 6.13 Pontos dominantes de contornos ideais extraídos pelo método de *splitting* seguido do método de selecção (desvio angular).

Como se pode observar o processo de selecção consegue resolver a descrição dos segmentos de recta de direcção não coincidente com os códigos de direcção usados no método de *splitting*.

6.3 Detalhes da implementação

Como já foi referido, antes de se iniciar o varrimento da peça para obtenção das características 3D, é necessário que o sistema de visão efectue as três seguintes tarefas:

1. Procura/detecção do objecto.
2. Alinhamento da câmara com o centro de massa do objecto.
3. Extracção dos pontos dominantes.

A procura/detecção do objecto na área de trabalho é efectuada por comparação da imagem adquirida com a imagem do fundo (cor branca uniforme). Considera-se que foi detectado um objecto quando este ocupa no mínimo 3% da área adquirida por a câmara.

Para se efectuar o alinhamento é calculado o desvio entre o centro de massa da parte visível do objecto e o ponto médio da área total adquirida por a câmara. Em seguida, o robô desloca-se para uma nova posição determinada com base no cálculo anterior. Trata-se de um processo contínuo até o alinhamento se efectuar.

A extracção dos pontos dominantes é o resultado final do processamento da imagem adquirida após a conclusão das primeiras duas tarefas.

O fluxograma da Figura 6.14 apresenta a sequência de acções necessárias para efectuar as tarefas atribuídas ao sistema de visão.

É de salientar que todas as acções que envolvem o movimento da junta 1 do robô (única junta a ser movimentada uma vez que a câmara de vídeo se encontra sobre o ombro do robô) são entregues à hierarquia do manipulador.

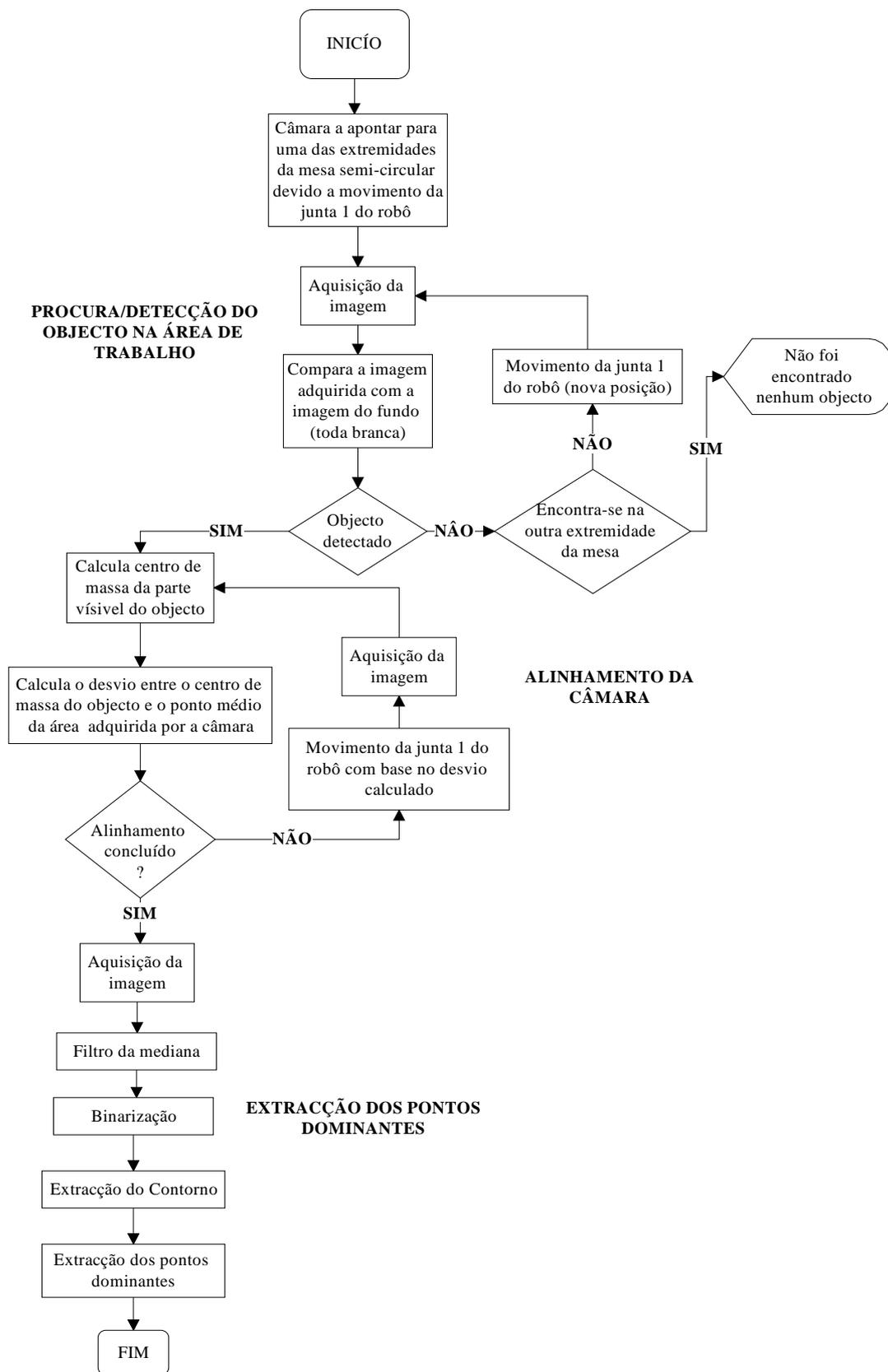


Figura 6.14 Fluxograma da sequência de acções atribuídas ao sistema de visão.

6.3.1 Resultados experimentais

A Figura 6.15 ilustra o comportamento do sistema de visão, para as diferentes fases de processamento, em três situações distintas.

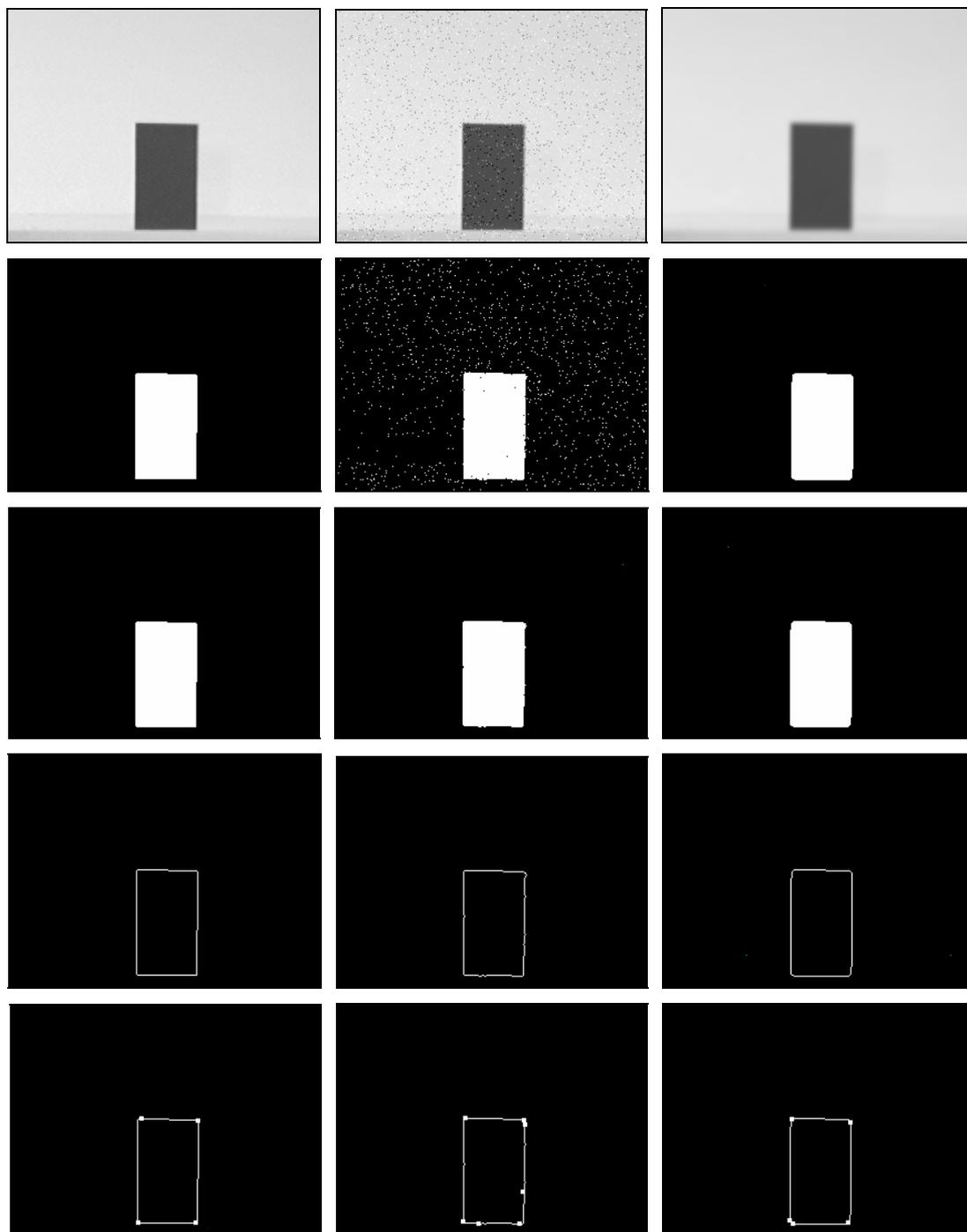


Figura 6.15 Comportamento do sistema de visão em três situações distintas. Por colunas, as imagens que constituem a situação normal, a situação resultante da introdução de ruído e a situação derivada de uma má focagem. Por linhas, as imagens resultantes da binarização, filtragem de mediana, descrição por *chain code* e da determinação dos pontos dominantes.

A primeira (imagens da primeira coluna) corresponde a uma situação de normal funcionamento do sistema. A segunda tem a ver com uma situação resultante da deliberada introdução de ruído por forma a se poder testar o seu efeito no comportamento do sistema.

Finalmente, a terceira situação é caracterizada pela aquisição de uma imagem resultante de uma má focagem da câmara. Note-se que a sequência de processamento filtragem da mediana/binarização foi substituída pela sequência inversa binarização/filtragem da mediana. No entanto, tendo em consideração que se está a aplicar uma filtragem de mediana e não outra técnica de processamento de imagem, tal facto não resulta em qualquer alteração no resultado final. Permite, simplesmente, observar de forma mais evidente o resultado desta filtragem.

Para as duas situações adversas testadas, observa-se, nomeadamente na determinação dos pontos dominantes, uma maior sensibilidade ao ruído. No entanto, o conjunto câmara/placa utilizado possibilita uma aquisição de imagens com baixo nível de ruído, pelo que as situações testadas acabaram por ter interesse meramente académico.

6.4 Sumário

Neste capítulo descreve-se o sistema de visão implementado, explicando-se ainda o porquê de algumas das soluções adoptadas. O objectivo final do sistema de visão consiste na restrição da área de trabalho do robô, permitindo posicioná-lo de seguida para a extracção do relevo. Os procedimentos com vista à concretização deste objectivo podem resumir-se nos seguintes passos:

1. Procura do objecto em toda a área de trabalho do robô por comparação de imagens.
2. Aquisição da imagem.
3. Aplicação do filtro da mediana para remoção de ruído inerente a qualquer processo de aquisição.
4. Binarização da imagem para separação desta em duas classes (fundo e objecto). Para estabelecer o nível de binarização foi utilizado o método de *Otsu*.
5. Extracção do contorno e sua representação através da técnica dos *chain codes*.
6. Estabelecimento dos pontos dominantes através da aplicação do método de *splitting* com distância incremental.
7. Selecção dos pontos dominantes através do método do desvio angular.

O sistema implementado apresenta algumas limitações, tais como: os objectos a processar terão que ter uma cor contrastante com o fundo da mesa de trabalho; o posicionamento fixo da câmara impede a obtenção da real dimensão dos objectos, uma vez que estes se podem posicionar a diferentes distâncias. Apesar das limitações referidas o sistema de visão mostrou-se robusto, nomeadamente ao apresentar alguma insensibilidade às condições de luminosidade, ruído e focagem imperfeita, conforme se comprova pelos resultados experimentais.

A primeira limitação não implicou qualquer problema para o desenvolvimento do trabalho em virtude de os modelos utilizados apresentarem suficiente contraste com o fundo da mesa. A segunda limitação fez com que não fosse possível restringir com suficiente precisão a área de trabalho do robô, tendo-se utilizado os sensores de ultrassons para complementar a informação obtida a partir do sistema de visão.

Capítulo 7

7. Aquisição do relevo

7.1 Introdução

A aquisição da 3D é um problema delicado, tendo sido abordado em muitos trabalhos, nomeadamente no campo da robótica.

No caso da robótica móvel, a representação do ambiente a 3D é uma tarefa complicada em virtude de este se encontrar em constante mutação devido ao movimento do robô. Na tentativa de obter uma boa representação foram testados vários tipos de sensores, combinações entre estes e diversas técnicas de combinação de dados.

Na robótica estática, o problema da representação também se coloca mas a um nível diferente, em virtude do ambiente em que o robô se insere ser mais estruturado, não estando por isso sujeito a grandes mutações. Contudo, essa representação tem que ser mais precisa do que no caso anterior de forma a permitir operações como as de soldadura, polimento, aplicação de colas, rebarbagem ou pintura.

Este capítulo descreve o método utilizado para aquisição do relevo da superfície visível de um objecto (para o robô), com base na combinação de dados adquiridos por diferentes sensores. Numa primeira fase os dados a combinar são provenientes de sensores de ultrassons e de uma câmara de vídeo; numa segunda fase são provenientes de sensores de ultrassons diferentes. Estes sensores foram montados no pulso do PUMA 560, dispostos em quadrado com uma distância entre os centros de aproximadamente 4 cm (Figura 7.1).

O capítulo está dividido em 4 partes. Na primeira parte é feito um breve resumo de alguns trabalhos nesta área. Na segunda parte introduz-se o modelo utilizado para os ultrassons. A terceira e quarta partes apresentam o procedimento de fusão/integração sensorial de dados que permite a descrição da superfície visível do objecto (dimensões e relevo).

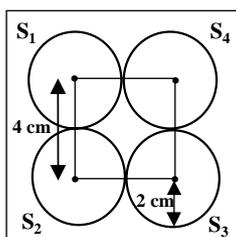


Figura 7.1 Vista frontal da disposição dos 4 sensores na garra do robô.

7.2 Introdução ao problema da aquisição do relevo

A aquisição, representação e análise do relevo de peças com base em dados provenientes de múltiplos sensores no contexto da robótica estática, são abordados por exemplo nos seguintes trabalhos: [Allen, 88], [Allen and Michelman, 90], [Nashman, 93] e [Tung, 96].

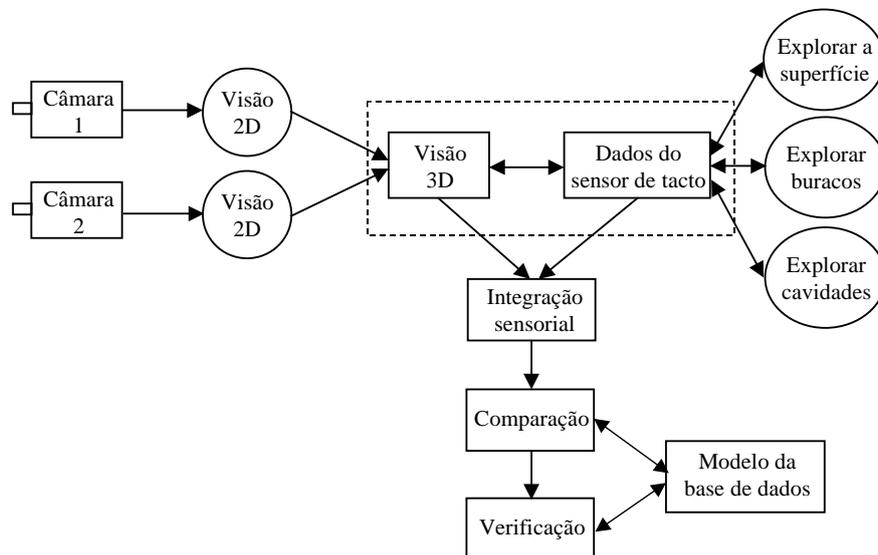


Figura 7.2 Perspectiva geral do sistema [Allen, 88].

No trabalho de Allen [Allen, 88] aborda-se o reconhecimento de objectos a partir da combinação de dados provenientes de um sistema de visão *stereo* passiva e da exploração efectuada por um sensor de tacto. Os objectos a serem reconhecidos estão fixos numa mesa de trabalho e a sua imagem é adquirida por um par de câmaras de vídeo. O sensor de tacto é montado no punho de um PUMA 560. Os objectos utilizados como modelos podem ter formas planas, ter volume, apresentar buracos ou ainda ter superfícies côncavas ou convexas. A Figura 7.2 apresenta uma perspectiva geral do sistema. O algoritmo de reconhecimento é composto pelos seguintes passos: 1) o sistema de visão adquire a imagem e identifica as regiões de interesse; 2) o sistema de tacto explora cada região identificada pelo sistema de visão; 3) os resultados de ambos são integrados para descrição da superfície; 4) a descrição obtida é comparada com a descrição dos modelos existentes em base de dados, sendo chamado o modelo mais consistente com a informação sensorial adquirida; 5) validação/verificação do modelo.

Em Allen e Michelman [Allen and Michelman, 90] o sistema sensorial é composto por uma “mão” com quatro “dedos”, cada um com 4 graus de liberdade. O sistema possui sensores de posição em cada junta para detectar a posição angular e sensores para medir a força em cada um dos tendões (extensores e flexores) que controlam as juntas. A mão foi montada no punho de um PUMA 560, ficando o sistema dotado com um total de 22 graus de liberdade. Em cada dedo foram montados sensores de tacto constituídos por material plástico piezoresistivo. O processo de reconhecimento comporta os seguintes passos: 1) O PUMA é movido para uma localização próxima de uma das extremidades do objecto a explorar; 2) o polegar e o indicador da mão são abertos o suficiente para permitir rodear o objecto sem entrar em contacto com ele; 3) o polegar é então movido lentamente até os sensores detectarem o contacto entre ele e o objecto; 4) o dedo indicador efectua o mesmo movimento; 5) as posições dos dois pontos de contacto são guardadas; 6) o PUMA e a mão são então movidos lentamente segundo o eixo do objecto a explorar até que o movimento de um dos

dedos falhe o contacto. A localização do objecto e dos seus eixos é determinada através de ajuda humana.

No trabalho de Nashman [Nashman, 93] pretende-se realizar tarefas de controlo de qualidade, mais concretamente a verificação/validação da geometria de peças mecânicas torneadas. O sistema inicialmente utilizado era composto por um robô cartesiano (CMM - *Coordinate Measuring Machine*) de grande precisão com três graus de liberdade. Este tipo de robôs é frequentemente utilizado nestas tarefas. O processo de verificação/validação envolvia a medida de características da superfície da peça torneada, sendo estes valores comparados com os valores especificados no modelo pré-definido num CAD (*Computer Aided Design*) ou num modelo guardado em base de dados. As peças eram aceites ou rejeitadas com base no erro entre os valores medidos e os valores especificados no modelo. Com a finalidade de simplificar o processo de medida e aumentar a velocidade e a flexibilidade da CMM mantendo a precisão, Nashman estudou um sistema de interacção entre uma câmara de vídeo e uma sonda de tacto. O sistema de visão fornece estimativas da posição de características de interesse da peça (por exemplo: saliências, sulcos, diâmetros, buracos e contornos). A partir da utilização de visão em tempo real a sonda é guiada para as características de interesse, efectuando as medidas para o processo de verificação/validação. A título de exemplo, enquanto a sonda varre toda a superfície da peça o seu movimento é controlado pela informação proveniente de três fontes: câmara, escalas da máquina e da própria sonda. O sistema de visão fornece informação acerca da posição de buracos, saliências, sulcos e outras características consideradas relevantes. As escalas da máquina, quando usadas juntamente com a visão, fornecem a distância entre a sonda e essas características. Os dados da sonda fornecem a distância entre esta e a superfície da mesa.

Em Tung [Tung, 96] utiliza-se um PUMA 762 para a execução da montagem de um conjunto de peças atiradas de forma aleatória para a sua área de trabalho. Para realização desta tarefa foi usado um sistema sensorial composto por um sensor de força/binário, um dispositivo pneumático montado no punho do robô para auxiliar no processo de montagem e duas câmaras de vídeo. Uma das câmaras encontra-se fixa numa plataforma em frente ao robô, permitindo visualizar toda a área de trabalho. A segunda está montada na garra, sendo usada para determinar a posição e orientação das peças. A orientação da garra é determinada pela informação obtida pelas câmaras.

Nos trabalhos acima mencionados, a aquisição do relevo é realizada sempre com recurso ao contacto. Esta opção apresenta um problema (abordado com frequência na literatura da especialidade) que é a transposição do espaço livre para a situação de contacto, para além das dificuldades que o movimento do sensor de contacto apresenta no seu deslocamento ao longo de qualquer superfície.

Em alternativa é possível adquirir o relevo sem contacto, através do recurso a visão *stereo* ou a sensores de distância (por exemplo: laser, infravermelhos e ultrassons), enquadrando-se este trabalho nesta perspectiva.

7.3 Modelo para o sensor de ultrassons

Num modelo simples para os ultrassons, a gama de medida pode ser interpretada como a distância ao obstáculo mais próximo na direcção do centro do feixe. Este modelo é no entanto impreciso por diversas razões. Em primeiro lugar falha em superfícies especulares, isto é, que reflectem a onda sonora como um espelho. Estas reflexões ocorrem quando as dimensões dos corpos reflectores ultrapassam o comprimento de onda do feixe incidente. Neste caso aplica-se a lei bem conhecida da óptica: o ângulo de reflexão é igual ao ângulo de incidência. Ou seja, corre-se o risco de o feixe reflectido não mais encontrar o sensor excepto se o feixe

incidente fazer um ângulo de 90° com a superfície. Em segundo lugar, para os ângulos de incidência oblíquos, ou nenhum retorno é detectado, ou quando é detectado resulta de reflexões sucessivas originando uma medida errada. Finalmente, não tem em consideração a largura do feixe, aceitando todas as reflexões detectadas sem ter em consideração se estas tiveram ou não origem em objectos situados na periferia do feixe.

O mapa de percepção com base numa grelha proposto por Elfes [Elfes, 87], utiliza um modelo para os ultrassons que impõe uma largura finita para o feixe sem ter em consideração as irregularidades angulares nem as superfícies especulares. Para modelo do feixe são utilizadas funções de densidade de probabilidade. O pré-processamento dos dados dos ultrassons incluiu o estabelecimento de um limite mínimo e de um máximo para a sua aceitação. Este modelo produziu bons resultados em ambientes sem superfícies especulares, tendo falhado completamente quando elas existiam.

Kuc e Siegel [Kuc and Siegel, 87] projectaram um modelo para os ultrassons com base nos princípios da acústica, no conhecimento do circuito de detecção da Polaroid e no comportamento dos ultrassons, nomeadamente a sua reflexão em superfícies especulares, cantos e bordas. A conclusão principal foi que a utilização deste modelo para pequenos ângulos de incidência (menor ou igual a 6°) a medida obtida a partir da onda reflectida é sempre igual à distância na perpendicular à superfície reflectora, independentemente do ângulo de incidência.

