



UNIVERSIDADE DE COIMBRA  
FACULDADE DE CIÊNCIAS E TECNOLOGIA  
DEPARTAMENTO DE ENGENHARIA ELECTROTÉCNICA E DE COMPUTADORES

# DETECÇÃO E CLASSIFICAÇÃO DE OBJECTOS EM AMBIENTE EXTERIOR PARA VEÍCULOS AUTÓNOMOS

Cristiano Premebida

Coimbra, 2006

# DETECÇÃO E CLASSIFICAÇÃO DE OBJECTOS EM AMBIENTE EXTERIOR PARA VEÍCULOS AUTÓNOMOS

Dissertação Submetida ao Departamento de Engenharia Electrotécnica da  
Faculdade de Ciências e Tecnologia da Universidade de Coimbra  
satisfazendo parcialmente os requisitos para a obtenção do grau de  
Mestre em Engenharia Electrotécnica e de Computadores,  
área de especialização em Automação e Robótica

Submetida por  
Cristiano Premebida

Sob orientação de  
Prof. Doutor Urbano José Carreira Nunes

Coimbra, 2006



# Resumo

A presente dissertação apresenta uma arquitectura multi-módulos desenvolvida para realizar a detecção, seguimento e classificação de objectos em ambiente exterior semi-estruturado utilizando para esse efeito a informação de um sensor laser scanner (LRF), como sensor principal, e de uma câmara de visão. Este trabalho enquadra-se no âmbito de um projecto mais global, que é o sistema de seguimento e detecção de múltiplos alvos (MTDTS), em desenvolvimento no Instituto de Sistemas e Robótica - Pólo de Coimbra (ISR-UC), e teve como principal objectivo investigar modelos de associação de dados e a classificação de objectos para o referido sistema. O trabalho incidiu maioritariamente no processamento de dados do LRF, designadamente nas tarefas de segmentação, seguimento, associação de dados e classificação. Para a classificação dos objectos propõe-se um classificador do tipo modelos de misturas Gaussianas (GMM), utilizando regras de decisão do tipo máxima verosimilhança (ML) e máxima probabilidade posterior (MAP), que foi projectado para tratar dos dados obtidos com o LRF. Um módulo de combinação foi implementado para utilizar a informação de outros classificadores disponíveis, utilizados no âmbito do projecto MTDTS, de forma a se obter mais robustez na classificação; as categorias dos objectos de interesse são: automóveis, árvores, postes, e peões. Os resultados experimentais, usando dados adquiridos em dois ambientes exteriores do campo universitário, demonstram a fiabilidade da arquitectura proposta nesta dissertação e validam, satisfatoriamente, a utilização do classificador GMM.



# Abstract

This thesis presents a multi-module architecture developed to cope with object detection, tracking and classification in semi-structured outdoor scenarios using sensory information from a Laser Range Finder (LRF), as the main sensor, and from a video camera. This work is part of a more general Multi-Target Detection and Tracking System (MTDTS), under development in ISR-UC, aiming to investigate data association techniques and object classification methods for the referred system. The experimental results are focused on using laser based data, emphasizing the segmentation, tracking, data association and classification tasks. In order to classify the detected objects, a Gaussian Mixture Model (GMM) classifier is proposed, using Maximum Likelihood (ML) and Maximum *A Posteriori* (MAP) decision rules, designed to treat the data coming from the LRF. A classifier combination module was implemented using the information from other classifiers available in the MTDTS system, in order to achieve more robustness in the classification; the object classes of interest are: vehicles, tree trunks, posts, and pedestrians. Experiments using real data acquired in two outdoor environments on the campus demonstrate the feasibility of the proposed architecture and validate satisfactorily the utilization of the GMM classifier.



# Índice

<b>Lista de Figuras</b>	<b>xiii</b>
<b>Lista de Tabelas</b>	<b>xv</b>
<b>Lista de Acrónimos e Glossário</b>	<b>xviii</b>
<b>1 Introdução</b>	<b>1</b>
1.1 Motivação . . . . .	2
1.2 Enquadramento do trabalho . . . . .	5
1.3 Objectivos da dissertação . . . . .	6
1.4 Trabalho realizado e contribuições . . . . .	8
1.5 Organização da tese . . . . .	10
<b>2 Aplicações do sensor LRF em sistemas MTDTS</b>	<b>11</b>
2.1 Características do sensor LRF . . . . .	13
2.2 Objectos sob seguimento usando LRF . . . . .	15
2.3 Segmentação e extracção de características . . . . .	17
2.4 Seguimento e associação de dados . . . . .	18
2.5 Classificador-GMM e combinação de classificadores . . . . .	18
2.6 Sistema MTDTS . . . . .	19
<b>3 Segmentação</b>	<b>21</b>
3.1 Métodos baseados na distância Euclidiana . . . . .	24



3.2	Métodos baseados no filtro de Kalman . . . . .	