Leonard e Durrant-Whyte no capítulo 2 do seu livro [Leonard and Durrant-White, 92] fazem uma boa descrição dos efeitos causados pela irregularidade angular. Eles recolheram conjuntos de leituras a partir do varrimento de superfícies especulares com pequenas incrementações no ângulo de incidência (0.588°) até atingir um ângulo máximo de 30° . Para ângulos de incidência pequenos, onde a intensidade do feixe é grande, encontraram sequências de leituras adjacentes que definem segmentos de recta horizontais. Essas sequências foram denominadas de “regiões de profundidade constante” (*regions of constant depth-RCD*), correspondendo a arcos em coordenadas cartesianas. A extracção destas regiões é efectuada com base na diferença entre um valor mínimo e um máximo de uma sequência de leituras adjacente ser inferior que um limite pré-estabelecido ($|V_{\max} - V_{\min}| < \delta_r$). A largura β de uma RCD é a diferença entre o ângulo da leitura efectuada mais à esquerda e o ângulo da leitura efectuada mais à direita. É pré-definido um valor mínimo para o β , normalmente entre 5° e 10° . Esta largura permite fazer a distinção entre “retornos fortes” e “retornos fracos”: considera-se retornos fracos se $\beta < \beta_{\min}$ e retornos fortes se $\beta \geq \beta_{\min}$. Só os retornos fortes são processados, sendo os fracos totalmente ignorados.

Harris e Recce em [Harris and Recce, 98] descrevem um modelo para os ultrassons resultante da análise de um grande conjunto de dados obtidos com um ultrassom da Polaroid. A recolha dos dados foi efectuada a partir de uma posição fixa com um espaçamento angular de 1.8° . Foram definidos dois modelos: um para superfícies rugosas e um para superfícies lisas. No primeiro caso, o modelo permite obter a média μ e o desvio σ das leituras efectuadas, em função da distância normal d e de um ângulo de incidência θ_b . No segundo caso o modelo baseia-se num conjunto de valores tabelados, com uma entrada - o ângulo de incidência - e três saídas: a probabilidade da direcção da onda de retorno, a estimativa da média dos retornos directos (média dos retornos directos menos a distância na direcção da normal) e o desvio dos retornos directos.

7.3.1 Modelo utilizado

Na especificação do modelo a implementar foram impostas à partida 2 condições:

1. Os ultrassons teriam que ser o mais direccionais possível de forma a distinguir com a maior precisão as diferenças de profundidade e a sua localização.
2. Todas as medidas que não pertencessem ao intervalo [25 cm, 80 cm] podiam ser ignoradas.

Tendo como ponto de partida os estudos feitos e as condições acima referidas foram efectuados testes experimentais. Como superfície reflectora foram usados objectos que posteriormente foram utilizados como modelos.

Uma das conclusões dos testes confirmou os estudos efectuados, ou seja, para ângulos de incidência pequenos (menor ou igual a aproximadamente 8°) a probabilidade de termos um retorno directo é máximo. Neste caso, a medida obtida a partir da onda reflectida é igual à distância na perpendicular à superfície reflectora e a sua precisão é a melhor que se pode obter com os ultrassons.

Por outro lado, verificou-se algumas vezes que os valores obtidos a partir dos ultrassons com uma posição fixa tinham flutuações mais ou menos relevantes consoante o ponto da superfície para onde apontavam. Curiosamente este tipo de resultados nunca foi referido nas publicações anteriormente nomeadas, provavelmente por não terem tido necessidade de efectuar este tipo de teste. Como para este trabalho era importante esclarecer o problema, foram efectuados três tipos de teste sempre com o ultrassom fixo numa posição e utilizando como superfície reflectora um dos objectos posteriormente utilizado como modelo: 1) com o feixe dos ultrassons perpendicular a uma região plana do modelo (assinalada com **1** na Figura 7.3) e variando o ângulo de incidência até um máximo de aproximadamente 8° ; 2) com o feixe de ultrassons perpendicular a uma região plana do modelo mas a apontar para um ponto situado nos limites da objecto (assinalado com **2** na Figura 7.3); 3) com o feixe de ultrassons perpendicular a uma região plana mas a apontar para uma zona próxima de uma transição de profundidade (assinalada com **3** na Figura 7.3).

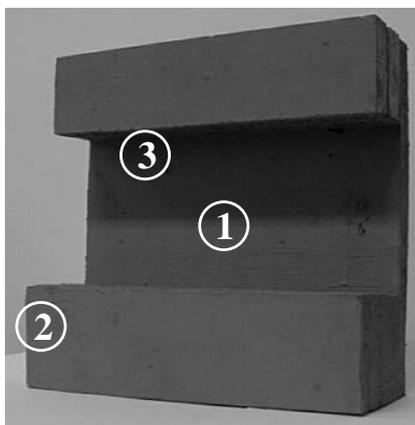


Figura 7.3 Modelo utilizado nos testes.

Os resultados obtidos foram os seguintes:

Ultrassom na posição 1

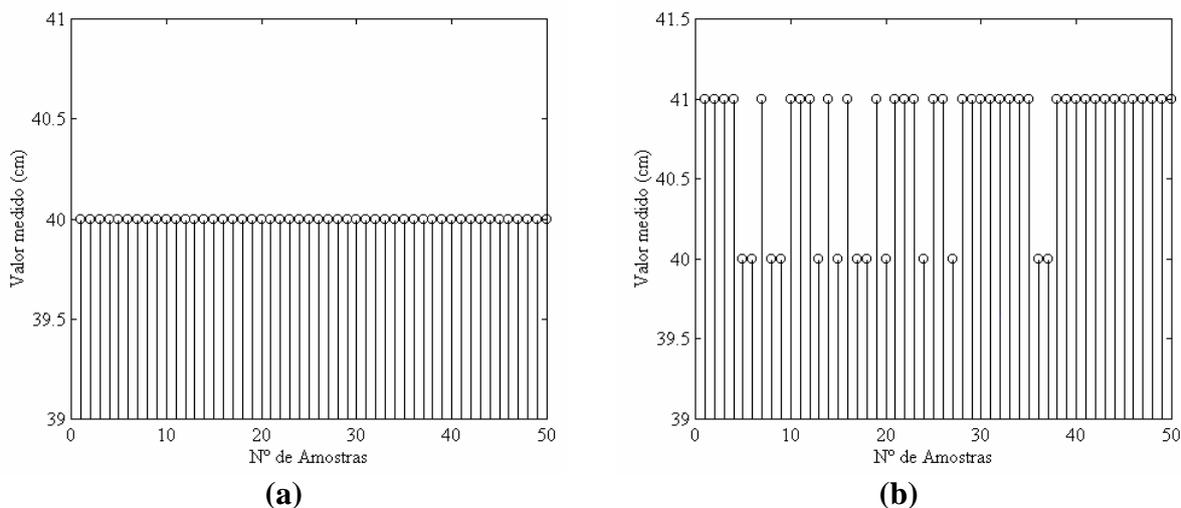


Figura 7.4 Dados do ultrassom quando posicionado de forma a apontar para o ponto 1. (a) Normal à superfície; (b) Com um ângulo de aproximadamente 8° relativamente à superfície.

Ultrassom nas posições 2 e 3

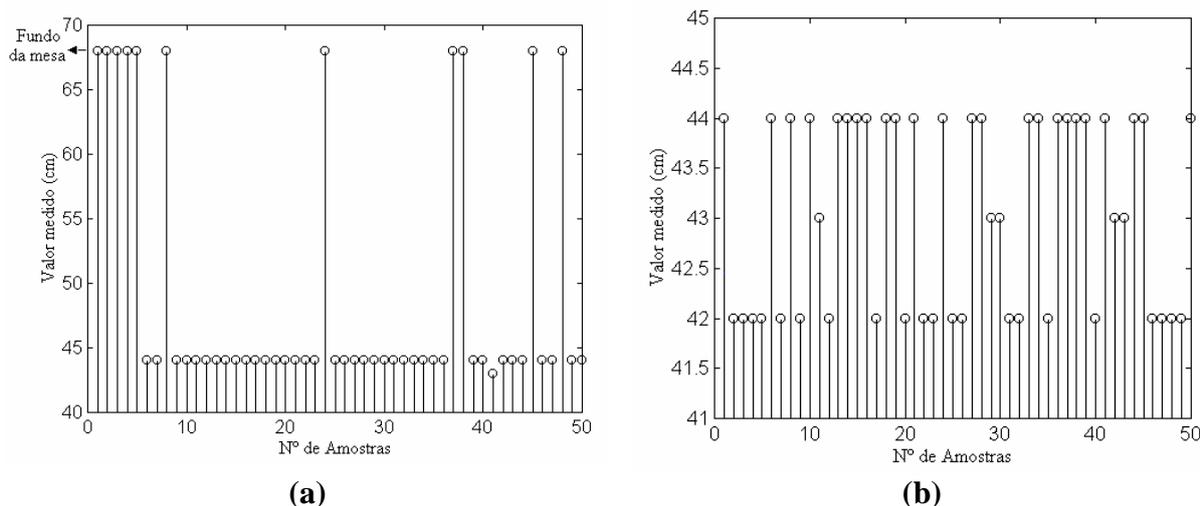


Figura 7.5 Dados do ultrassom. (a) ultrassom a apontar para o ponto 2; (b) ultrassom a apontar para o ponto 3.

As variações de medida verificadas na Figura 7.4 b) não são significativas, pois devem-se essencialmente ao arredondamento efectuado no cálculo da distância, podendo-se considerar que até um ângulo menor ou igual a 8° os valores obtidos praticamente não têm flutuação (menor ou igual a 1 cm). Na Figura 7.5 a) os valores mais altos (68 cm) correspondem à distância ao fundo da mesa onde os objectos são posicionados e as outras medidas correspondem à distância real ao objecto. Na Figura 7.5 b) as variações efectuadas devem-se a pequenas reflexões em virtude da proximidade de zonas com diferentes profundidades.

A partir da análise destes resultados foram definidos dois conceitos: **Pontos de profundidade constante (PPC)** e **Regiões de profundidade constante (RPC)**. Os PPC são pontos que representam uma determinada área, que está relacionada com o diâmetro do sensor de ultrassons e com o ângulo de abertura do respectivo feixe. Contudo, em termos práticos o ponto considerado corresponde ao apontado pelo centro do sensor na direcção da

normal à superfície. As RPC são regiões cuja dimensão máxima corresponde a um quadrado de 4 cm de lado. Este valor é imposto pela distância entre os centros dos sensores de ultrassons, podendo no entanto haver RPCs com valores inferiores e uma configuração rectangular. A configuração em quadrado ou rectângulo das RPC resultam da disposição dos sensores de ultrassons no punho do robô.

Pontos de profundidade constante (PPC)

Um vector de medidas adquirido a partir de um único ultrassom fixo define um ponto de profundidade constante quando a diferença entre o valor máximo e a média desse vector são menores que um determinado limite pré-estabelecido. Isto é,

$$\text{se } \text{MAX}_{\text{vector de medida}} - \bar{x}_{\text{vector de medida}} < \delta \Rightarrow \text{temos um PPC} \quad (7.1)$$

com

$$\bar{x}_{\text{vector de medida}} = \frac{1}{n} \sum_{j=1}^n x_j \quad n - \text{N}^\circ \text{ de elementos do vector de medida}$$

No caso desta aplicação o δ utilizado é igual a 1 cm e a distância atribuída ao PPC é a média do vector de medida.

O estabelecimento desta definição permitiu fazer a distinção entre os retornos fortes e fracos dos ultrassons. Consideram-se retornos fortes aqueles que permitem definir um PPC e fracos os restantes, sendo unicamente processados os retornos fortes.

Regiões de profundidade constante (RPC)

As regiões de profundidade constante são extraídas com base nos PPC. Neste caso são utilizadas sempre os quatro PPC correspondentes aos quatro ultrassons utilizados.

Considera-se que uma determinada região constitui uma RPC quando a diferença em valor absoluto entre as médias resultantes dos PPC (fazendo todas as combinações possíveis) não ultrapassa um determinado limite pré-estabelecido. Se a seguinte condição

$$\begin{aligned} & \left| \bar{x}_{\text{sensor1}} - \bar{x}_{\text{sensor2}} \right| < \delta_R \quad \text{e} \quad \left| \bar{x}_{\text{sensor1}} - \bar{x}_{\text{sensor3}} \right| < \delta_R \quad \text{e} \quad \left| \bar{x}_{\text{sensor1}} - \bar{x}_{\text{sensor4}} \right| < \delta_R \quad \text{e} \\ & \left| \bar{x}_{\text{sensor2}} - \bar{x}_{\text{sensor3}} \right| < \delta_R \quad \dots \end{aligned}$$

se verificar, então os PPC definem uma RPC, cuja profundidade é igual à média das médias dos PPC (expressão 7.2).

$$\bar{x}_{RPC} = \frac{1}{n_{PPC}} \sum_{j=1}^{n_{PPC}} \bar{x}_{j(PPC)} \quad n_{PPC} - \text{N}^\circ \text{ de PPC} \quad (7.2)$$

No caso desta aplicação o δ_R é igual a 1.5 cm. A utilização deste valor resultou dos testes experimentais efectuados e da incapacidade de distinguir com os ultrassons diferenças de profundidade inferiores aproximadamente a 1.8 cm.

7.4 Determinação das dimensões do objecto

A limitação do sistema de visão em adquirir a dimensão efectiva dos objectos já referida no Cap. 6, impôs a utilização dos sensores de ultrassons de forma a obter-se com maior precisão as reais dimensões do objecto (largura x altura). Para alcançar esse objectivo utilizou-se a **fusão sensorial dos dados** provenientes de dois tipos de sensores: ultrassons e câmara de vídeo.

A interacção entre os dados provenientes destes sensores é do tipo **cooperativo** (descrita anteriormente – Cap. 2), pois os dados resultantes do processo de visão (pontos dominantes) são utilizados para posicionar o robô, de forma a que os ultrassons montados no seu punho fiquem a apontar para as zonas próximas dos limites da peça. Os dados dos ultrassons complementam a informação já existente, obtendo-se deste modo um modelo inicial mais rigoroso do ambiente.

Após a obtenção dos pontos dominantes do contorno do objecto, a complementação dessa informação para uma determinação mais rigorosa dos limites do objecto envolve as seguintes tarefas:

1. Posicionamento do manipulador de forma a que o ultrassom 1 fique a uma distância da superfície do objecto igual a 35 cm.
2. Procura do limite inferior esquerdo (vista frontal do objecto).
3. Procura do limite inferior direito.
4. Procura do limite superior da peça.

O algoritmo correspondente à execução da primeira tarefa é o que se apresenta na Figura 7.6.

Inicialmente o robô é posicionado a uma altura fixa relativamente ao chão (imposta pela altura da mesa que serve de base aos objectos) e a uma distância igualmente fixa do centro de massa do objecto (acção 1).

A redefinição do referencial do robô (acção 2) é realizada para garantir o deslocamento deste num plano paralelo ao plano de projecção da imagem do objecto, qualquer que seja a sua localização sobre a mesa.

O valor de referência (35 cm) referido na acção 5 foi igualmente determinado de uma forma experimental tendo em consideração os seguintes dois pontos:

1. O valor mínimo mensurável com precisão pelos ultrassons utilizando esta configuração de *hardware* é aproximadamente 30 cm. A utilização deste valor como referência é contudo contraproducente porque é fácil surgir distâncias inferiores provocando a aquisição de medidas erradas.
2. Um compromisso entre a direccionalidade e o ângulo de abertura do feixe. Para se obter a máxima direccionalidade era conveniente a utilização como valor de referência da distância mínima mensurável. Contudo, pretendia-se uma maior sobreposição dos feixes dos ultrassons de forma a garantir uma melhor cobertura de uma região de 4 x 4 sem grande prejuízo da direccionalidade, tendo-se então optado pela utilização de um valor de referência igual a 35 cm. É ainda de salientar que foi testado um valor de referência igual a 40 cm, tendo-se observado uma detecção antecipada (relativamente à distância de 35 cm) dos pontos onde se verificavam as transições de profundidade existentes numa superfície, o que originava mapas de superfície com maior erro.

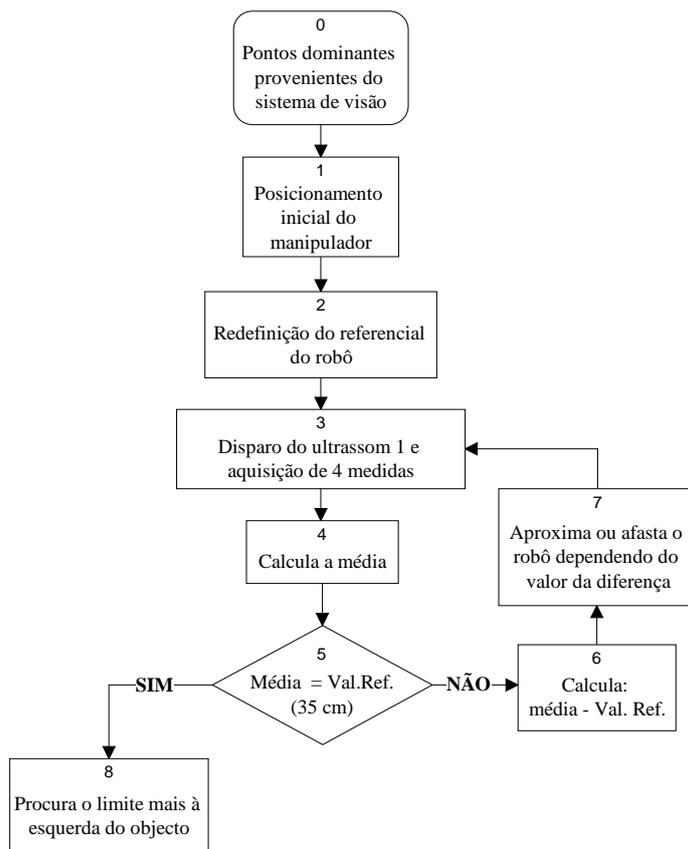


Figura 7.6 Posicionamento do manipulador com o ultrassom 1 a 35 cm da superfície do objecto.

Após o posicionamento do robô tem início a procura do limite inferior esquerdo do objecto (Figura 7.7 (a)) através da utilização de um único sensor (ultrassom 1). Esta procura é efectuada incrementando a posição do robô de 0.5 cm na horizontal até se encontrar o limite pretendido.

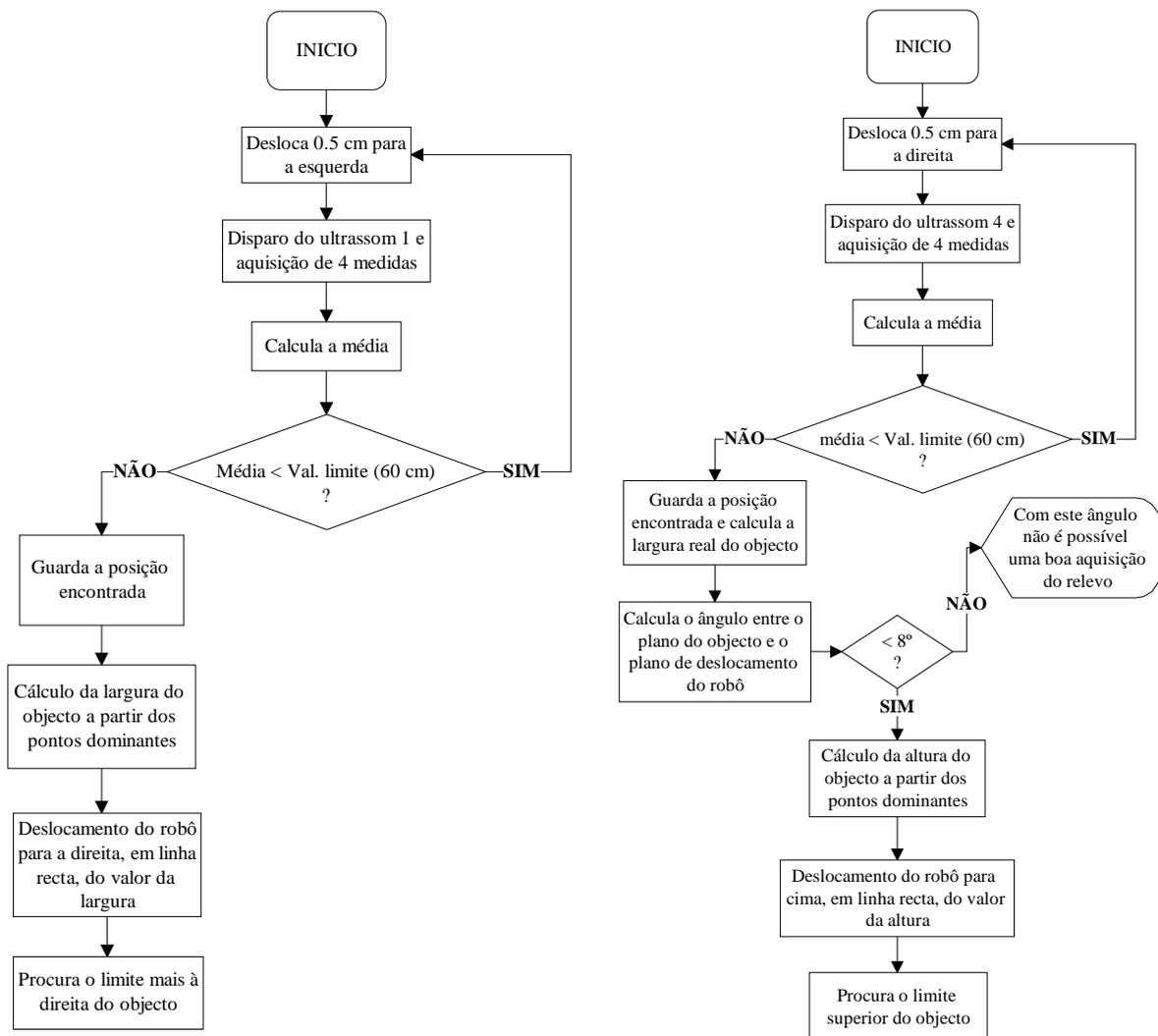


Figura 7.7 Fluxograma da procura dos limites laterais do objecto. (a) Procura limite esquerdo; (b) Procura o limite à direita.

A obtenção do limite inferior direito e do limite superior é efectuada de forma semelhante ao caso anterior (algoritmo das Figura 7.7 (b) e Figura 7.8). Após a obtenção do limite inferior direito é calculado o ângulo entre o plano do objecto e o plano de deslocamento do robô. Esse ângulo é determinado com base na largura real do objecto e na diferença de profundidade existente entre o limite inferior esquerdo e o limite inferior direito. Esta forma de efectuar o cálculo impõe a seguinte restrição: os dois limites inferiores do objecto têm que ter o mesmo relevo, isto é, a diferença de profundidade medida pelo ultrassom tem que se dever exclusivamente à inexistência de paralelismo entre o plano do objecto e o plano de deslocamento do robô. No caso de o ângulo determinado ser superior a 8° , o varrimento da superfície do objecto não é efectuada, em virtude de as condições para aquisição do relevo com rigor não estarem satisfeitas.

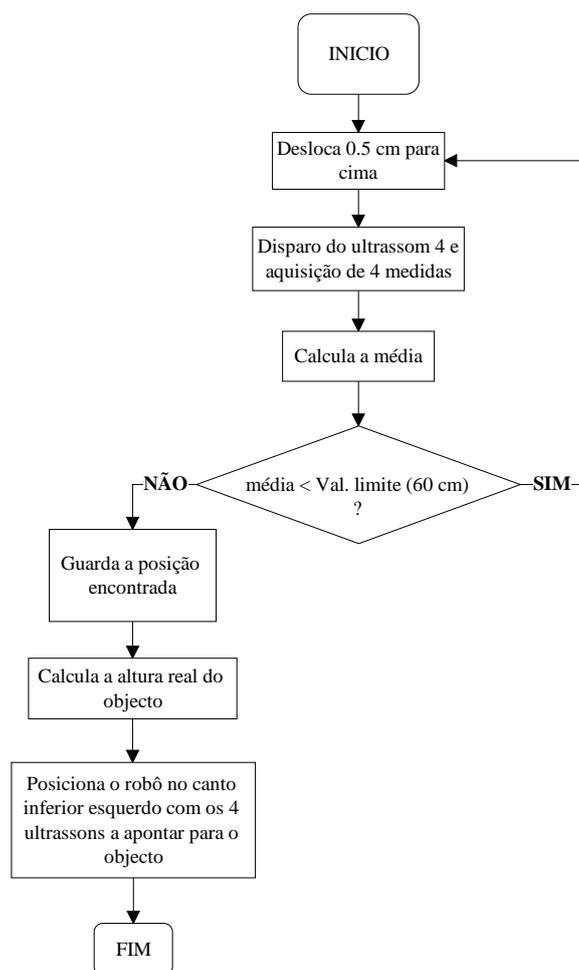


Figura 7.8 Fluxograma da procura do limite superior do objecto.

Após a determinação das dimensões do objecto tem início o varrimento da sua superfície visível para aquisição do relevo.

7.5 Aquisição do relevo da superfície visível

No processo de aquisição do relevo estão sempre em avaliação os quatro pontos candidatos à definição de uma RPC.

Na Figura 7.9 apresenta-se a sequência de acções a executar para se efectuar a aquisição do relevo da superfície visível do objecto. É de salientar que este processo de aquisição é relativamente lento devido essencialmente à baixa velocidade da onda sonora e ao número de

medidas que é necessário efectuar para extrair uma RPC. Como o número mínimo de medidas é 40 e cada aquisição demora 6 ms, o tempo mínimo para extrair uma RPC é de 240 ms.

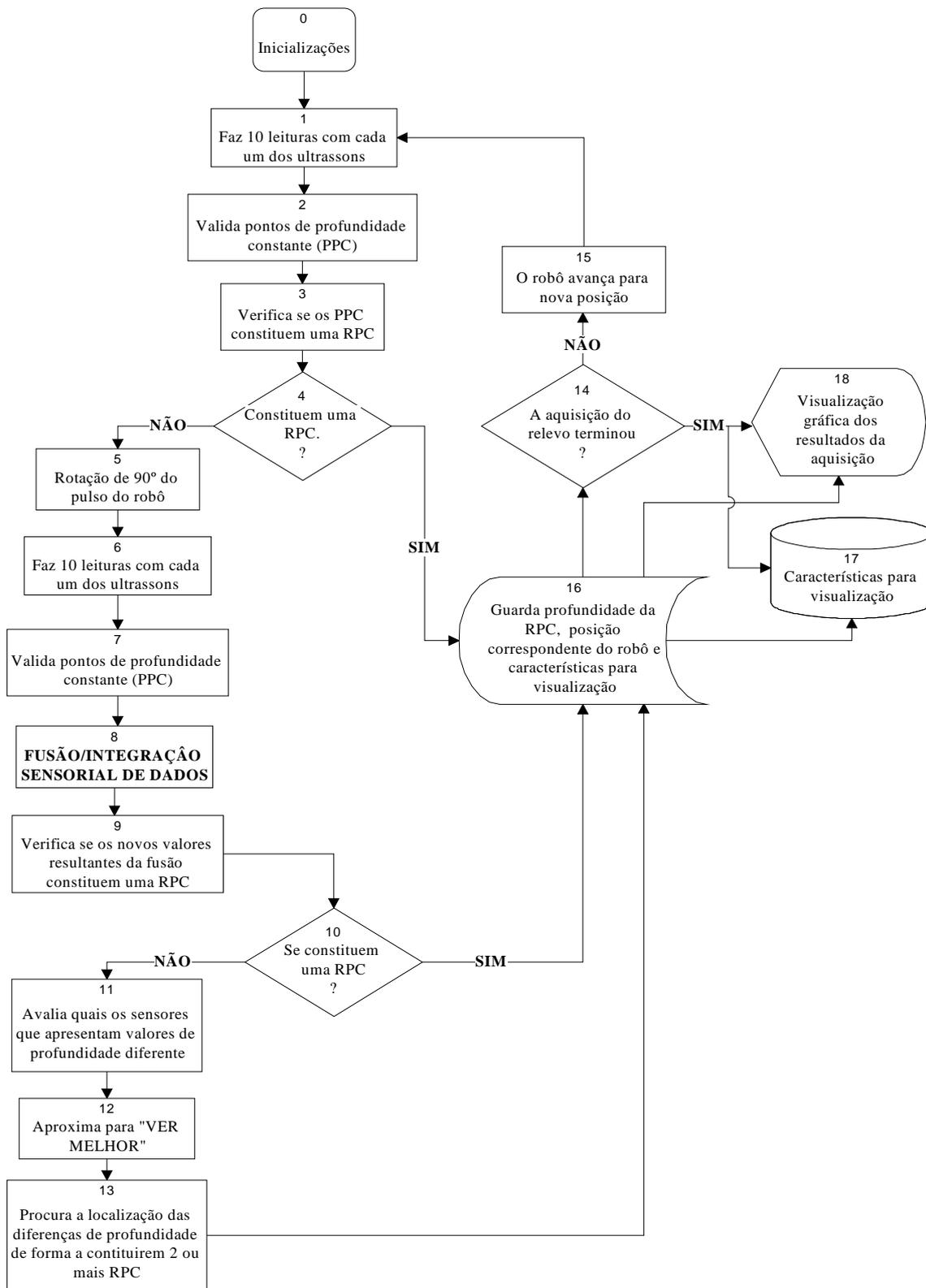


Figura 7.9 Fluxograma da aquisição do relevo.

Em seguida explica-se com mais detalhe as acções que correspondem à aquisição do relevo para uma melhor compreensão de todo o processo.

Acção 1: Corresponde à leitura dos 4 sensores de ultrassons, sendo efectuadas 10 medidas com cada sensor. As 10 medidas serão denominadas a partir deste momento por **vector de medida**. A dimensão adoptada para o vector teve como base o resultado de testes experimentais. Estes permitiram concluir que uma dimensão aproximadamente igual a 10 facilitava a obtenção de um PPC com maior rigor, uma vez que reduzia a influência do aparecimento de um valor de medida esporádico superior ou inferior em algumas unidades (3 a 4 unidades) aos restantes elementos. Esta influência era particularmente notada quando surgia um PPC com um valor de profundidade superior aos restantes originando os seguintes inconvenientes:

- A condição para extrair uma RPC era falsa, impedindo a sua obtenção de imediato.
- O resultado do processo de fusão sensorial nesse ponto é afectado, podendo posteriormente reflectir-se na impossibilidade de extrair a RPC. Uma situação deste tipo ocasionava a procura de uma região de diferente profundidade inexistente na superfície.
- Perdas de tempo no processo de aquisição.

Acções 2 e 3: Tem como objectivo a validação dos PPC e a posterior extracção da RPC segundo o processo anteriormente referido.

Acção 4: No caso de estarmos perante uma RPC, é guardada a informação necessária à caracterização da superfície, assim como a localização do robô na posição em que foram adquiridos esses dados; no caso oposto (não estarmos perante uma RPC) segue-se para o processo de fusão sensorial para obtenção de novos valores para os pontos candidatos a formarem uma RPC.

Acção 5: Efectua uma rotação de 90° do pulso do robô de forma a colocar novos sensores a apontar para os 4 pontos candidatos a constituírem uma RPC. Este procedimento deve-se a que sensores diferentes a apontar para o mesmo ponto não tem exactamente o mesmo comportamento.

Acções 6 e 7: Fazem a aquisição de novos valores e a validação destes como PPC.

Acção 8: Corresponde ao processo de fusão sensorial ponto a ponto. Isto é, em cada um dos 4 pontos é realizada fusão sensorial entre as medidas adquiridas pelos dois sensores que apontaram para esse ponto. A interacção entre os dados é do tipo **competitivo** (descrita anteriormente – Cap. 2). É ainda de salientar que a técnica de fusão usada em cada ponto pode não ser igual para todos os pontos, dependendo de uma avaliação prévia dos valores adquiridos.

A integração sensorial de dados (função “transformação de dados”) só é efectuada no caso em que os sensores apontam para um dos limites do objecto.

O processo de fusão/integração sensorial é explicado com detalhe no ponto seguinte.

Acção 9: Verifica-se se após o processo de fusão sensorial já é possível extrair uma RPC.

Acção 10: No caso de estarmos perante uma RPC, é guardada a informação necessária à caracterização da superfície assim como a localização do robô na posição em que foram

adquiridos esses dados; no caso oposto (não é uma RPC) segue para a avaliação de qual/quais os sensores que apresentam valores de profundidade diferente.

Acções 11, 12 e 13: Na Acção 11 é realizada uma avaliação de qual/quais os sensores que apresentam valores de profundidade diferentes (Tabela 7-1). O número de profundidades diferentes indica o número máximo de sub-regiões (RPC) que são passíveis de detectar numa região com as dimensões de 4x4 (dimensões máximas de uma RPC). Partindo desta avaliação é tomada a decisão de qual/quais os sensor/es a ficarem activos para procurar o local onde ocorre a diferença de profundidade.

Na acção 12 é efectuada uma aproximação para “VER MELHOR”. Esta aproximação coloca o sensor que vai detectar a diferença de profundidade a uma distância de 35 ± 1 cm (35 cm valor de referência), com o objectivo de minimizar o ângulo de abertura do feixe e a probabilidade de reflexões.

A acção 13 corresponde à procura do ponto onde se encontra a diferença de profundidade. Esta procura é efectuada através de deslocamentos do robô de 0.5 cm num percurso máximo de 4 cm (correspondente ao valor máximo do lado de uma RPC).

Acção 14: Verifica-se se o varrimento da superfície já terminou. Em caso afirmativo é então possível a visualização. No caso contrário continua o varrimento.

Acções 15, 16 e 17: São acções de armazenamento e visualização dos resultados.

Sensor 1	Sensor 2	Sensor 3	Sensor 4	Nº de profundidades diferentes detectadas
x	x	x	x	0
x	x	o	o	2
x	o	o	x	2
x	o	o	o	1
o	x	o	o	1
o	o	x	o	1
o	o	o	x	1
x	x	o	—	3
x	o	—	x	3
—	o	x	x	3
o	x	x	—	3

Tabela 7-1 Detecção de diferentes profundidades como consequência dos PPC obtidos (os sensores que apresentam o mesmo símbolo significa um valor para o PPC igual).

7.5.1 Fusão/Integração sensorial de dados provenientes de ultrassons

A fusão sensorial é efectuada com os dados provenientes de dois ultrassons que apontaram para o mesmo ponto. A técnica de fusão sensorial a usar pode variar de ponto para ponto, dependendo de uma pré-avaliação efectuada que toma em consideração os valores previamente calculados da média e da variância dos diferentes vectores de medida. É atribuído valor zero à média e à variância sempre que o vector de medida não constitua um PPC.

A Figura 7.10 apresenta o fluxograma do processo de fusão/integração sensorial de dados. Foram implementadas três técnicas de fusão sensorial: Média, Filtro de *Kalman* e Lógica

Fuzzy. É ainda de salientar que os resultados finais dos 4 processos de fusão aplicados (um para cada ponto) podem ser modificados através de uma heurística, caso os ultrassons apontem para qualquer dos limites do objecto. Desta forma é possível considerar o método heurístico como uma técnica de integração, tendo em consideração a definição apresentada no Cap. 2.

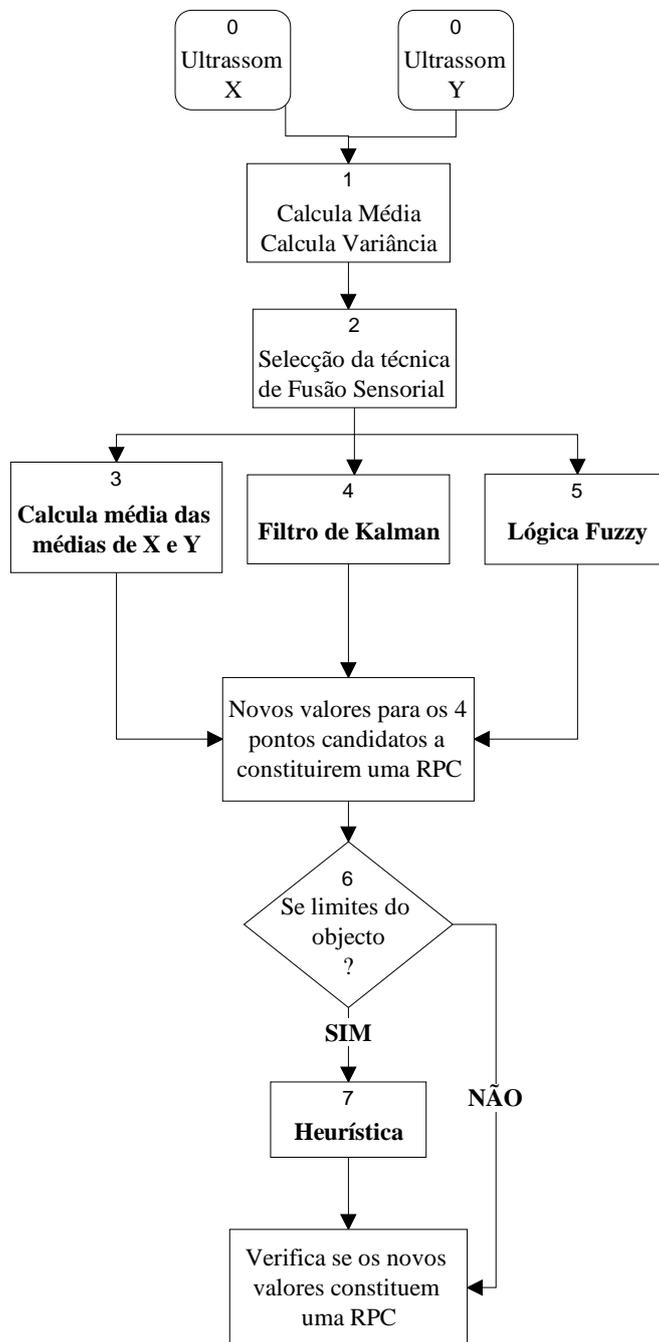


Figura 7.10 Fluxograma do processo de fusão sensorial de dados.

O algoritmo que faz a selecção da técnica de fusão sensorial (acção 2) a utilizar é descrito em pseudo código da seguinte forma:

1. início
2. m_X = média de X
- m_Y = média de Y

```

v_X = variância de X
v_Y = variância de Y
3. se ((m_X ≠ 0) e (m_Y ≠ 0)) então
    se ( |m_X - m_Y| > 1.5) então
        resultado = Fuzzy
    senão
        se ((v_X ≠ 0) ou (v_Y ≠ 0)) então
            resultado = Filtro de Kalman
        senão
            resultado = Média das médias de X e Y (  $\frac{médiaX + médiaY}{2}$  )
    senão
        se (m_X = 0) então
            resultado = m_Y
        senão
            resultado = m_X
fim

```

Em seguida apresenta-se com detalhe as técnicas de fusão sensorial referidas, as situações em que são aplicadas, assim como a técnica de integração implementada.

O “Decisor” *Fuzzy*

O “decisor” *fuzzy* é aplicado nas situações em que a diferença entre as médias dos dois sensores que apontaram para o mesmo ponto são superiores a 1.5 cm, sendo denominadas a partir deste momento de **médias divergentes**. O valor utilizado para referência é o mesmo que é utilizado para extrair as RPC, tendo sido determinado experimentalmente. Esta situação ocorre com frequência em pontos como o número 3 da Figura 7.3. Nesta situação colocava-se a questão de decidir qual dos valores seria o “mais correcto” (qual deveria ter maior peso no resultado final?). A escolha entre estes dois valores pode ser decidida tendo em consideração a informação adquirida por os sensores nos pontos vizinhos P₂ e P₄ (Figura 7.11) A informação destes pontos pode ser utilizada para atribuir uma quantidade de crença a cada uma das medidas divergentes. Considerando as medidas convergentes dos sensores vizinhos como correctas (X₂, Y₂, X₄ e Y₄) a crença será tanto maior quanto menor for a diferença para as medidas P₁. O cálculo desta crença poderia ser feito por um algoritmo exacto mas o recurso à lógica *fuzzy* permite uma melhor visualização e mais fácil sintonização através de um simples ajuste das funções pertença.

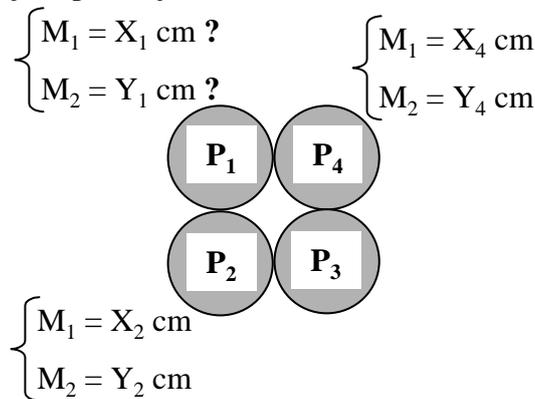


Figura 7.11 Exemplo da situação em que ocorre médias divergentes (ponto 1).

Para implementar o que foi descrito anteriormente utilizou-se um “decisor” *fuzzy* projectado com a ajuda da ferramenta *fuzzyTECH*. Esta cobre todos os passos necessários à implementação de um controlador/“decisor” *fuzzy*: definição da estrutura, definição das variáveis linguísticas, definição de regras e *debugging* interactivo [Inform Software Corporation, 96] [Altrock, 95]. Após o desenvolvimento do “decisor” *fuzzy* e seu afinamento permite ainda gerar o código ANSI C correspondente, tornando desta forma fácil a integração da lógica *fuzzy* com o código da restante implementação. É ainda de referir que esta ferramenta possibilita a utilização da teoria descrita no Cap. 2 (em 2.2.3).

O primeiro passo consistiu em definir a estrutura do “decisor” *fuzzy* numa janela denominada *Project Editor* do *fuzzyTECH*. É aqui que se define as entradas e saídas do sistema e a sua interacção. A partir da definição desta estrutura é depois possível aceder directamente às variáveis linguísticas e à definição das regras. A Figura 7.12 apresenta a estrutura do “decisor” *fuzzy* implementado. Nesta é possível distinguir, para além da entrada “diferença para a vizinhança” (*dif_viz*) e da saída “grau de confiança” (*Grau_Confia*), a estratégia de inferência e o método de desfuzificação utilizados. Como estratégia de inferência utilizou-se a MAX-MIN (MAX na acumulação e MIN na activação) e o método de desfuzificação escolhido foi o CoM, ambos explicados com detalhe no Cap.2 (em 2.2.3).

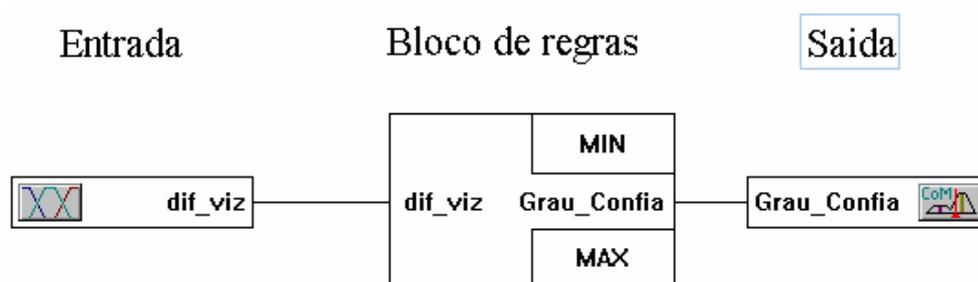


Figura 7.12 Estrutura do “decisor” *fuzzy*.

O segundo passo tem a ver com a definição das variáveis linguísticas e das funções de pertença, o que se consegue realizar facilmente através de uma interface gráfica apropriada. A Figura 7.13 apresenta as funções de pertença para a entrada “*dif_viz*”. Estas funções foram definidas tendo em consideração a diferença em valor absoluto das médias divergentes relativamente às médias apresentadas pelas vizinhanças laterais, nunca sendo utilizada a vizinhança na diagonal. Foram definidas 5 funções de pertença triangulares para caracterizar a entrada “*dif_viz*”: zero (ZE), positivo pequeno (PP), positivo médio (PM), positivo grande (PG) e positivo muito grande (PMG).

A Figura 7.14 apresenta as funções de pertença que caracterizam a saída “*Grau_Confia*”. Foram igualmente definidas 5 funções de pertença mas do tipo *singleton* (que podem ser consideradas um caso especial das funções de pertença triangulares): zero (ZE), positivo pequeno (PP), positivo médio (PM), positivo grande (PG) e positivo muito grande (PMG). Com o método de desfuzificação utilizado (CoM), o resultado era mesmo caso fossem definidas funções de pertença triangulares, pois este método tem apenas em consideração o valor máximo das funções de pertença activadas pelo processo de inferência.

No terceiro passo são definidas as regras e a sua credibilidade. A Tabela 7-2 apresenta o conjunto de regras definido, tendo sido atribuído a mesma credibilidade (*DoS-Degree of Support*) a todas as regras.

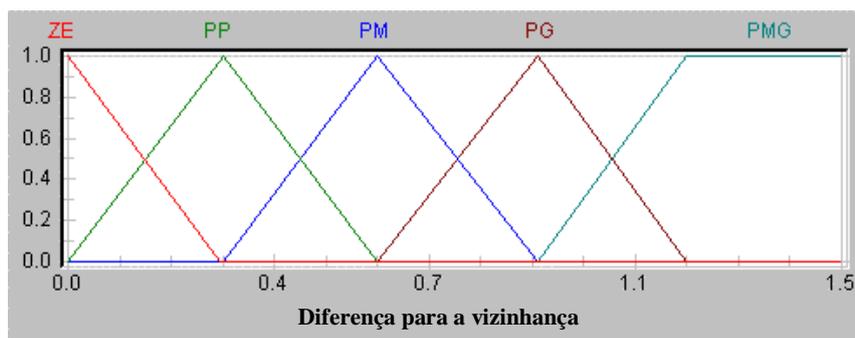


Figura 7.13 Funções de pertinência da entrada “dif_viz”.

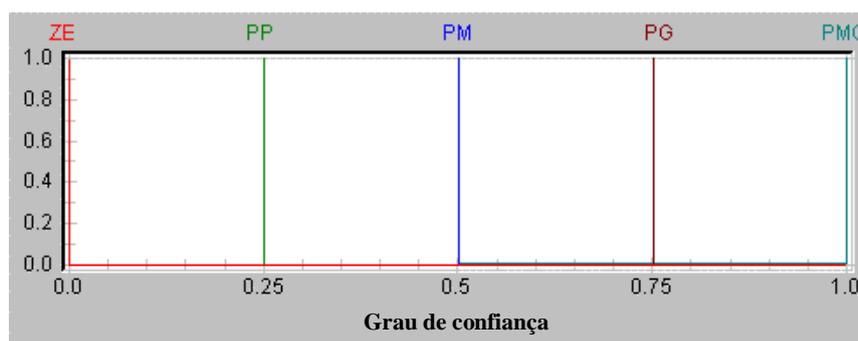


Figura 7.14 Funções de pertinência da saída “Grau_Confia”.

IF	THEN	
dif_viz	DoS	Grau_Confia
ZE	1.00	PMG
PP	1.00	PG
PM	1.00	PM
PG	1.00	PP
PMG	1.00	ZE

Tabela 7-2 Conjunto de regras.

O quarto e último passo consiste no afinamento do “decisor” *fuzzy*, tendo sido utilizado unicamente o *debugging* interativo neste processo em virtude da simplicidade do problema. É de salientar que em casos mais complexos é usual a utilização de técnicas *neuro-fuzzy* ou algoritmos genéticos, estando as primeiras incluídas no pacote *fuzzyTECH*.

Com o “decisor” *fuzzy* implementado, a valores de entrada pequenos ($dif_viz \leq 0.7$) corresponde um grau de confiança/crença grande ($Grau_Confia \geq 0.42$); para valores de entrada grandes ($dif_viz > 0.7$) corresponde um grau de confiança/crença baixo ($Grau_Confia < 0.42$) sendo igual a 0 para todos os valores cuja diferença ultrapasse o valor 1.2.

O resultado final, que corresponde à distância atribuída ao ponto que apresenta médias divergentes (estimativa da distância), é definido de acordo com expressão 7.3,

$$D_{Pos(1234)} = \frac{(C_{-1} + C_{-2} + C_{-3} + C_{-4})}{C_{-T}} \times M_{-X} + \frac{(C_{-5} + C_{-6} + C_{-7} + C_{-8})}{C_{-T}} \times M_{-Y} \tag{7.3}$$

onde,

$D_{Pos(1\ 2\ 3\ 4)}$ – Estimativa da distância para a posição 1, 2, 3 ou 4,

C_1 a C_8 – Graus de confiança parciais resultante do processo de desfuzificação quando são aplicadas as entradas de 1 a 8 ao “decisor” *Fuzzy* (ver Exemplo 7.1),

$$C_T = C_1 + C_2 + C_3 + C_4 + C_5 + C_6 + C_7 + C_8,$$

M_X – Média do vector de medida adquirido pelo ultrassom X quando aponta para a posição 1, 2, 3 ou 4,

M_Y – Média do vector de medida adquirido pelo ultrassom Y quando aponta para a mesma posição que o ultrassom X .

Exemplo 7.1

Considere-se que a posição 1 (corresponde à posição original apontada pelo sensor 1 – sem a rotação de 90°) apresenta os seguintes valores para a média dos dois sensores que foram posicionados a apontar para esse ponto:

$M1_X = 42$ cm (Média do vector de medida obtido pelo sensor X quando apontava para a posição 1)

$M1_Y = 44$ cm (Média do vector de medida obtido pelo sensor Y quando apontava para a posição 1)

Resolução : Para extrair o valor que serve de entrada são utilizadas as 2 médias das duas posições localizadas lateralmente à posição que apresentou a divergência (neste caso as da posição 2 e da posição 4), nunca sendo utilizada a posição na diagonal. Neste caso as entradas seriam as seguintes:

$$\begin{array}{ll} \text{Ent1} = | M1_X - M2_X | & \text{Ent5} = | M1_Y - M2_X | \\ \text{Ent2} = | M1_X - M2_Y | & \text{Ent6} = | M1_Y - M2_Y | \\ \text{Ent3} = | M1_X - M4_X | & \text{Ent7} = | M1_Y - M4_X | \\ \text{Ent4} = | M1_X - M4_Y | & \text{Ent8} = | M1_Y - M4_Y | \end{array}$$

Após o processo de inferência e a aplicação do método de desfuzificação obtém-se 8 saídas correspondendo cada uma delas a uma entrada:

$$\begin{array}{ll} C_1 = \text{Resultado do CoM}_1 & C_5 = \text{Resultado do CoM}_5 \\ C_2 = \text{Resultado do CoM}_2 & C_6 = \text{Resultado do CoM}_6 \\ C_3 = \text{Resultado do CoM}_3 & C_7 = \text{Resultado do CoM}_7 \\ C_4 = \text{Resultado do CoM}_4 & C_8 = \text{Resultado do CoM}_8 \end{array}$$

A estimativa da medida para a posição 1 é então calculada através da expressão 7.3.

Filtro de Kalman

O filtro de *Kalman* é utilizado para estimar a distância a um determinado ponto pertencente à superfície do objecto, permitindo deste modo uma obtenção mais rigorosa do relevo.

Considera-se que os ultrassons X e Y fornecem informação redundante relativamente ao relevo do objecto. Então, o estado a ser estimado é o relevo do objecto que pode ser considerado inalterável ao longo do tempo. Isto é, $x_k = x$ para todo o k . A medida do relevo z_x e z_y efectuadas pelos ultrassons $X (U_X)$ e $Y (U_Y)$ respectivamente podem ser modeladas como

$$z_x = x + v_x \quad \text{e} \quad z_y = x + v_y$$

onde v_x e v_y são variáveis aleatórias Gaussianas de média zero com variâncias σ_x^2 e σ_y^2 .

Assumindo que as medidas a partir do U_x estão inicialmente disponíveis, então

$$\hat{x}_0 = z_x \quad \text{e} \quad P_0 = \sigma_x^2$$

podem ser consideradas informação à priori acerca do estado de x . Quando as medidas a partir de U_y ficam disponíveis, a estimativa para x é dada por

$$\begin{aligned} \hat{x} &= \hat{x}_0 + K \left[z_y - H \hat{x}_0 \right] \\ &= \hat{x}_0 + P_0 H^T (H P_0 H^T + R)^{-1} \left[z_y - H \hat{x}_0 \right] \\ &= z_x + \sigma_x^2 (\sigma_x^2 + \sigma_y^2)^{-1} [z_y - z_x] \\ &= \frac{\sigma_y^2}{\sigma_x^2 + \sigma_y^2} \times z_x + \frac{\sigma_x^2}{\sigma_x^2 + \sigma_y^2} \times z_y \end{aligned} \tag{7.4}$$

onde $R = \sigma_y^2$.

As variâncias σ_x^2 e σ_y^2 na estimação de \hat{x} podem ser interpretados como fornecendo uma média pesada das medidas z_x e z_y , onde a medida com menor variância tem maior peso no processo de estimação.

No caso desta aplicação as medidas z_x e z_y são iguais à média do vector de medidas recolhidas pelos ultrassons X e Y . Os valores de σ_x^2 e σ_y^2 correspondem à variância das respectivas amostras.

Média das médias

A média das médias é aplicada quando nenhuma das condições de aplicação das duas técnicas anteriormente referidas não satisfeita.

O valor adoptado para a respectiva posição é então definido pela seguinte expressão,

$$Pos_{(1234)} = \frac{M_X + M_Y}{2} \tag{7.5}$$

onde,

M_X = Média do vector de medida adquirido pelo ultrassom X .

M_Y = Média do vector de medida adquirido pelo ultrassom Y .

Método heurístico

Como técnica de integração utilizou-se um método heurístico para garantir uma aquisição mais precisa do relevo nos limites exteriores do objecto. É unicamente aplicada no caso em que o robô se encontra nesses limites, garantindo que valores sem sentido adquiridos pelos ultrassons em consequência de apontarem para essas posições críticas influenciem as características do relevo a obter.

A implementação deste método baseia-se num conjunto de regras do tipo

se (aponta para limite superior) então
Valor da Posição 1 = Valor da Posição 2
Valor da Posição 4 = Valor da Posição 3
senão
...

Com estas regras, nos limites exteriores do objecto, só os valores adquiridos por alguns dos ultrassons são considerados. No caso acima só os valores dos ultrassons 2 e 3 são considerados, sendo os outros ignorados.

7.6 Sumário

Neste capítulo foi descrito o modelo dos ultrassons utilizado e as técnicas de fusão/integração sensorial de dados implementadas.

Na definição do modelo para os ultrassons foram introduziu-se os conceitos de pontos de profundidade constante (PPC) e de regiões de profundidade constante (RPC). Os PPC permitem fazer a distinção entre os retornos fortes e fracos, sendo unicamente processados os fortes. As RPC permitem atribuir a uma área o mesmo valor de profundidade.

As técnicas de fusão sensorial de dados implementadas foram três: “decisor” *fuzzy*, filtro de *Kalman* e a média. Só uma destas técnicas pode ser utilizada na estimação da distância a um ponto. Contudo técnicas diferentes podem ser utilizadas em pontos diferentes.

Como técnica de integração foi utilizado o método heurístico. Este é somente utilizado nas situações em que os ultrassons apontam para os limites do objecto.

Capítulo 8

8. Resultados experimentais

8.1 Introdução

O objectivo do sistema descrito nos capítulos anteriores prende-se com o levantamento e representação do relevo da superfície visível de objectos localizados sobre uma mesa posicionada em torno do robô, dentro da sua área de trabalho.

Foram efectuados um conjunto de testes experimentais para avaliar o desempenho das diferentes partes que constituem o sistema implementado, nomeadamente:

1. As opções relativamente às configurações de *hardware* (Cap. 4);
2. A rede CAN utilizada na comunicação de dados (Cap.5);
3. As técnicas de processamento de imagem para obtenção dos pontos dominantes do objecto (Cap.6);
4. modelo adoptado para os ultrassons (Cap. 7);
5. As técnicas de fusão/integração sensorial aplicadas (Cap.7);

Contudo, os resultados apresentados reflectem somente o desempenho dos dois últimos pontos (4 e 5), sendo os outros (pontos 1, 2 e 3) avaliados de uma forma indirecta, isto é, sem um adequado funcionamento seria impossível a obtenção de qualquer resultado.

Para efectuar a avaliação foram utilizados cinco modelos em madeira com a forma quadrada/rectangular apresentando regiões de diferentes profundidades na sua superfície. Esta apresenta uma certa rugosidade, o que aumenta a probabilidade de detecção do eco resultante do feixe emitido pelos ultrassons. No caso das superfícies lisas (especulares) a probabilidade do feixe reflectido nunca encontrar o receptor é muito elevada como já anteriormente foi referido. Os modelos obedecem ainda a algumas restrições impostas quer pela limitação dos ultrassons quer pela disposição destes no punho do robô:

- A diferença de profundidade entre duas regiões da superfícies nunca pode ser inferior a 1.8 cm devido à dificuldade apresentada pelos ultrassons em distinguir e localizar os pontos de transição de profundidade com um valor inferior a este número.

- Os ângulos formados pelos diferentes planos que constituem o relevo são sempre iguais a 90° devido à disposição dos ultrassons formarem um quadrado.

Os modelos são colocados sobre a mesa de trabalho e a superfície a adquirir deverá fazer um ângulo inferior a 8° com o plano de deslocamento do robô.

8.2 Testes realizados

Os resultados dos testes experimentais que se incluem neste trabalho apresentam um grau crescente de dificuldade na aquisição, uma vez que o seu relevo se caracteriza por um número cada vez maior de regiões com diferentes profundidades e cantos. Foram realizados dois testes com cada modelo. O primeiro teste foi efectuado com um ângulo de inclinação aproximadamente igual a 0° entre o plano da superfície do objecto e o plano de deslocamento do robô. O segundo teste foi efectuado com um determinado ângulo de inclinação (menor que 8°) entre os dois planos anteriormente referidos. A realização do segundo teste teve como finalidade avaliar se a qualidade do resultado obtido era semelhante ao da situação mais favorável (primeiro teste), especialmente na detecção dos pontos onde ocorrem as transições de profundidade.

Os resultados obtidos são apresentados com e sem filtragem. No primeiro caso os valores das RPCs foram filtrados de forma a representar com a mesma cor (correspondendo a distâncias) todas as RPCs cuja diferença de profundidades entre elas fosse inferior a 1.5 cm. No segundo caso as RPCs são representadas com o valor realmente obtido para a profundidade. A apresentação dos resultados sem filtragem tem como finalidade avaliar o comportamento real do sistema no processo de aquisição da superfície.

As Figura 8.1(a), Figura 8.7(a), Figura 8.13(a), Figura 8.19(a) e Figura 8.25(a) apresentam os modelos utilizados nos testes com as respectivas dimensões.

As Figura 8.1(b), Figura 8.7(b), Figura 8.13(b), Figura 8.19(b) e Figura 8.25(b) apresentam o mapa de superfície a 2D. Todas estas figuras foram geradas pelo *software* de processamento de imagem desenvolvido. Podem observar-se as dimensões do objecto (altura x largura) determinadas por aplicação de fusão sensorial entre os dados da câmara e os dos ultrassons, e as distâncias das RPCs relativamente ao plano em que o robô se deslocou. É ainda possível distinguir claramente as RPCs adquiridas. As Figura 8.4, Figura 8.10, Figura 8.16, Figura 8.22 e Figura 8.28 apresentam a mesma informação que as anteriores, mas representam a superfície adquirida quando existe um ângulo de inclinação entre o plano de deslocamento do robô e o plano de superfície do objecto. Nestes casos, as superfícies planas apresentam RPCs com profundidades diferentes.

As Figura 8.2(a), Figura 8.5(a), Figura 8.8(a), Figura 8.11(a), Figura 8.14(a), Figura 8.17(a), Figura 8.20(a), Figura 8.23(a), Figura 8.26(a) e Figura 8.29(a) constituem as representações a 3D do relevo obtido, tendo sido geradas pelo *Matlab* a partir de informação previamente guardada em ficheiro. Os valores das RPCs foram filtrados de forma a representar com a mesma cor (correspondendo a distâncias) todas as RPCs cuja diferença de profundidades entre elas fosse inferior a 1.5 cm. As Figura 8.2(b), Figura 8.5(b), Figura 8.8(b), Figura 8.11(b), Figura 8.14(b), Figura 8.17(b), Figura 8.20(b), Figura 8.23(b), Figura 8.26(b) e Figura 8.29(b) apresentam o mesmo tipo de informação a 2D.

As Figura 8.3(a), Figura 8.6(a), Figura 8.9(a), Figura 8.12(a), Figura 8.15(a), Figura 8.18(a), Figura 8.21(a), Figura 8.24(a), Figura 8.27(a) e Figura 8.30(a) apresenta-se o mesmo tipo de informação do caso anterior mas sem filtragem.

É ainda de referir que os resultados apresentados são apenas um exemplo podendo os mesmos apresentarem pequenas diferenças, essencialmente na detecção dos pontos onde acontecem as transições de profundidade (podendo por exemplo ser detectados com ± 0.5 cm ou mesmo ± 1.0 cm).

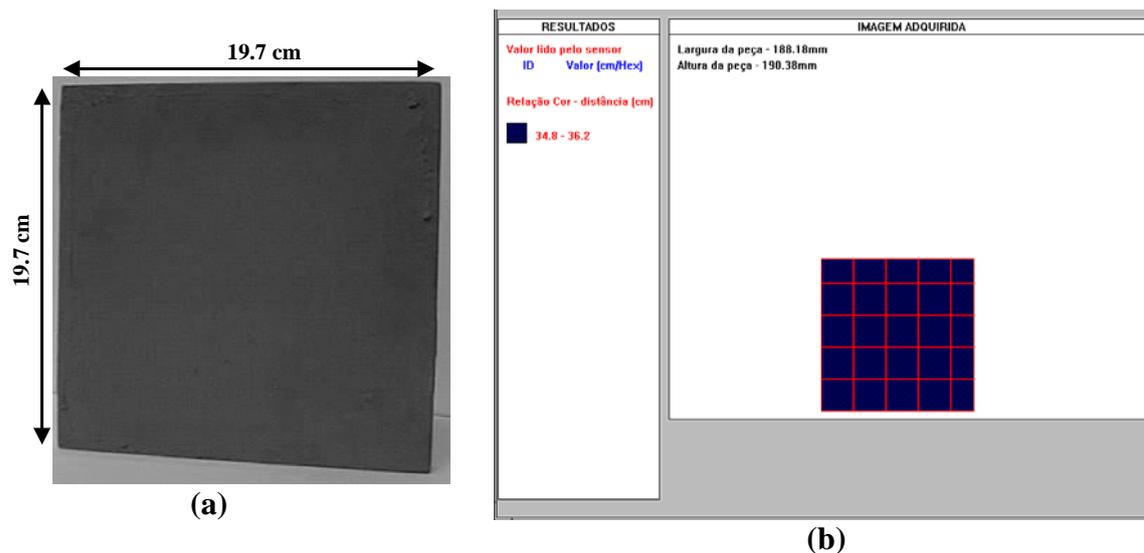


Figura 8.1 Modelo 1 – superfície plana. (a) Dimensões do modelo; (b) Representação do mapa de superfície a 2D.

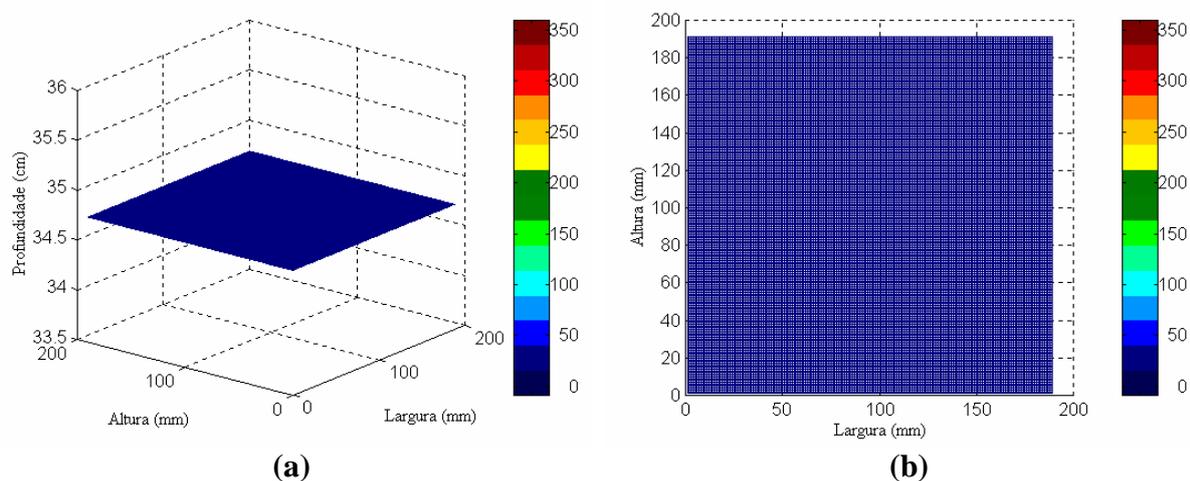


Figura 8.2 Mapa de superfície com filtragem dos valores obtidos para as RPCs. (a) 3D; (b) 2D.

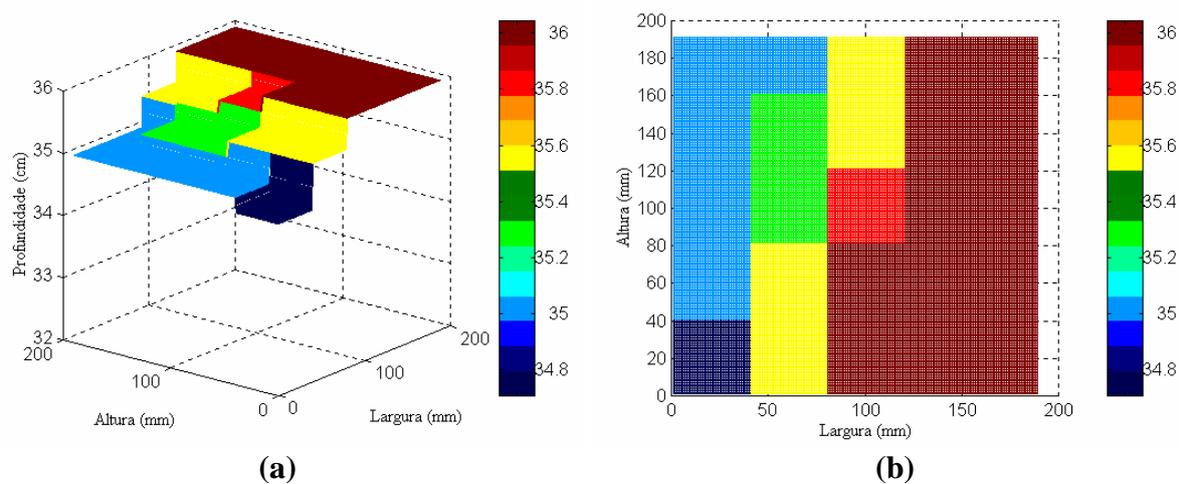


Figura 8.3 Mapa de superfície sem filtragem dos valores obtidos para as RPCs. (a) 3D; (b) 2D.

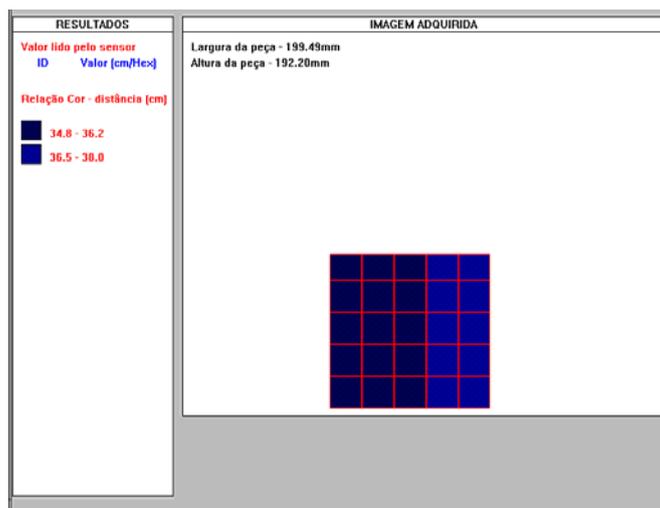


Figura 8.4 Mapa de superfície a 2D do modelo 1. Aquisição efectuada com inclinação relativamente ao plano de deslocamento do robô.

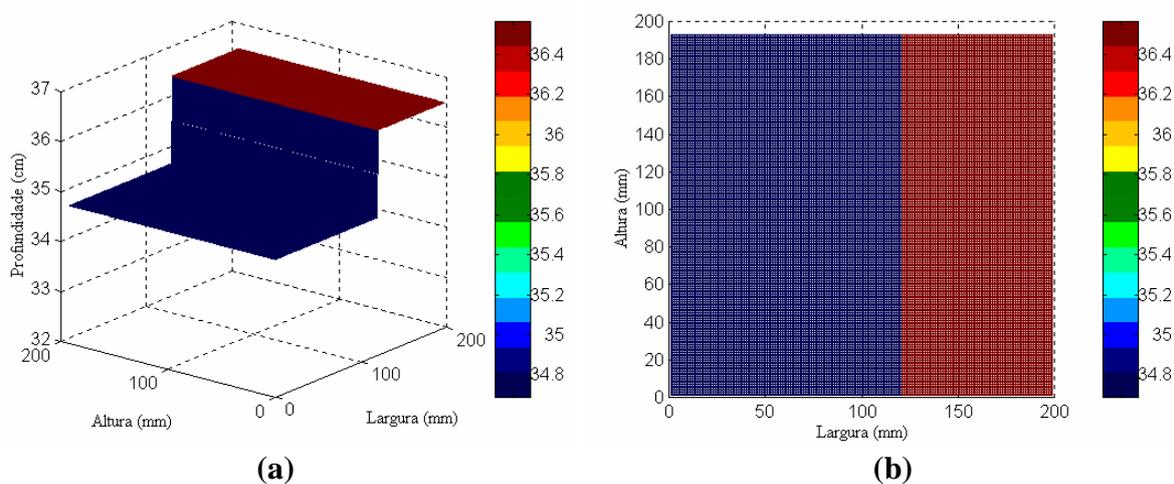


Figura 8.5 Mapa de superfície com filtragem dos valores obtidos para as RPCs. (a) 3D; (b) 2D.

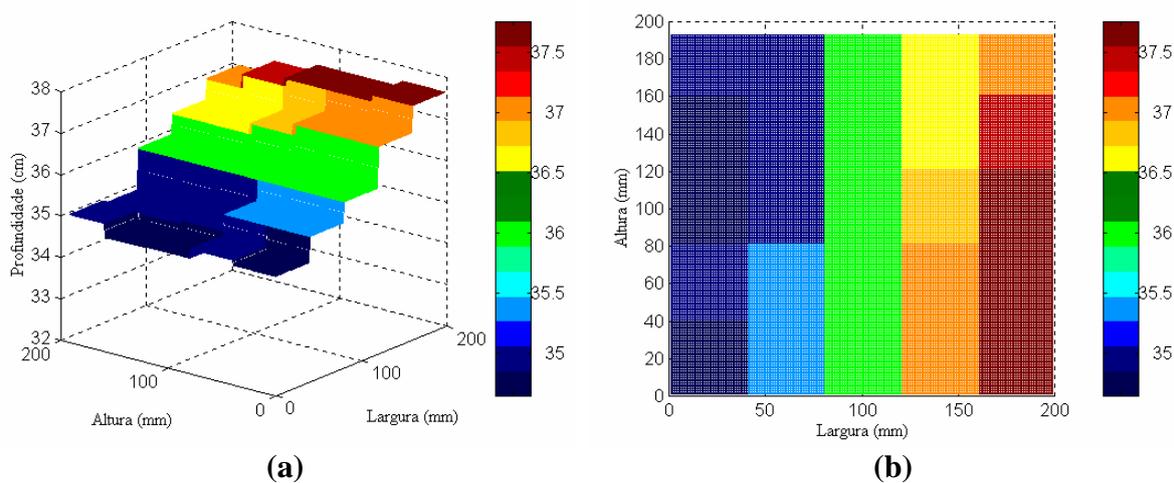


Figura 8.6 Mapa de superfície sem filtragem dos valores obtidos para as RPCs. (a) 3D; (b) 2D.

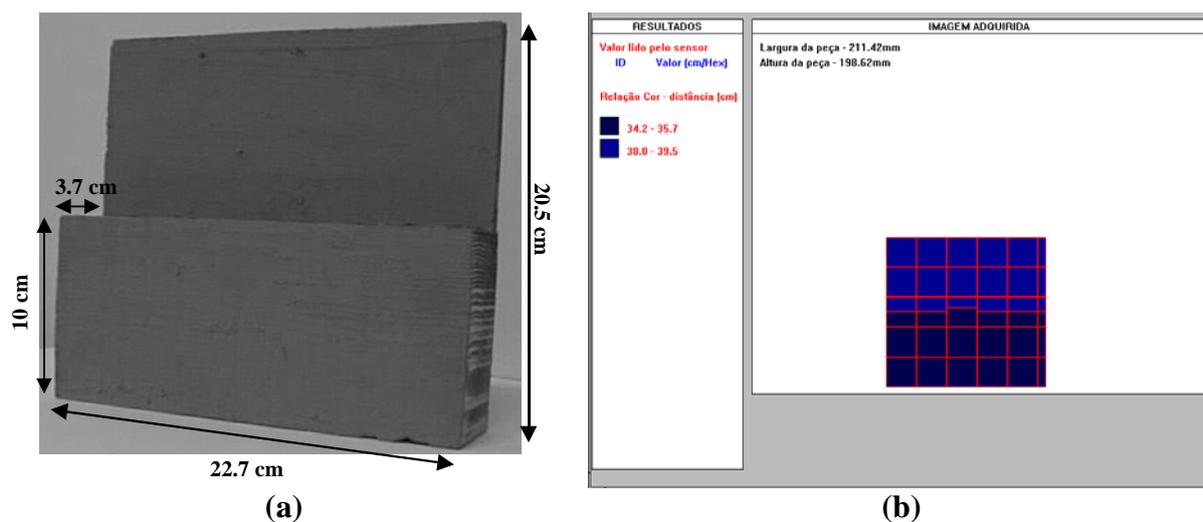


Figura 8.7 Modelo 2 – 2 profundidades diferentes. (a) Dimensões do modelo; (b) Representação do mapa de superfície a 2D.

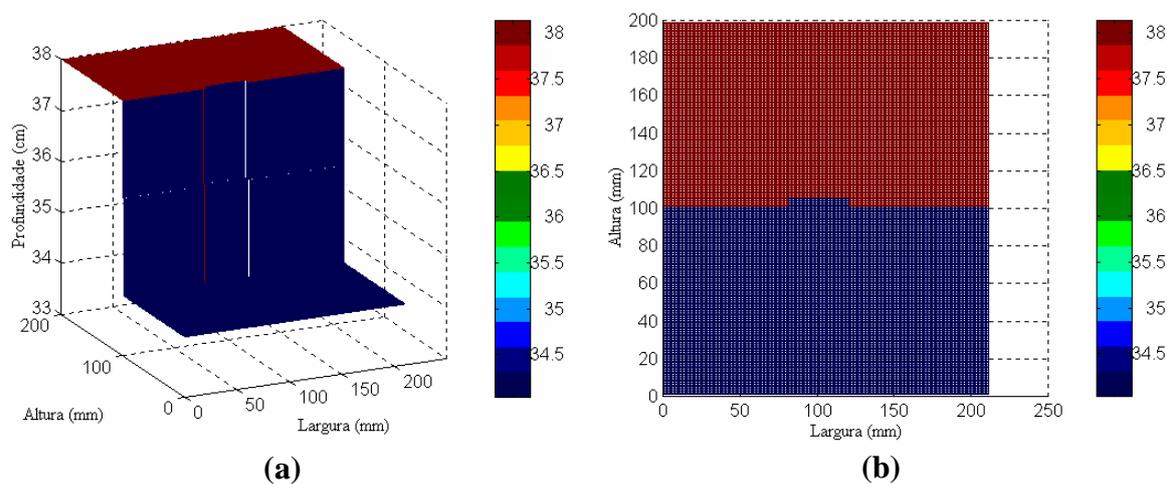


Figura 8.8 Mapa de superfície com filtragem dos valores obtidos para as RPCs. (a) 3D; (b) 2D.

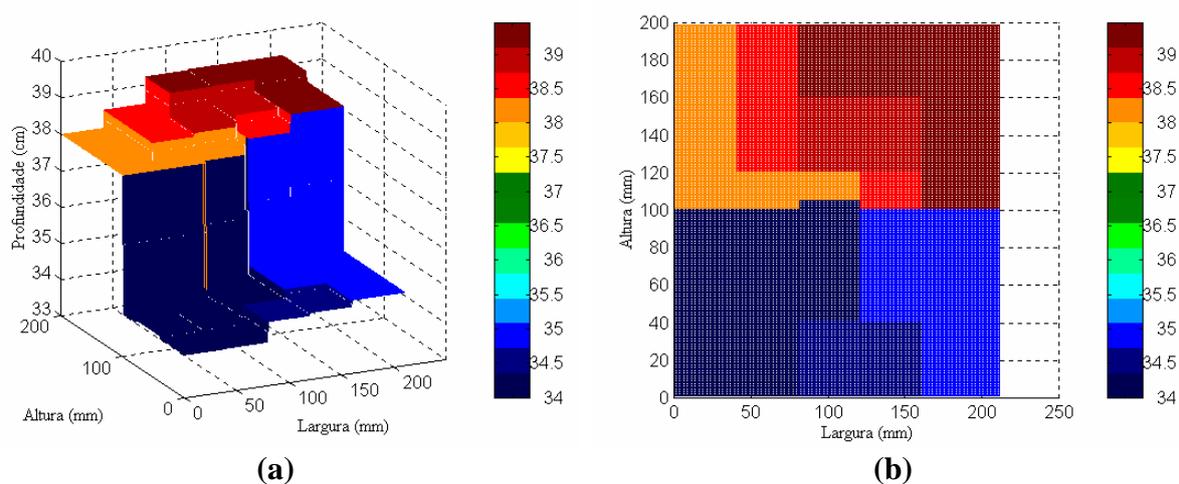


Figura 8.9 Mapa de superfície sem filtragem dos valores obtidos para as RPCs. (a) 3D; (b) 2D.

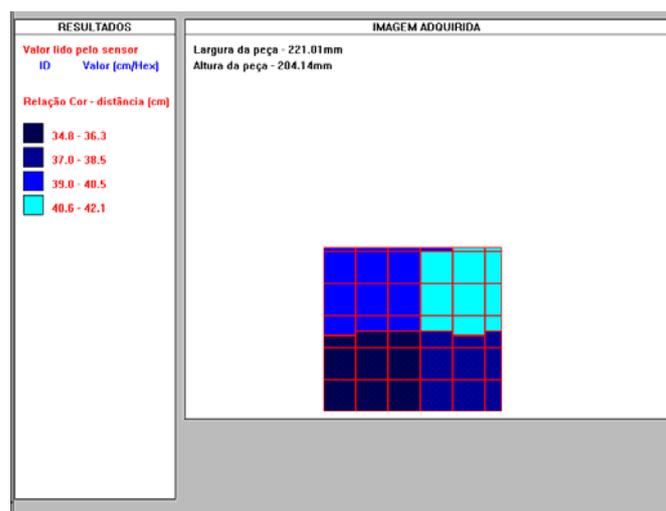


Figura 8.10 Mapa de superfície a 2D do modelo 2. Aquisição efectuada com inclinação relativamente ao plano de deslocamento do robô.

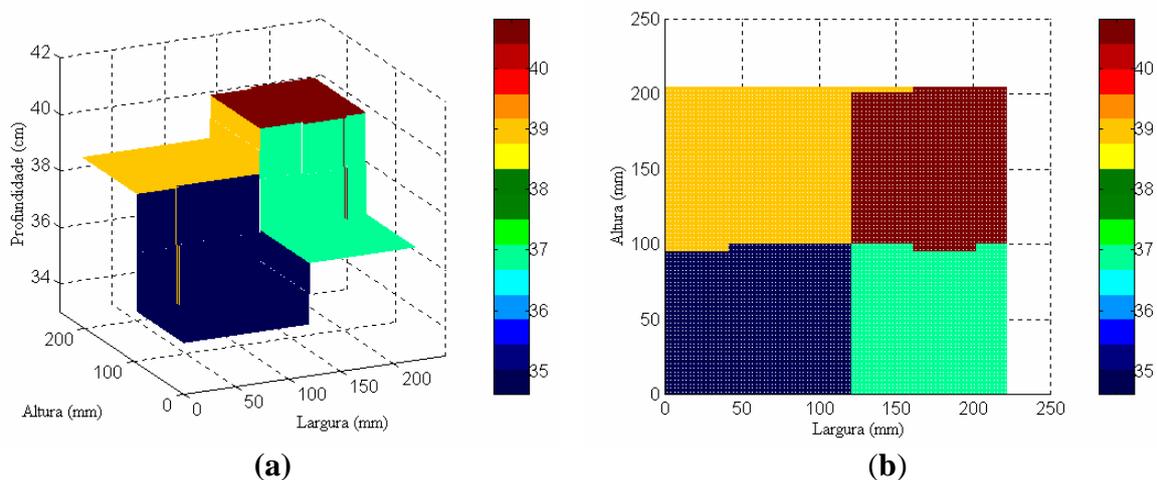


Figura 8.11 Mapa de superfície com filtragem dos valores obtidos para as RPCs. (a) 3D; (b) 2D.

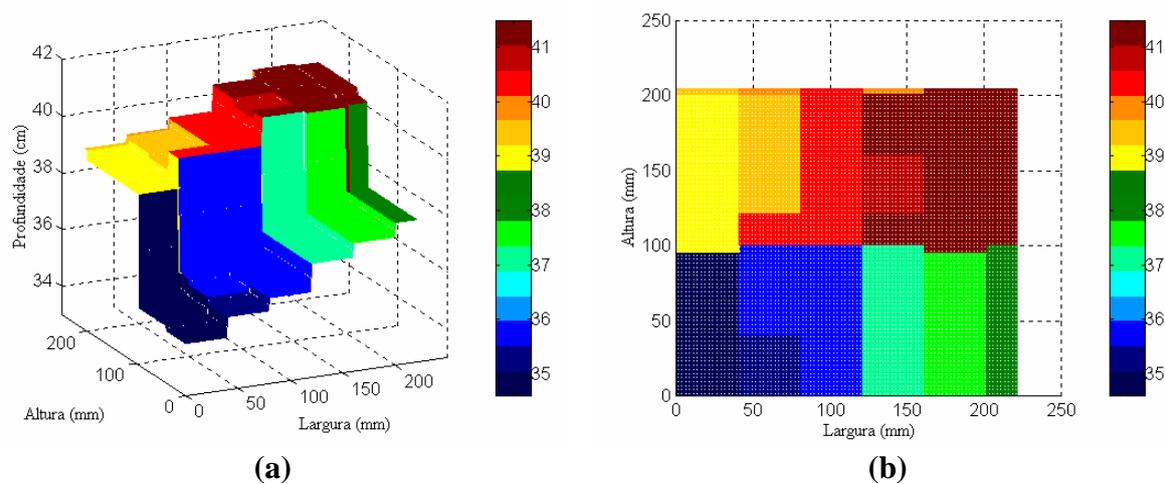


Figura 8.12 Mapa de superfície sem filtragem dos valores obtidos para as RPCs. (a) 3D; (b) 2D.

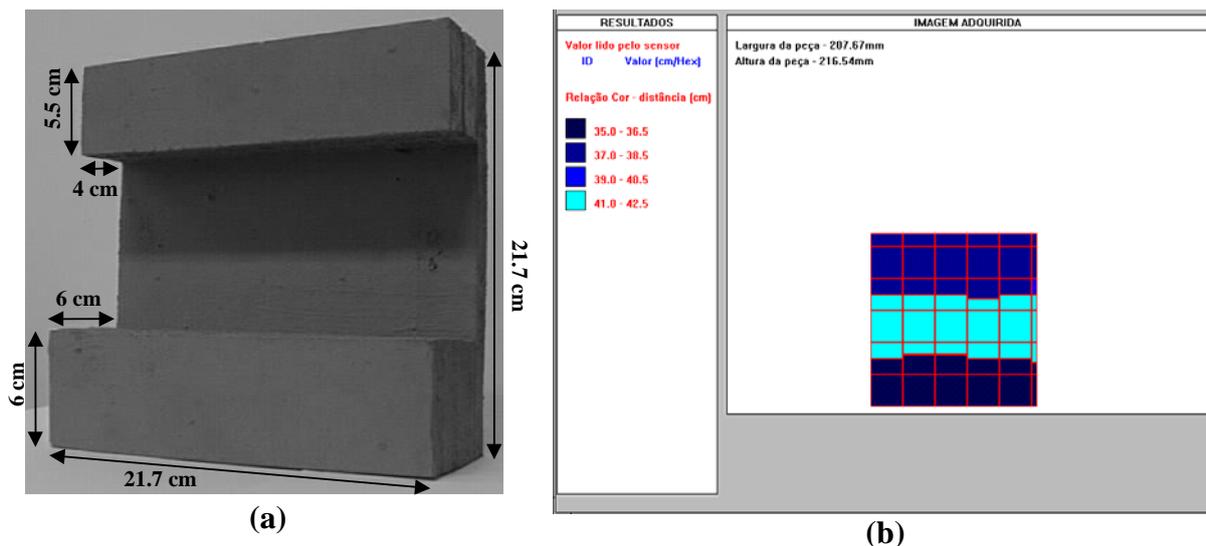


Figura 8.13 Modelo 3 – 3 profundidades diferentes. (a) Dimensões do modelo; (b) Representação do mapa de superfície a 2D.

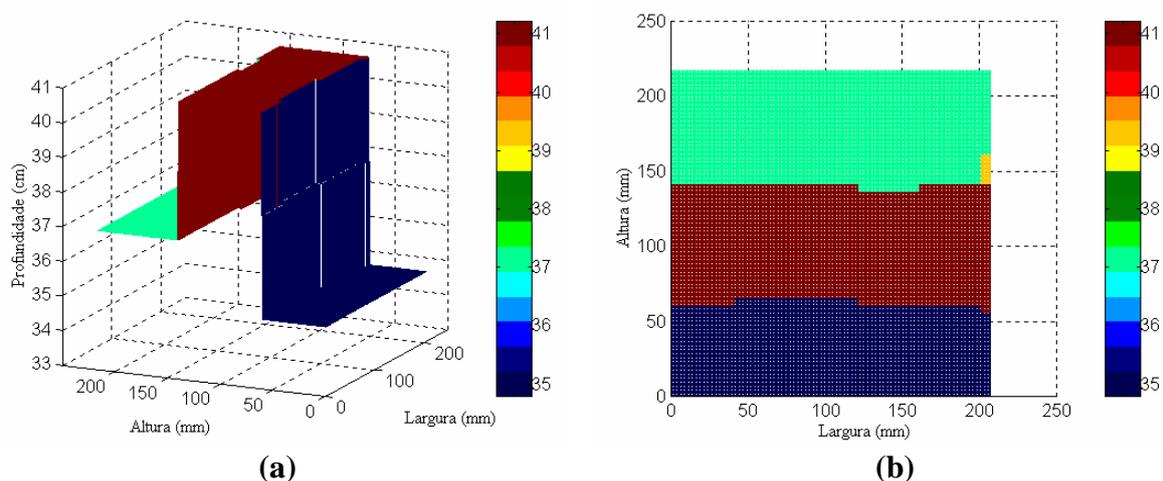


Figura 8.14 Mapa de superfície com filtragem dos valores obtidos para as RPCs. (a) 3D; (b) 2D.

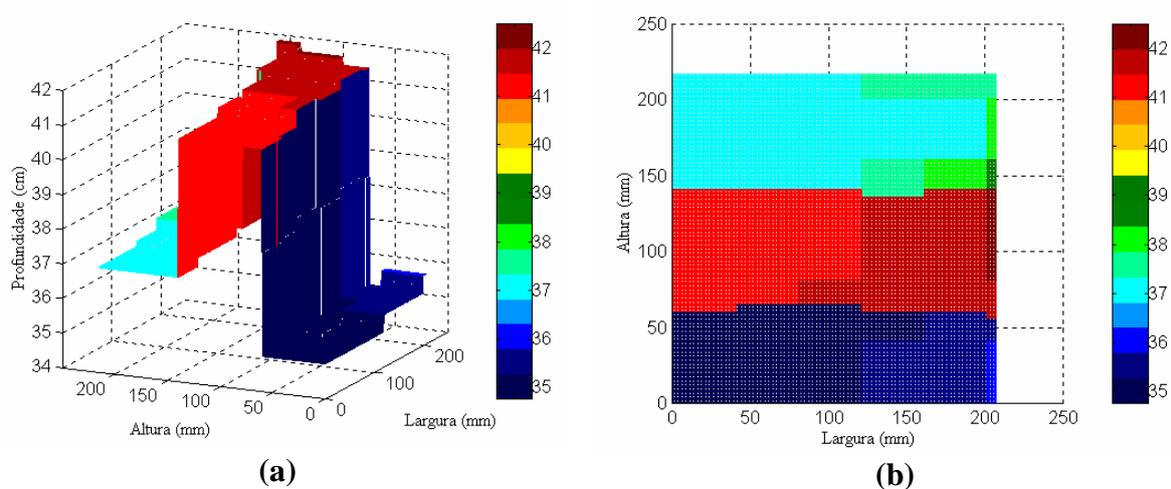


Figura 8.15 Mapa de superfície sem filtragem dos valores obtidos para as RPCs. (a) 3D; (b) 2D.

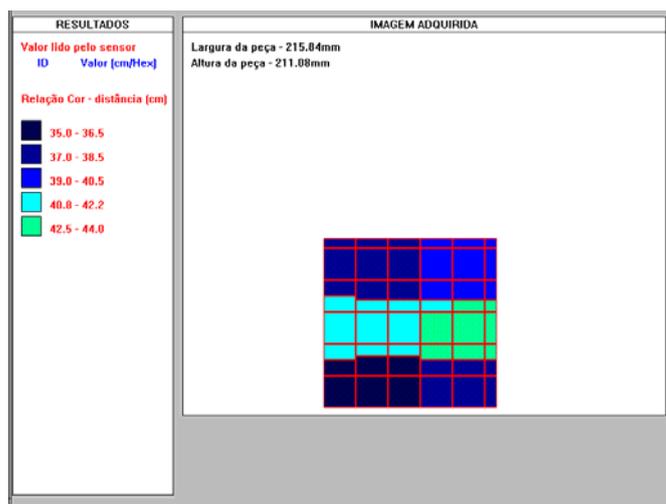


Figura 8.16 Mapa de superfície a 2D do modelo 3. Aquisição efectuada com inclinação relativamente ao plano de deslocamento do robô.

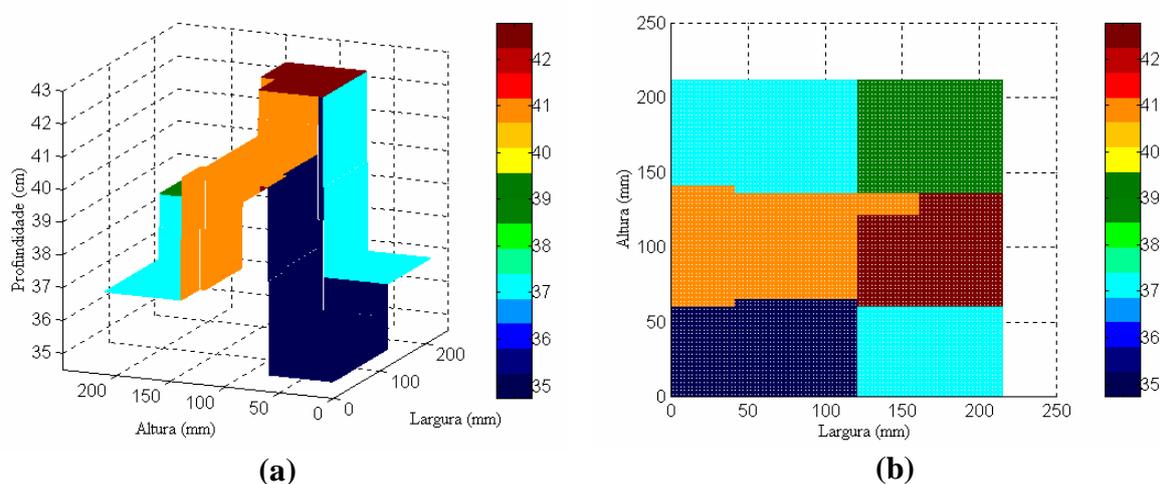


Figura 8.17 Mapa de superfície com filtragem dos valores obtidos para as RPCs. (a) 3D; (b) 2D.

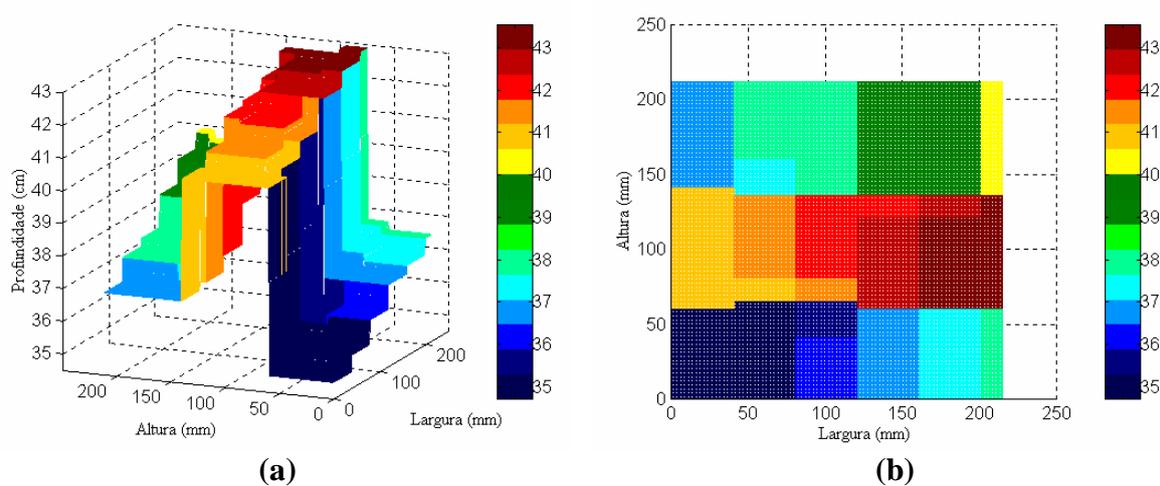


Figura 8.18 Mapa de superfície sem filtragem dos valores obtidos para as RPCs. (a) 3D; (b) 2D.

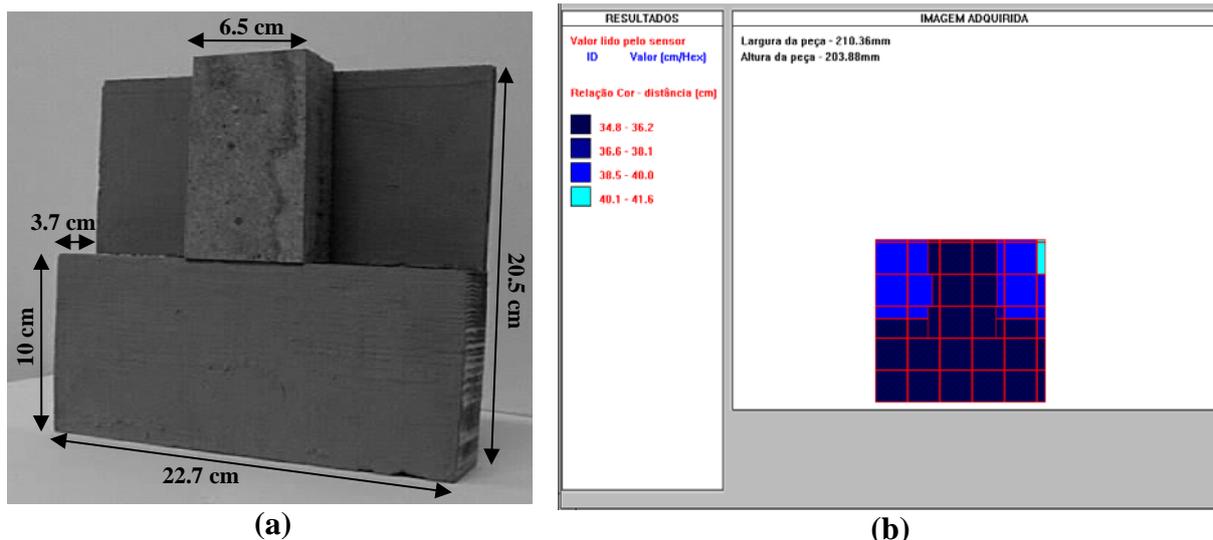


Figura 8.19 Modelo 4 – 2 profundidades diferentes e cantos. (a) Dimensões do modelo; (b) Representação do mapa de superfície a 2D.

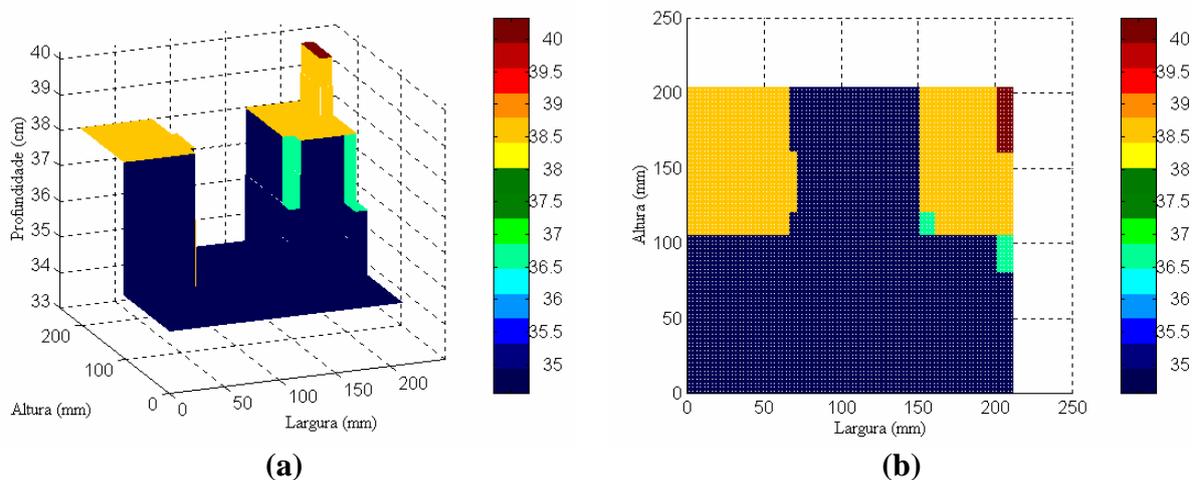


Figura 8.20 Mapa de superfície com filtragem dos valores obtidos para as RPCs. (a) 3D; (b) 2D.

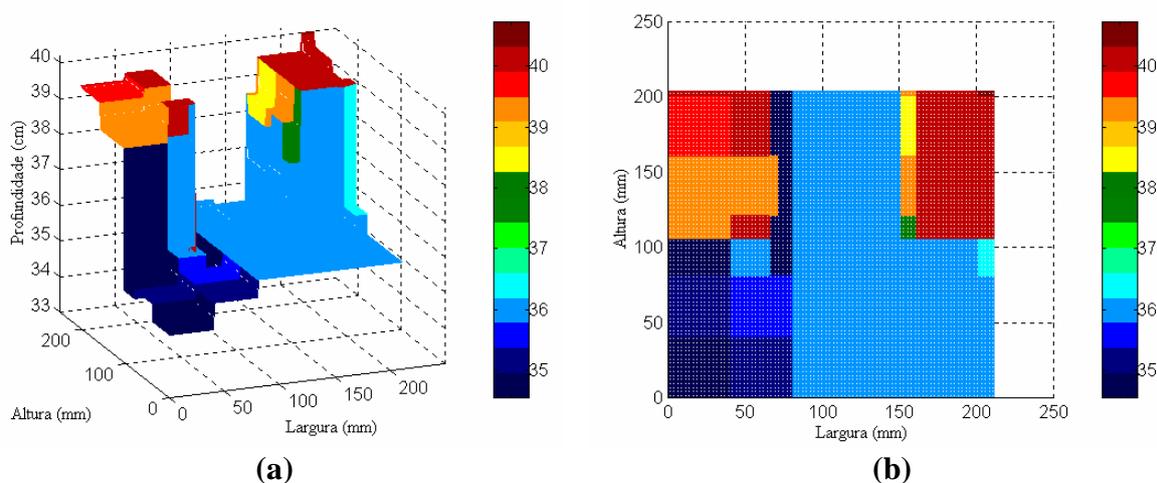


Figura 8.21 Mapa de superfície sem filtragem dos valores obtidos para as RPCs. (a) 3D; (b) 2D.

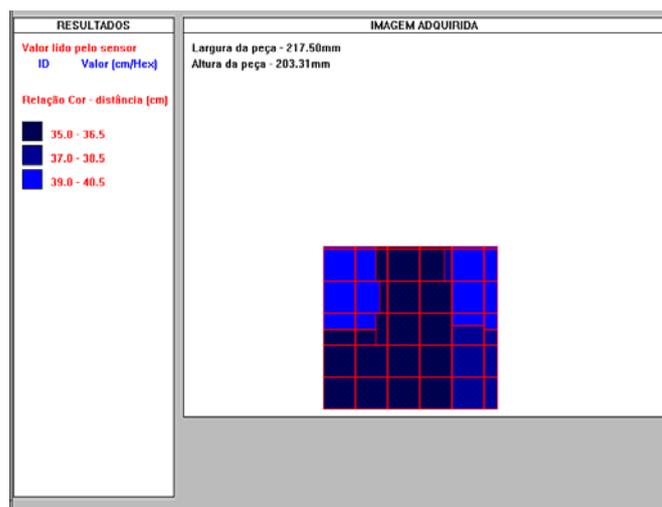


Figura 8.22 Mapa de superfície a 2D do modelo 4. Aquisição efectuada com inclinação relativamente ao plano de deslocamento do robô.

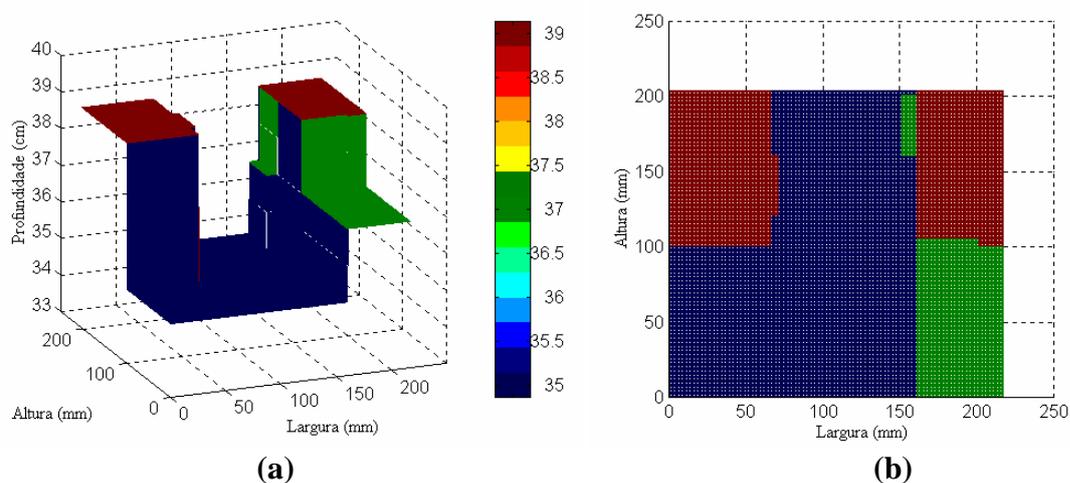


Figura 8.23 Mapa de superfície com filtragem dos valores obtidos para as RPCs. (a) 3D; (b) 2D.

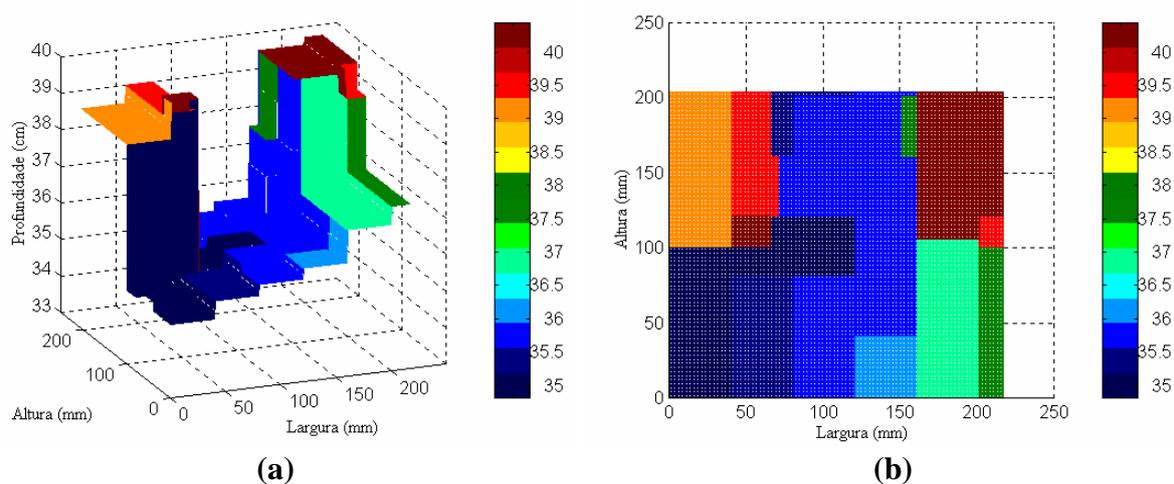


Figura 8.24 Mapa de superfície sem filtragem dos valores obtidos para as RPCs. (a) 3D; (b) 2D.

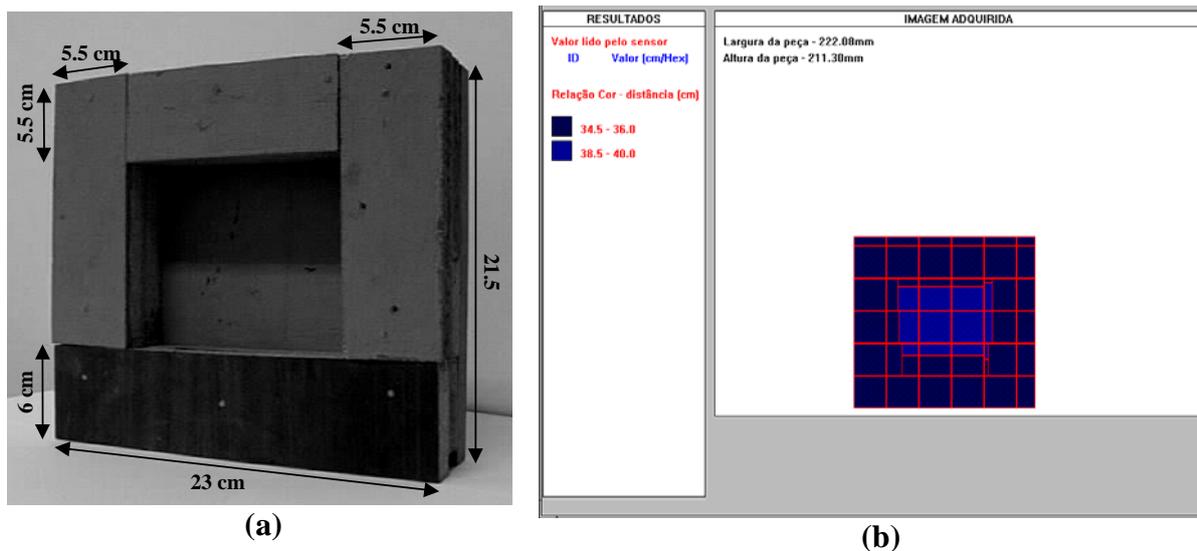


Figura 8.25 Modelo 5 – 2 profundidades diferentes e cantos. (a) Dimensões do modelo; (b) Representação do mapa de superfície a 2D.

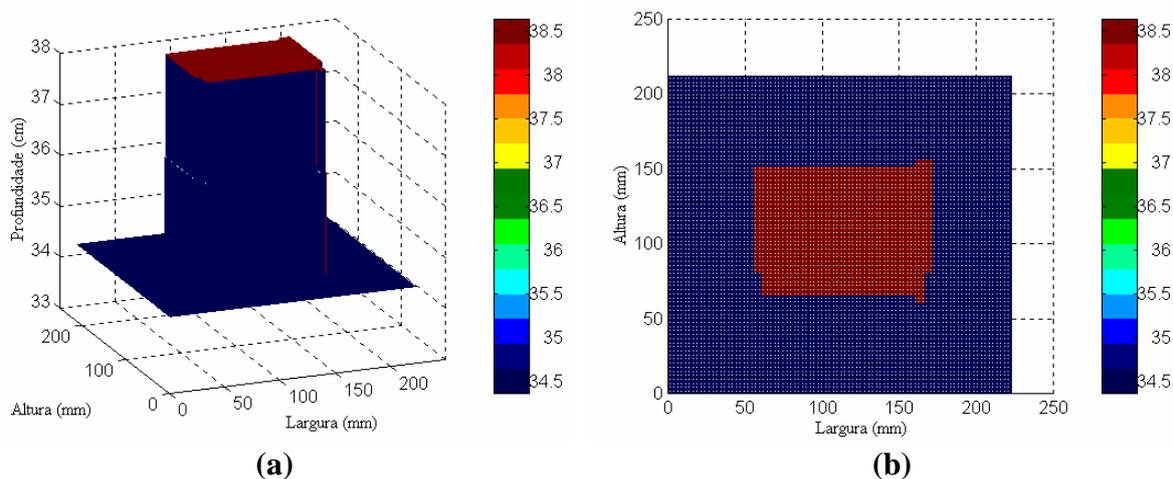


Figura 8.26 Mapa de superfície com filtragem dos valores obtidos para as RPCs. (a) 3D; (b) 2D.

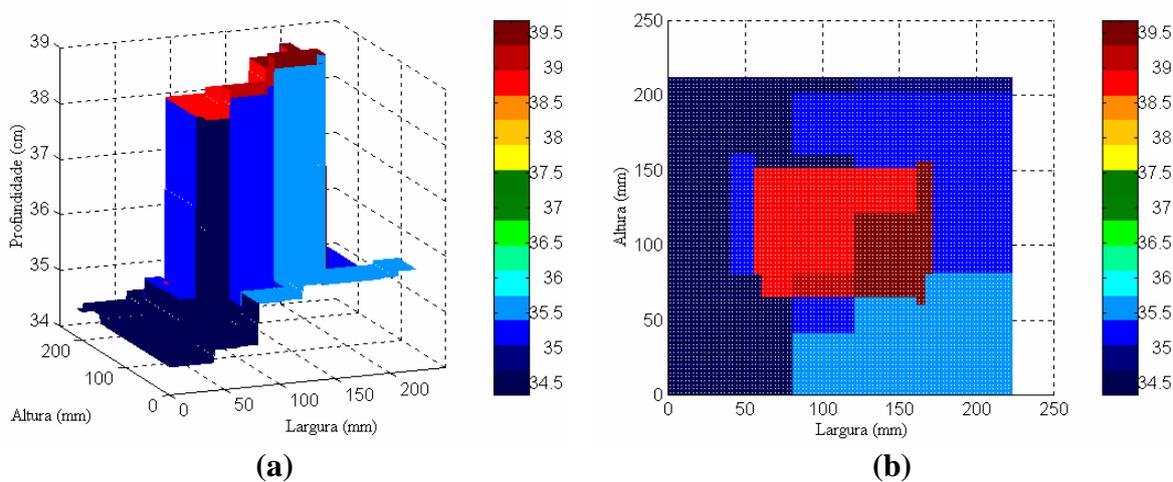


Figura 8.27 Mapa de superfície sem filtragem dos valores obtidos para as RPCs. (a) 3D; (b) 2D.

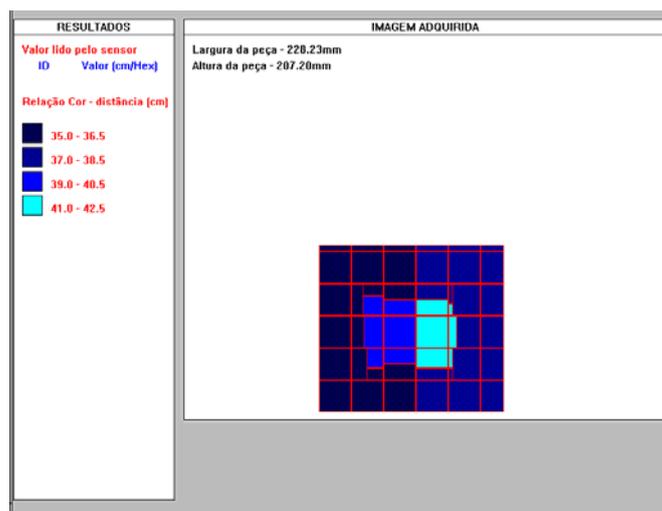


Figura 8.28 Mapa de superfície a 2D do modelo 5. Aquisição efectuada com inclinação relativamente ao plano de deslocamento do robô.

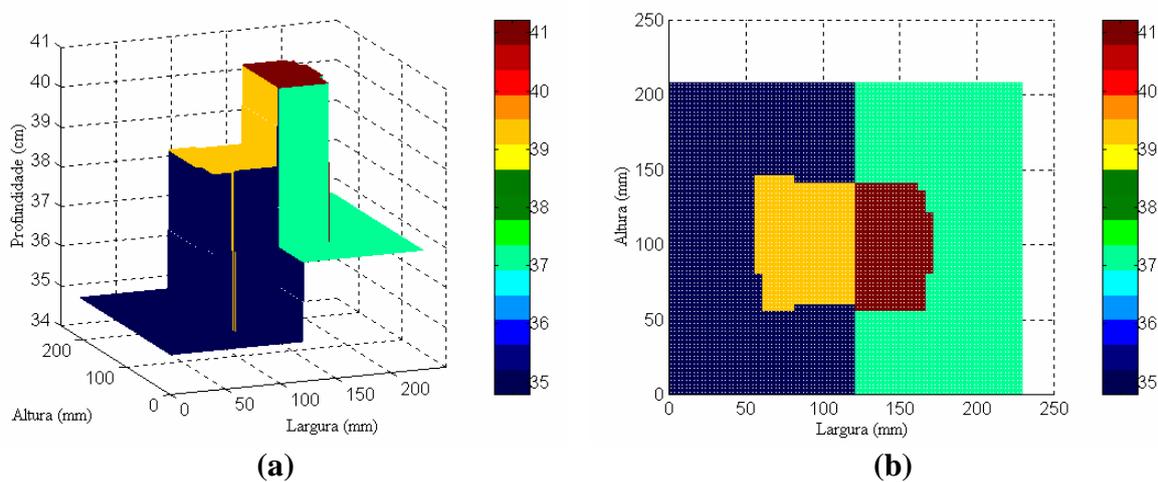


Figura 8.29 Mapa de superfície com filtragem dos valores obtidos para as RPCs. (a) 3D; (b) 2D.

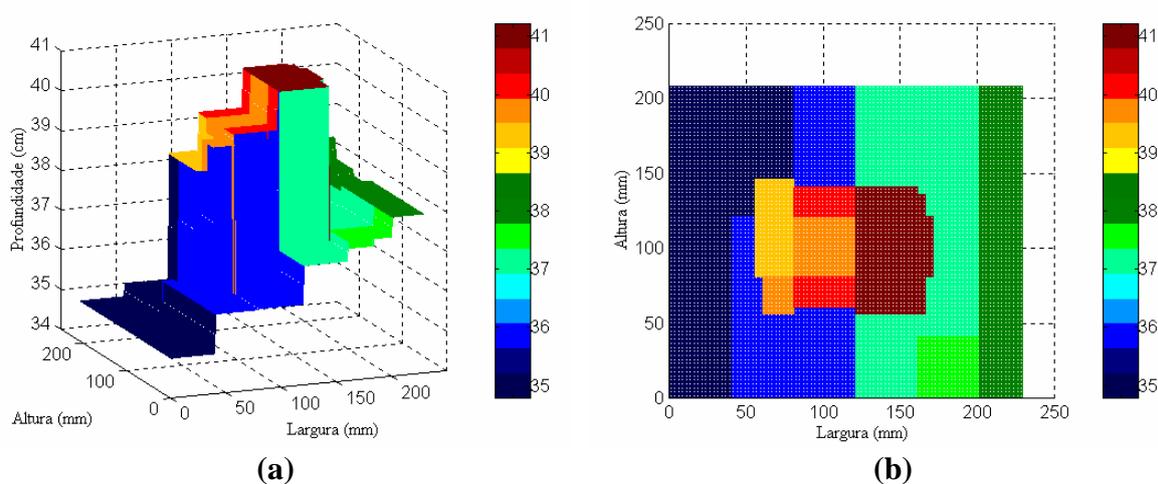


Figura 8.30 Mapa de superfície sem filtragem dos valores obtidos para as RPCs. (a) 3D; (b) 2D.

8.3 Apreciação dos resultados

A aquisição da superfície dos modelos utilizados foi bastante morosa devido essencialmente à baixa velocidade da onda sonora e ao número de medidas que são efectuadas para extrair uma RPC (aproximadamente 240 ms). Como facilmente se conclui o processo é tanto mais lento quanto maior for a irregularidade do relevo da superfície devido sobretudo ao processo de detecção dos pontos onde ocorrem as transições de profundidade. O tempo de aquisição da superfície do primeiro modelo testado foi de aproximadamente 8 min enquanto o do último modelo foi de aproximadamente 30 min. Tempos desta ordem de grandeza inviabilizam a utilização de um sistema deste género em aplicações que requeiram execução em tempo real.

A precisão dos mapas de superfície obtidos com este sistema é de aproximadamente 1.5 cm. Esta precisão e as restrições que as superfícies a adquirir têm que apresentar, impedem que seja viável a utilização de um sistema com esta configuração em aplicações industriais que requerem grande precisão e contacto, como são por exemplo a soldadura, esmerilagem e polimento de superfícies. Contudo os mapas de superfície assim adquiridos possibilitam para além do reconhecimento de objectos, ainda a definição de trajectórias para o seguimento de superfícies necessária a aplicações industriais que não requerem contacto como é o caso da pulverização à pistola quer para a pintura quer na aplicação de colas, diluentes ou outros líquidos em determinadas superfícies. Nestes casos poderia ser desenvolvido um algoritmo simples de validação da distância utilizando os ultrassons de forma a que a distância entre a pistola e a superfície fosse aproximadamente constante e permitisse a execução destas tarefas em tempo real. Simultaneamente será ainda possível a detecção de obstáculos que se interpussem entre a pistola e a superfície fornecendo indicações ao sistema para parar momentaneamente o processo ou procurar uma trajectória alternativa.

É ainda de referir que apesar dos 6 graus de liberdade que este robô possui, 3 dos quais no pulso, não foi possível orientar os sensores de ultrassons de forma a que estes ficassem sempre perpendiculares à superfície, o que possibilitaria a aquisição de relevos mais complexos (superfícies côncavas, convexas ou planas “escamadas” com qualquer inclinação). Esta impossibilidade deve-se a que os 3 graus de liberdade existentes no pulso foram implementados com duas juntas torcionais e uma rotativa. Caso tivessem sido implementados com duas juntas rotativas e uma torcional já era possível, de uma forma simples, fazer a orientação dos ultrassons.

Capítulo 9

9. Conclusões e sugestões para trabalho futuro

9.1 Conclusões

Esta tese aborda o estudo e desenvolvimento de técnicas de fusão/integração sensorial de dados provenientes de sensores de baixo custo (ultrassons da *Polaroid Inc.* e de uma câmara de vídeo monocromática), para a aquisição e posterior representação do relevo de objectos num mapa de superfície.

Os resultados obtidos podem ser resumidos nos seguintes pontos:

1. Estabelecimento de um modelo de referência RCS.
2. Concepção e desenvolvimento de uma infraestrutura experimental baseada num robô PUMA 560.
3. Desenvolvimento de um sistema de comunicações adaptado à transmissão de dados sensoriais.
4. Desenvolvimento de técnicas de processamento de imagem para obtenção dos pontos dominantes.
5. Definição de um modelo para os ultrassons.
6. Implementação de técnicas de fusão/integração sensorial.

Ponto 1: O modelo de referência utilizado forneceu a estrutura conceptual para a análise global do sistema e para o desenvolvimento do suporte lógico da aplicação. Este modelo revelou-se flexível e eficaz, permitindo uma integração relativamente simples das diferentes componentes do sistema. A diversidade dos dispositivos físicos necessários para o levantamento do relevo da superfície pretendida, foi igualmente bem integrada por esta arquitectura. A estrutura computacional inerente à arquitectura facilitará também a integração de diferentes linguagens de programação, caso seja necessário.

Ponto 2: Estudou-se a arquitectura original do PUMA 560 para avaliação das suas possibilidades de utilização no desenvolvimento de uma aplicação desta natureza. Foram detectados problemas na parte de controlo que restringiam a sua utilização, nomeadamente: recursos para sensorização que não eram suportados, a impossibilidade de realizar controlo em tempo real devido à inexistência de um porto ALTER e a comunicação com o robô ser

somente possível através do protocolo RS-232 a uma velocidade máxima de 9600 bit/s, o que tornava todo o processo lento.

Para resolver estes problemas o controlador do PUMA foi totalmente substituído por placas da *Trident Robotics* e respectivo *software*. Esta solução mostrou-se bastante eficiente no controlo do robô, após a resolução de algumas dificuldades iniciais que obrigaram a alterações do *software*. O controlo do PUMA passou a ser totalmente efectuado a partir do PC. Por outro lado, esta solução deixa ainda em aberto um conjunto de possibilidades, como por exemplo a utilização de outro tipo de controladores nas juntas do robô para além do convencional controlador PID (Proporcional + Integral + Derivativo).

Ponto 3: Foi desenvolvido um sistema de comunicações adaptado à transmissão de dados sensoriais. Este sistema de comunicações teve como base uma rede CAN em que o protocolo da camada de aplicação foi implementado sem obedecer a nenhuma das propostas de especificação existentes (Cal, CANOpen, PCAL e SDS). Esta opção deveu-se essencialmente a que as necessidades para esta aplicação em termos de volume de mensagens e número de nós (apenas um) ser reduzido.

A opção por este tipo de rede realçou quatro grandes virtudes: simplicidade do hardware necessário, facilidade de implementação de funções básicas de transferência (devido a não ser necessário a implementação de códigos de detecção/correção de erros), a facilidade com que é possível aumentar o número de nós sem modificações significativas ao nível do *software* implementado e fiabilidade (nenhum erro foi detectado na transmissão de dados durante todo o tempo de utilização da rede).

Ponto 4: Foi implementado um sistema de visão cuja finalidade é a restrição da área de trabalho do robô. Este sistema permite localizar um objecto na área de trabalho do robô, fazendo em seguida a sua aquisição a 2D para extracção dos pontos dominantes do contorno. A partir desta informação o robô é guiado para a posição onde se inicia o varrimento da superfície visível do objecto.

O sistema apresenta uma limitação no respeitante à cor dos objectos a utilizar: estes terão que apresentar uma cor contrastante com o fundo da mesa de trabalho. Contudo, apesar desta limitação o sistema mostrou-se robusto, nomeadamente ao apresentar alguma insensibilidade às condições de luminosidade, ruído e focagem imperfeita, conforme comprovam os resultados experimentais apresentados no Cap. 6.

Ponto 5: Foi definido um novo modelo para os ultrassons com base nos estudos já efectuados e nos testes experimentais realizados. Para definir o modelo foram introduzidos dois conceitos: pontos de profundidade constante (PPC) que permitem distinguir retornos fortes dos fracos (sendo só processados os fortes); regiões de profundidade constante (RPC) extraídas a partir dos pontos de PPC e que permitem definir uma área cuja profundidade é a mesma em qualquer ponto relativamente a um referencial (nesta aplicação o plano de deslocamento do robô).

Este modelo, em conjunto com a configuração de *hardware* adoptado para os ultrassons (Cap. 4), permitiram inclusivamente o abandono da utilização dos infravermelhos como detectores mais rigorosos (em virtude da sua melhor direcionalidade) das localizações onde se encontravam as diferenças de profundidade.

Ponto 6: Foram utilizadas a lógica *fuzzy*, um filtro de *Kalman* e a média como técnicas de fusão sensorial de dados e um método heurístico como técnica de integração sensorial de dados.

A lógica *fuzzy* não foi utilizada na sua forma mais “tradicional”, ou seja servindo de base à implementação de um controlador. Neste caso, foi utilizada para desenvolvimento de um “decisor” *fuzzy* que gera um valor de crença para cada uma das medidas de distância a um ponto (médias divergentes) com base na vizinhança das mesmas, permitindo desta forma seleccionar o valor mais correcto para esse ponto.

O filtro de *Kalman* funciona de uma forma semelhante ao “decisor” *fuzzy* atribuindo o valor de crença em cada medida (média do vector de medida) em função da variância dos dados que constituem o vector de medida (o valor da crença é mais elevado quanto menor for a variância dos dados).

A média é aplicada apenas quando nenhum dos dois métodos anteriores o foi.

O método heurístico é aplicado como técnica de integração sensorial de dados somente quando os ultrassons apontam para o limite do objecto.

A aplicação conjunta do modelo dos ultrassons e das técnicas de fusão/integração sensorial de dados produziram bons resultados, conforme se pode comprovar pelos resultados experimentais anteriormente apresentados.

9.2 Sugestões para trabalho futuro

O bom desempenho do sistema experimental de levantamento do relevo da superfície de objectos através da utilização de sensores de baixo custo em conjunto com técnicas de fusão/integração sensorial de dados estimula a continuação da investigação neste domínio.

Na linha de investigação desta dissertação, podemos identificar alguns pontos que merecem um estudo mais aprofundado:

- Geração automática de trajectórias com base no mapa de superfície obtido, para execução por exemplo de tarefas sem contacto com pulverização à pistola e com validação em tempo real da distância (por utilização dos ultrassons) entre esta e a superfície.
- Aumentar os graus de liberdade do sistema de forma poder posicionar os ultrassons (feixe) sempre na perpendicular à superfície a adquirir ou utilizando outro robô que tivesse 3 graus de liberdade no pulso mas implementados com duas juntas rotativas e uma torcional.
- Levantamento de superfícies com diferentes formatos, como por exemplo triangulares, circulares, com buracos, côncavas ou convexas.
- Estudo de outros sensores de baixo custo, como por exemplo a combinação entre díodos laser ou sensores de tacto com os ultrassons, de forma a permitir a detecção das diferentes profundidades com maior precisão e com menor diferença de profundidade entre elas.
- Cooperação entre manipuladores para a realização de tarefas complexas.
- Estudo de técnicas de optimização, como por exemplo os algoritmos genéticos, para avaliar diferentes disposições dos sensores com base na característica de diferentes superfícies de forma a seleccionar o melhor par disposição-características.

BIBLIOGRAFIA

- [Albus and Proctor, 96] J. S. Albus and F. G. Proctor. A Reference Model Architecture for Intelligent Hybrid Control Systems. Proceedings of the International Federation of Automatic Control, San Francisco, CA, June 30-July 5, 1996.
- [Albus, 91] J. S. Albus. Outline for a Theory of Intelligence. IEEE Transactions on Systems, Man and Cybernetics, Vol. 21, No.3, May/June 1991, pp 473-509.
- [Albus, 94] J. S. Albus. A Reference Model Architecture for Intelligent Systems Design. NISTIR 5502, National Institute of Standards and Technology, Gaithersburg, MD, September 1994.
- [Albus, 96a] J. S. Albus. The Engineering of Mind. Technical Report, National Institute of Standards and Technology, 1996.
- [Albus, 96b] James S. Albus. An Engineering Architecture for Intelligent Systems. Proceedings of the American Association for Artificial Intelligent (AAAI), Fall Symposium Series, 1996.
- [Allen and Michelman, 90] P. K. Allen and Paul Michelman. Acquisition and Interpretation of 3-D Sensor Data from Touch. IEEE Transactions on Robotics and Automation, Vol. 6, No. 4, August 1990, pp 397-404.
- [Allen, 88] P. K. Allen. Integrating Vision and Touch for Object Recognition Tasks. International Journal of Robotic Research (7), No. 6, 1988, pp 15-33.
- [Altrock, 95] Constantin Von Altrock. Fuzzy Logic & NeuroFuzzy Applications Explained. Prentice-Hall Inc, ISBN 0-13-368465-2, 1995.
- [Anderson and Donath, 90] Tracy L. Anderson and Max Donath. Animal Behavior as a Paradigm for Developing Robot Autonomy. Robotics and Autonomous Systems, 6, 1990, pp. 145-168.
- [Arkin, 89] Ronald C. Arkin. Towards the Unification of Navigational Planning and Reactive Control. Working Notes of the AAAI Spring Symposium on Robot Navigation Stanford University, March 28-30, 1989.
- [Arkin, 90] Ronald C. Arkin. Integrating Behavioral, Perceptual, and World Knowledge in Reactive Navigation. Robotics and Autonomous Systems, 6 (1990), pp. 105-122.
- [Azeredo et al., 96] J. Azeredo, J. Meinders and R. Oliveira. Determination of cell number and size of a population of Pseudomonas Fluorescens by

- image analysis. RECPAD'96, Portuguese Conference on Pattern Recognition, Guimarães-Portugal, pp 455-459, March 21-22, 1996.
- [Barshan and Kuc, 90] B. Barshan and R. Kuc. Differentiating Sonar Reflections from Corners and Planes by Employing an Intelligent Sensor. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(6):560-569, June 1990.
- [Bauzil et al, 81] G. Bauzil, M. Briot and P. Ribes. A navigation sub-system using ultrasonic sensors for the mobile robot HILARE. *Proceedings of the 1st Conference on Robot Vision and Sensory Controls*, Stratford-upon-Avon, UK, pp. 47-58 and pp. 681-698, 1981.
- [Beckerman and Oblow, 90] Martin Beckerman and E. M. Oblow. Treatment of Systematic Errors in the Processing of Wide-Angle Sonar Sensor Data for Robotic Navigation. *IEEE Transactions on Robotics and Automation*, Vol. 6, No. 2, April 1990, pp 137-145.
- [Borenstein and Koren, 89] J. Borenstein and Y. Koren. Real-Time Obstacle Avoidance for Fast Mobile Robots. *IEEE Transactions Systems Man and Cybernetics* 19, pp 1179-1189, September 1989.
- [Bosch, 91] Robert Bosch. CAN Specification version 2.0, 1991.
- [Brooks and Iyengar, 98] Richard R. Brooks and S. S. Iyengar. *Multi-Sensor Fusion Fundamentals and Applications with Software*. Prentice Hall, ISBN 0-13-901653-8, 1998.
- [Brooks, 85] Rodney A. Brooks. A Robust Layered Control System For A Mobile Robot. A. I. Memo 864, AI Laboratory, MIT, September 1985.
- [Brooks, 86] Rodney A. Brooks. Achieving Artificial Intelligence Through Building Robots. A. I. Memo 899, AI Laboratory, MIT, May 1986.
- [Brooks, 89] Rodney A. Brooks. A Robot that Walks; Emergent Behaviors from a Carefully Evolved Network. A. I. Memo 1091, AI Laboratory, MIT, February 1989.
- [Brooks, 90] Rodney A. Brooks. *The Behavior Language; User's Guide*. A. I. Memo 1227, AI Laboratory, MIT, April 1990.
- [Brooks, 91] Rodney A. Brooks. Integrated Systems Based on Behaviors. *SIGART Bulletin*, vol. 2, No. 4, August 1991, pp. 46-50.
- [Brooks, 94] R. A. Brooks. Coherent Behavior from Many Adaptive Processes. *Proceedings of the Third International Conference on Simulation of Adaptive Behavior*. pp 22-29.

- [CAN in Automation, 95] CAN in Automation International Users and Manufacturers Group. Data Link Layer. May 1995.
- [Cena and Valenzano, 97] Gianluca Cena and Adriano Valenzano. An Improved CAN Fieldbus for Industrial Applications. IEEE Transactions on Industrial Electronics, Vol. 44, No. 4, August 1997, pp 553-564.
- [Connell, 89] Jonathan H. Connell. A Behavior-Based Arm Controller. IEEE Transactions on Robotics and Automation, Vol. 5, No. 6, December 1989, pp 784-791.
- [Corke, 94] Peter Ian Corke. High-Performance Visual Closed-Loop Robot Control. PhD. Thesis, University of Melbourne, July 1994.
- [Corke, 96] Peter Ian Corke. Robotics Toolbox for use with MATLAB (release 4), Division of Manufacturing Technology, Preston, Australia, August 1996.
- [Craig, 89] John J. Craig. Introduction to Robotics – Mechanics and Control second edition. Addison Wesley Publishing Company, 1989.
- [Creative Labs, 92] Creative Labs. Video Blaster Developer Kit, 1992.
- [Crowley, 89] J. L. Crowley. World Modeling and Position Estimation for a Mobile Robot Using Ultra-Sonic Ranging. In Proc. IEEE Int. Conf. Robotics and Automation, pp 674-681.
- [Doty and Iglesia, 96] Keith L. Doty and Erik de la Iglesia. Sharp IR Sensor for Hack for Analog Distance Measurement. University of Florida, Department of Electrical Engineering, EEL 5934: Intelligent Machines Design Laboratory, 1996.
- [Drumheller, 87] M. Drumheller. Mobile Robot Location Using Sonar. IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-9(2):325-332, March 1987.
- [E. Davies, 90] E. Davies. Machine Vision: Theory, Algorithms, Practicalities. Academic Press, 1990.
- [Elfes, 87] Alberto Elfes. Sonar-Based Real-World Mapping and Navigation. IEEE Journal of Robotics and Automation, Vol. RA-3, No. 3, June 1987, pp 249-265.
- [Flynn, 88] A. M. Flynn. Combining Sonar and Infrared Sensors for Mobile Robot Navigation. The International Journal of Robotics Research, Vol. 7, No. 6, pp 5-14, December 1988.
- [Gonzalez and Wintz, 87] R. Gonzalez and P. Wintz. Digital Image Processing. Addison Wesley, 2^a edition, 1987.

- [Gonzalez and Woods, 92] Rafael C. Gonzalez and Richard E. Woods. Digital Image Processing. Addison-Wesley Publishing Company, ISBN 0-201-50803-6, 1992.
- [Goodridge et al., 96] Steven G. Goodridge, Michael G. Kay, and Ren C. Luo. Multilayered Fuzzy Behavior Fusion for Real-Time Reactive Control of Systems with Multiple Sensors. IEEE Transactions on Industrial Electronics, Vol. 43, No. 3, June 1996.
- [Hall, 92] David L. Hall. Mathematical Techniques in Multisensor Data Fusion. Artech House, ISBN 0-89006-558-6, 1992.
- [Harris and Recce, 98] Kenneth D. Harris and Michael Recce. Experimental modelling of time-of-flight sonar. Robotics and Autonomous Systems 24 (1998) 33-42.
- [Huang, 96] Hui-Min Huang. An Architecture and a Methodology for Intelligent Control. IEEE Expert, April 1996.
- [Hutchinson and Kak, 89] Seth A. Hutchinson and Avinash C. Hak. Planning Sensing Strategies in a Robot Work Cell with Multi-Sensor Capabilities. IEEE Transactions on Robotics and Automation, Vol. 5, No. 6, December 1989, pp 765-782.
- [IEC, 97] International Electrotechnical Commission (IEC). IEC 1131-Programmable Controllers, Part 7 – Fuzzy Control Programming, Januar 1997.
- [Inform Software Corporation, 96] Inform Software Corporation. FuzzyTECH reference manual, 1996.
- [Jang et al, 97] J.-S. R. Jang, C.-T. Sun E. Mizutani. Neuro-Fuzzy And Soft Computing – A computational approach to learning and machine intelligence. Prentice-Hall Inc, ISBN 0-13-261066-3, 1997.
- [K. S. Fu et al., 87] K. S. Fu, R. C. Gonzalez, and C. S. G. Lee. Robotics Control, Sensing, Visio, and Intelligence. New York, McGraw-Hill, ISBN 0070226253, 1987.
- [Kasturi and Trivedi, 90] R. Kasturi and M. Trivedi. Image Analysis Applications. Marcel Dekker, 1990.
- [Kuc and Siegel, 87] Roman Kuc and M. W. Siegel. Physically Based Simulation Model for Acoustic Sensor Robot Navigation. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. PAMI-9, No. 6, November 1987, pp 766-778.
- [Leonard and Durrant-White, 92] John J. Leonard and Hugh F. Durrant-Whyte. Directed Sonar Sensing for Mobile Robot Navigation. Kluwer Academic Publishers, ISBN 0-7923-9242-6, 1992.

- [Lima and Campilho, 94] R. Lima and A. Campilho. Thresholding selection methods: a comparative study. Proc. RECPAD'94, Portuguese Conference on Pattern Recognition, 1994.
- [Luo and Kay, 89] Ren C. Luo and Michael G. Kay. Multisensor Integration and Fusion in Intelligent Systems. IEEE Transactions on Systems, Man, and Cybernetics, Vol. 19, No. 5, September/October 1989, pp 901-931.
- [Luo and Kay, 95] Ren C. Luo and Michael G. Kay. Multisensor Integration and Fusion for Intelligent Machines and Systems. Edited by Ren C. Luo and Michael G. Kay, ISBN 0-89391-836-6, 1995.
- [Luo et al, 88] Ren C. Luo, Min-Hsiung Lin and Ralph S. Scherp. Dynamic Multi-Sensor Data Fusion System for Intelligent Robots. IEEE Journal of Robotics and Automation, Vol. 4, No. 4, August 1988, pp 386-396.
- [Maes and Brooks, 91] Pattie Maes and Rodney A. Brooks. Learning to Coordinate Behaviors. In Engelmores [Engelmores91], pp. 796-802.
- [Maes, 89] Pattie Maes. How to do the Right Thing. A. I. Memo 1180, AI Laboratory, MIT, December 1989.
- [Malcolm and Smithers, 90] Chris Malcolm and Tim Smithers. Symbol Grounding via a Hybrid Architecture in an Autonomous Assembly System. Robotics and Autonomous Systems, 6 (1990), pp. 123-144.
- [Manyika and Durrant-White, 94] James Manyika and Hugh Durrant-Whyte. Data Fusion and Sensor Management a decentralized information-theoretic approach. Ellis Horwood, Series in Electric and Electronic Engineering, ISBN 0-13-30-3132-2, 1994.
- [Maslin, 83] Gerald D. Maslin. A Simple Ultrasonic Ranging System. 102nd Convention of the Audio Engineering Society at Cincinnati, Ohio on May 12, 1983.
- [Mataric, 92a] Maja J. Mataric. Behavior-Based Control: Main Properties and Implications. Proceedings, IEEE International Conference on Robotics and Automation, Workshop on Architectures for Intelligent Control Systems, Nice, France, May 1992, 46-54.
- [Mataric, 92b] Maja J. Mataric. Integration of Representation Into Goal-Driven behavior-Based Robots. IEEE Transactions on Robotics and Automation, Vol. 8, No. 3, June 1992, pp 304-312.
- [Mataric, 97] Maja J. Mataric. Behavior-Based Control: Examples from Navigation, Learning, and Group Behavior. Journal of Experimental and Theoretical Artificial Intelligence, Special issue

- on Software Architectures for Physical Agents, 9(2-3), H. Hexmoor, I. Horswill, and D. Kortenkamp, eds., 1997, 323-336.
- [Mataric, 98] Maja J. Mataric. Behavior-Based Robotics as a Tool for Synthesis of Artificial Behavior and Analysis of Natural Behavior. *Trends in Cognitive Science*, Vol. 2, No. 3, pp 82-87, March 1998.
- [Monteiro, 97] Nuno Duarte Monteiro. *Processamento e Análise de Imagem Para a Apreensão e Descrição de Contornos de Objectos*. Relatório de estágio realizado no âmbito deste projecto de doutoramento por o aluno Nuno Monteiro da licenciatura de LESI sob a orientação de Jaime Fonseca, Universidade do Minho, Maio de 1997.
- [Moreira et al., 96] Nuno Moreira, Paulo Alvito, Pedro Lima. First Steps Towards an Open Control Architecture for a PUMA 560. *Proceedings of the 2nd Portuguese Conference on Automatic Control*, Porto, Portugal, September 11-13, 1996.
- [Nashman, 93] Marilyn Nashman. The Use of Vision and Touch Sensors for Dimensional Inspection Tasks. *The American Society of Mechanical Engineers, "Manufacturing Review"*, vol. 6, N° 2, June 1993.
- [Nunes, 95] Urbano José Carreira Nunes. *Controlo de Robôs com Realimentação Sensorial no Espaço Tarefa*. Tese de Doutoramento, Universidade de Coimbra, Fevereiro de 1995.
- [Pearl, 88] Judea Pearl. *Probabilistic Reasoning in Intelligent Systems: Network of Plausible Inference*. Morgan Kaufmann Publishers Inc, ISBN 1-55860-479-0, 1988.
- [Polaroid, 95] Polaroid, Ultrasonic Ranging Developer's Kit.
- [Rodriguez, 94] Miguel Rodriguez. *Modélisation d'un Agent Autonome: Approche Constructiviste de L'Architecture de Contrôle et de la Représentation des Connaissances*. Thèse pour obtenir le grade de Docteur ès sciences, Université de Neuchâtel, Suisse, 1994.
- [Roth and Schilling, 96] H. Roth and K. Schilling. Hierarchically Organised Control Strategies For Mobile Robots Based on Fused Sensor Data. *Proceedings of Mechatronics'96*, Guimarães-Portugal, 18-20 September 1996, pp 95-100.
- [Russo and Ramponi, 94] Fabrizio Russo, Giovanni Ramponi. Fuzzy Methods for Multisensor Data Fusion. *IEEE Transactions on Instrumentation and Measurement*, Vol. 43, No. 2, April 1994, pp 288-294.
- [Santos, 95] Vítor Manuel Ferreira dos Santos. *Navegação Autónoma de Robots: Interpretação dos dados sensoriais e Navegação Local*. Tese de Doutoramento, Universidade de Aveiro, Junho de 1995.

- [Santos, 97] Cristina Santos. Uma Exploração do Modelo de Referência RCS e da Fusão Sensorial em Robótica. Tese de Mestrado em Electrónica Industrial, Universidade do Minho, 1997.
- [Shaulkoff, 89] R. Shaulkoff. Digital Image Processing and Computer Vision. John Wiley, 1989.
- [Song and Tang, 96] Kai-Tai Song and Wen-Hui Tang. Environment Perception for a Mobile Robot Using Double Ultrasonic Sensors and a CCD Camera. IEEE Transactions on Industrial Electronics, Vol. 43, No. 3, June 1996, pp 372-379.
- [STZP, 93a] STZP. Universal CAN I/O board – Manual. August 1993.
- [STZP, 93b] STZP. CAN-DLL for PCI 02 – User Manual. December 1993.
- [STZP, 95] STZP. PC-CAN Interface PCI 02 – User Manual. June 1995.
- [Tavares, 93] A. Tavares. Técnicas de Visão Computacional para Reconhecimento e Medição de Aparas de Madeira. Tese de Mestrado, Universidade de Coimbra, 1993.
- [Tech Reach Inc, 95] Tech Reach Inc. An Introduction to Multisensor Data Fusion. October 21, 1995.
- [Trident Robotics, 94a] Trident Robotics and Research, Inc. User's Manual TRC100, January 1994.
- [Trident Robotics, 94b] Trident Robotics and Research, Inc. User's Manual TRC004, April 1994.
- [Tung, 96] Chao-Ping Tung. Integrating Sensing, Task Planning, and Execution for Robotic Assembly. IEEE Transactions on Robotics and Automation, Vol. 12, No. 2, April 1996, pp 187-201.
- [Unimation, 83] Unimation. Technical Manual Users Guide to Val. Unimation Robotics, February 1983.
- [Welch and Bishop, 95] Greg Welch and Gary Bishop. An Introduction to the Kalman Filter. University of North Carolina, Department of Computer Science, TR 95-041.
- [Zadeh, 65] L. A. Zadeh. Fuzzy Sets. Informational Control, vol.8, pp. 338-353, 1965.

ANEXOS

ANEXO I

CINEMÁTICA DO MANIPULADOR PUMA 560

A cinemática é o estudo do movimento sem ter em consideração as forças que lhe deram origem, sendo estudadas a posição, velocidade e aceleração e todas as derivações das variáveis de posição. A cinemática dos robôs do tipo manipulador envolve o estudo da geometria no instante em que se efectua o movimento, em particular como é que as diversas ligações se movem relativamente umas às outras [Craig, 89].

Os manipuladores são constituídos por um conjunto de elementos mecânicos, denominados ligações, conectados entre si por juntas. Cada junta tem um grau de liberdade: translacional ou rotacional. Para um manipulador com n juntas numeradas de 1 até n , existem $n+1$ ligações numeradas de 0 até n . A ligação 0 é a base do manipulador, geralmente fixa, as n ligações transportam a parte terminal até ao ponto pretendido. A junta i conecta as ligações i e $i-1$.

Qualquer robô pode ser descrito cinematicamente através de 4 parâmetros para cada ligação. Dois parâmetros descrevem a própria ligação e outros dois descrevem a junta que faz a conexão entre duas ligações adjacentes.

Os parâmetros da ligação definem a localização relativa de dois eixos no espaço (Figura A.1), sendo denominados **comprimento da ligação** e **ângulo de torção**. O comprimento da ligação, a_{i-1} , é a distância entre juntas adjacentes, medida ao longo da normal comum. O ângulo de torção, α_{i-1} , representa o ângulo formado pelas juntas adjacentes.

Os parâmetros da junta (Figura I.1) são a **distância da junta** e o **ângulo da junta**. A distância da junta, d_i , representa a distância entre as ligações adjacentes à junta i medida ao longo da normal comum. O ângulo da junta, θ_i , representa o ângulo formado pelas ligações adjacentes à junta i .

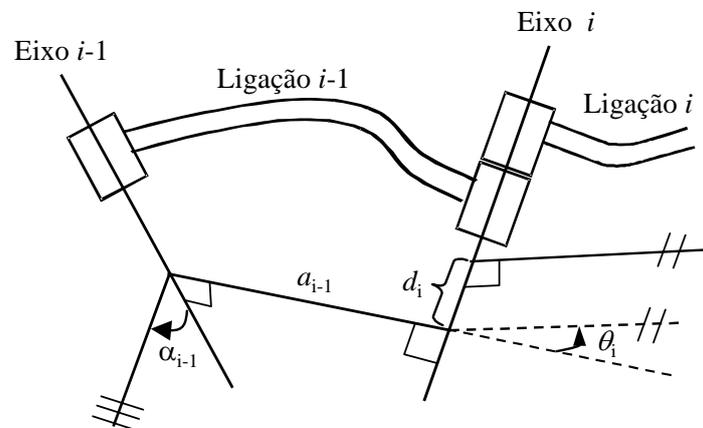


Figura I.1 Parâmetros da ligação e da junta.

Para simplificar a descrição da localização de cada ligação foi definido um quadro de coordenadas, em que o quadro i está relacionado com a ligação i . A figura I.2 apresenta a localização dos quadros $\{i-1\}$ e $\{i\}$ para um manipulador genérico.

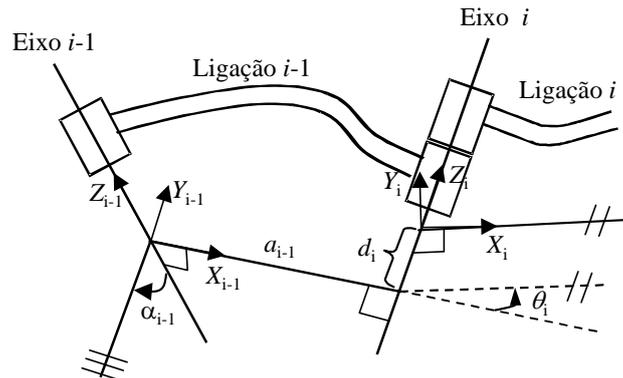


Figura I.2 Localização dos quadros $\{i-1\}$ e $\{i\}$ para um manipulador genérico.

Os parâmetros da ligação e da junta para o caso da Figura I.2 podem ser resumidos da seguinte forma:

Comprimento da ligação a_i – Distância entre os eixos Z_i e Z_{i+1} medida ao longo de X_i .

Ângulo de torção α_i – ângulo entre Z_i e Z_{i+1} , medido em torno de X_i .

Distância da junta d_i – Distância entre X_{i-1} e X_i , medido ao longo de Z_i .

ângulo da junta θ_i – ângulo entre X_{i-1} e X_i , medido em torno de Z_i .

Denavit e Hartenberg (DH) propuseram um método matricial para fazerem uma atribuição sistemática das coordenadas do sistema a cada ligação, representando o resultado numa matriz de transformação homogénea de 4×4 .

$${}^{i-1}T_i = \begin{bmatrix} \cos \theta_i & -\sin \theta_i \cos \alpha_i & \sin \theta_i \sin \alpha_i & a_i \cos \theta_i \\ \sin \theta_i & \cos \theta_i \cos \alpha_i & -\cos \theta_i \sin \alpha_i & a_i \sin \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (\text{A1.1})$$

Cada quadro de coordenadas representa a ligação relativamente ao sistema de coordenadas da ligação anterior, isto é

$${}^0T_i = {}^0T_{i-1} {}^{i-1}T_i \quad (\text{A1.2})$$

onde 0T_i é a transformação homogénea que descreve a relação entre o sistema de coordenadas i relativamente ao sistema de coordenadas iniciais do mundo (0).

Pârametros cinemáticos para o PUMA 560

Os parâmetros cinemáticos do manipulador PUMA 560 são necessários para o cálculo da cinemática directa e inversa do manipulador. Na tabela I-1 encontram-se expressos os valores dos parâmetros de DH para o PUMA 560 utilizado nesta aplicação.

Junta	α_i (graus)	a_i (mm)	d_i (mm)	θ_i	θ_{\min} (graus)	θ_{\max} (graus)
1	-90	0	0	θ_1	-160	160

2	0	431.8	0	θ_2	-225	40
3	90	-20.3	150	θ_3	-51	232
4	-90	0	433.07	θ_4	-110	170
5	90	0	0	θ_5	-100	100
6	0	0	0	θ_6	-270	270

Tabela I-1 Parâmetros para o PUMA 560.

Utilizando a matriz de transformação homogénea (A1.1), que permite relacionar o quadro de coordenadas associado a uma junta com o quadro de coordenadas da junta anterior, foi possível obter as seguintes matrizes para cada par de quadros:

$${}^0T_1 = \begin{bmatrix} \cos \theta_1 & 0 & -\sin \theta_1 & 0 \\ \sin \theta_1 & 0 & \cos \theta_1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^1T_2 = \begin{bmatrix} \cos \theta_2 & -\sin \theta_2 & 0 & a_2 \cos \theta_2 \\ \sin \theta_2 & \cos \theta_2 & 0 & a_2 \sin \theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^2T_3 = \begin{bmatrix} \cos \theta_3 & 0 & \sin \theta_3 & a_3 \cos \theta_3 \\ \sin \theta_3 & 0 & -\cos \theta_3 & a_3 \sin \theta_3 \\ 0 & 1 & 0 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^3T_4 = \begin{bmatrix} \cos \theta_4 & 0 & -\sin \theta_4 & 0 \\ \sin \theta_4 & 0 & -\cos \theta_4 & 0 \\ 0 & -1 & 0 & d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$${}^4T_5 = \begin{bmatrix} \cos \theta_5 & 0 & \sin \theta_5 & 0 \\ \sin \theta_5 & 0 & -\cos \theta_5 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad {}^5T_6 = \begin{bmatrix} \cos \theta_6 & -\sin \theta_6 & 0 & 0 \\ \sin \theta_6 & \cos \theta_6 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Cinemática directa

A posição do manipulador PUMA 560 pode ser especificada pela matriz homogénea, wT_T , que descreve a posição e orientação do referencial da ferramenta, $\{T\}$, relativamente ao referencial do mundo, $\{W\}$. Trata-se de uma representação no espaço Cartesiano.

$${}^0T_6 = \begin{bmatrix} & | & & \\ \mathbf{R} & | & \mathbf{p} & \\ \hline & & & \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} & | & & \\ \mathbf{n} | \mathbf{o} | \mathbf{a} & | & \mathbf{p} & \\ \hline & & & \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (\text{A1.3})$$

Outra forma de representar matematicamente a posição do manipulador é através do vector J que contém os seis ângulos da junta (trata-se de um representação no espaço das juntas):

$$J = [J_1 \ J_2 \ J_3 \ J_4 \ J_5 \ J_6]^T \quad (A1.4)$$

A conversão entre a representação no espaço das juntas, J , e a representação no espaço cartesiano, wT_T , é designado por **transformação cinemática directa**.

Os elementos de wT_T podem ser obtidos repetindo a expressão (A1.2)

$${}^0T_n = {}^0T_1 \ {}^1T_2 \dots \ {}^{n-1}T_n \quad (A1.5)$$

a qual é o produto das matrizes de transformação para cada ligação. Para um manipulador com 6 graus de liberdade, 3 em translação e 3 em rotação, a posição e orientação do ponto terminal relativamente ao quadro de base do sistema é dada por

$${}^0T_6 = {}^0T_1 \ {}^1T_2 \ {}^2T_3 \ {}^3T_4 \ {}^4T_5 \ {}^5T_6 \quad (A1.6)$$

Os elementos da matriz 0T_6 podem ser calculados pelas seguintes expressões:

$$\begin{aligned} n_x &= c_1[c_{23}(c_4c_5c_6 - s_4s_6) - s_{23}s_5c_6] - s_1(s_4c_5c_6 + c_4s_6) \\ n_y &= s_1[c_{23}(c_4c_5c_6 - s_4s_6) - s_{23}s_5c_6] + c_1(s_4c_5c_6 + c_4s_6) \\ n_z &= -s_{23}[c_4c_5c_6 - s_4s_6] - c_{23}s_5c_6 \end{aligned}$$

$$\begin{aligned} o_x &= c_1[-c_{23}(c_4c_5s_6 + s_4c_6) + s_{23}s_5s_6] - s_1(-s_4c_5s_6 + c_4c_6) \\ o_y &= s_1[-c_{23}(c_4c_5s_6 + s_4c_6) + s_{23}s_5s_6] + c_1(-s_4c_5s_6 + c_4c_6) \\ o_z &= s_{23}(c_4c_5s_6 + s_4c_6) + c_{23}s_5s_6 \end{aligned}$$

$$\begin{aligned} a_x &= c_1(c_{23}c_4s_5 + s_{23}c_5) - s_1s_4s_5 \\ a_y &= s_1(c_{23}c_4s_5 + s_{23}c_5) + c_1s_4s_5 \\ a_z &= -s_{23}c_4s_5 + c_{23}c_5 \end{aligned}$$

$$\begin{aligned} p_x &= c_1(s_{23}d_4 + a_3c_{23} + a_2c_2) - s_1d_3 \\ p_y &= s_1(s_{23}d_4 + a_3c_{23} + a_2c_2) + c_1d_3 \\ p_z &= c_{23}d_4 - a_3s_{23} - a_2s_2 \end{aligned}$$

onde

$$c_i = \cos(\theta_i), \quad s_i = \sin(\theta_i), \quad c_{ik} = \cos(\theta_i + \theta_k) \quad e \quad c_{ik} = \sin(\theta_i + \theta_k)$$

As equações anteriores constituem a cinemática directa para o PUMA 560. Estas especificam como se calcula a posição e orientação do quadro {6} relativamente ao quadro {0} do robô.

Cinemática inversa

A transformação cinemática inversa consiste em: dada a posição e a orientação relativamente ao mundo, $\{W\}$, do referencial da ferramenta, $\{T\}$, na forma de matriz homogénea wT_T , obter o vector de ângulos das juntas correspondentes, J , de forma a que a ferramenta terminal seja posicionada conforme o pretendido.

Para o manipulador PUMA 560 a transformação cinemática inversa não tem uma solução única. A solução particular escolhida depende de quatro variáveis que nos indicam a configuração em que se encontra o braço. Estas quatro variáveis permitem escolher uma solução entre várias possíveis, sendo definidas em [K. S. Fu et al., 87] como *ARM*, *ELBOW*, *WRIST* e *FLIP*.

$$ARM = \begin{cases} +1, & \text{braço à direita dos ombros} \\ -1, & \text{braço à esquerda dos ombros} \end{cases}$$

$$ELBOW = \begin{cases} +1, & \text{cotovelo acima dos ombros} \\ -1, & \text{cotovelo abaixo dos ombros} \end{cases}$$

$$WRIST = \begin{cases} +1, & \text{pulso para cima} \\ -1, & \text{pulso para baixo} \end{cases} \quad FLIP = \begin{cases} +1, & \text{torcer a orientação do pulso} \\ 0, & \text{não torcer a orientação do pulso} \end{cases}$$

As expressões implementadas para efectuar a transformação cinemática inversa foram:

$$J_1 = \text{atan2}(p_y, p_x) - \text{atan2}(d_3, \sqrt{p_x^2 + p_y^2 - d_3^2})$$

$$J_3 = \text{atan2}(k, \sqrt{w}) - \text{atan2}(a_3, d_4)$$

onde,

$$k = (c_1 \times p_x + s_1 \times p_y)^2 + p_z^2 - d_4^2 - a_3^2 - a_2^2$$

$$w = 4 \times a_2^2 \times (a_3^2 + d_4^2) - k^2$$

(O J_3 é sempre calculado antes do J_2)

$$J_2 = J_{23} - J_3$$

onde,

$$J_{23} = \text{atan2}(((d_4 + a_2 \times s_3) \times k - (a_2 \times c_3 + a_3) \times p_z), ((a_2 \times c_3 + a_3) \times k - (d_4 + a_2 \times s_3) \times p_z))$$

$$J_4 = \text{atan2}(s_1 \times a_x - c_1 \times a_y, -c_{23} \times (c_1 \times a_x + s_1 \times a_y) + s_{23} \times a_z)$$

$$J_5 = \text{atan2}(w_1, s_{23} \times (c_1 \times a_x + s_1 \times a_y) - c_{23} \times a_z)$$

onde,

$$w_1 = c_4 \times (c_{23} \times (c_1 \times a_x + s_1 \times a_y) - s_{23} \times a_z) + s_4 \times (-s_1 \times a_x + c_1 \times a_y)$$

$$J_6 = \text{atan2} \left(\begin{array}{l} -c_5 \times (c_4 \times (c_{23} \times d - s_{23} \times o_z) + s_4 \times e) + s_5 \times (s_{23} \times d + c_{23} \times o_z), \\ -s_4 \times (c_{23} \times d - s_{23} \times o_z) + c_4 \times e \end{array} \right)$$

onde,

$$d = c_1 \times o_x + s_1 \times o_y$$

$$e = -s_1 \times o_x + c_1 \times o_y$$

É de salientar que a função ‘atan2’ existe no C nomeadamente no *Borland C++* utilizado para implementar esta aplicação.

No caso deste trabalho foram analisadas e corrigidas as primitiva ‘*pumaFwdKin*’ e ‘*pumaInvKin*’ (implementam a transformação cinemática directa e inversa respectivamente) definidas pela *Trident Robotics*, tendo sido efectuadas simulações com a *Robotic Toolbox for use with MATLAB* [Corke, 96] para comparação dos resultados.

ANEXO II

CARACTERÍSTICAS DAS PLACAS TRC041, TRC004 E TRC100 QUE SUBSTITUÍRAM O ANTIGO CONTROLADOR

ANEXO III

CARACTERÍSTICAS DA PLACA *POLAROID*

ANEXO IV

CARACTERÍSTICAS DAS PLACAS DE COMUNICAÇÃO PCI 02 E UNIVERSAL CAN I/O

ANEXO V

FOTOGRAFIAS DA INFRAESTRUTURA EXPERIMENTAL UTILIZADA

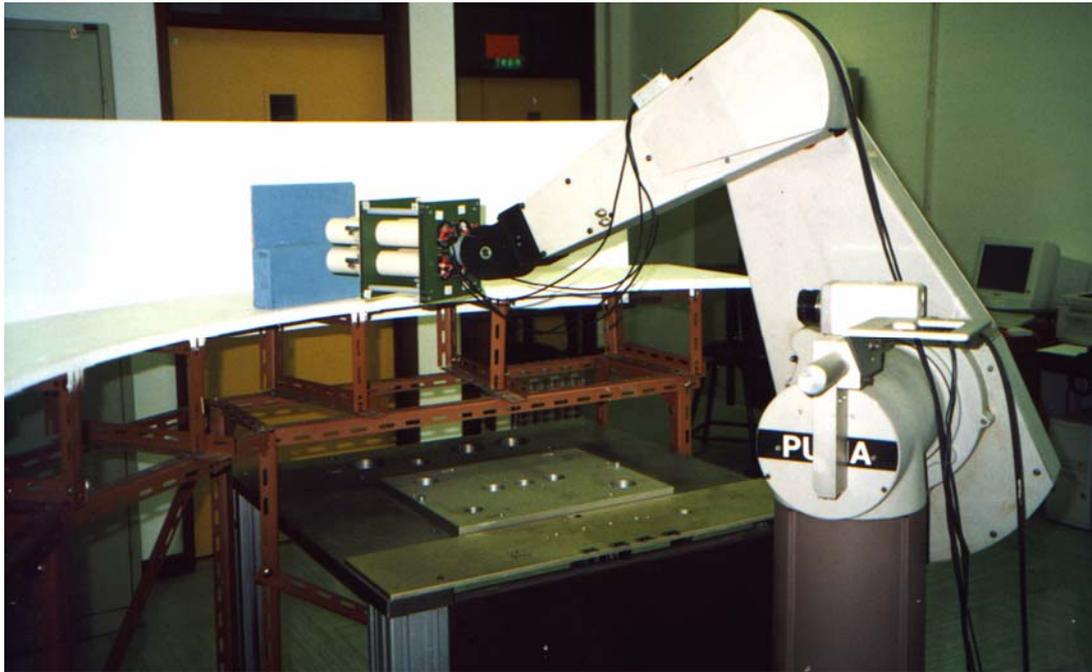


Figura V.1 Manipulador PUMA 560 com os sensores de ultrassons montados no punho e a câmara de vídeo sobre o ombro. Mesa onde são colocados os objectos que servem de modelo.

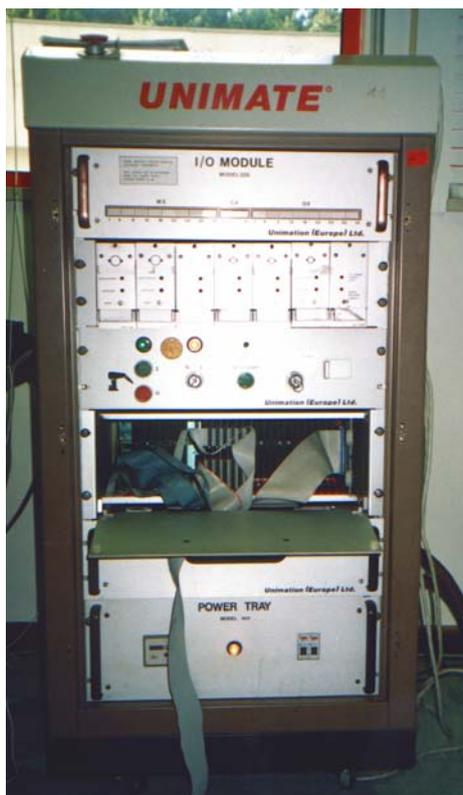


Figura V.2 Controlador do PUMA 560 onde é possível observar as 6 placas TRC041 que vão buscar os sinais de controlo do barramento para a TRC004. O *flat cable* liga a TRC004 à TRC100.

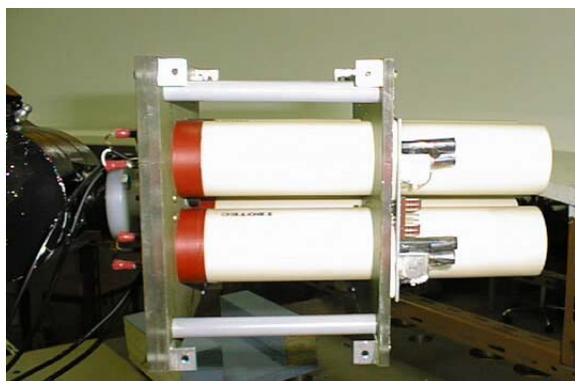


Figura V.3 Vista lateral e frontal dos ultrassons e infravermelhos que foram montados no punho do robô.

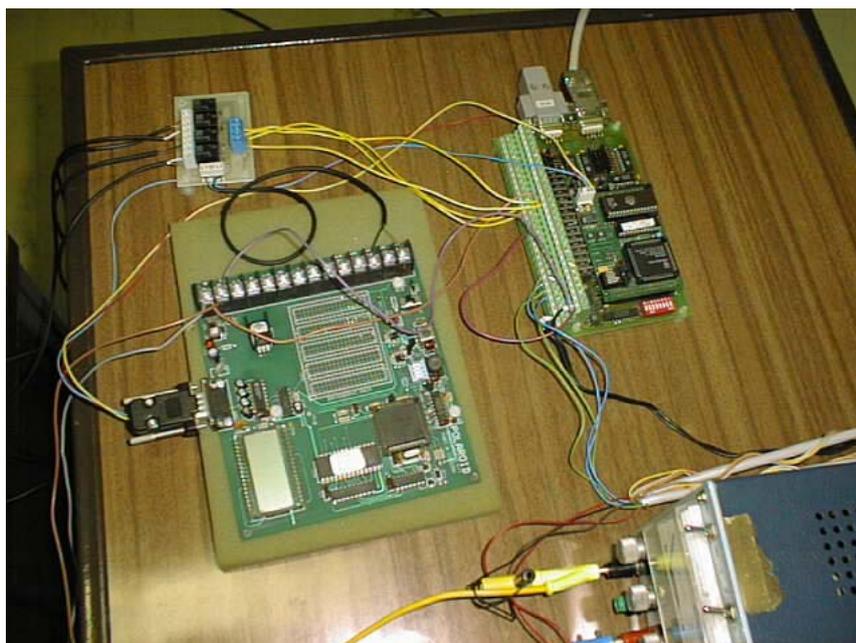


Figura V.4 Placa da *Polaroid INC.*, a Universal CAN I/O e a placa com relés que permite a activação de um único ultrassom num determinado instante.

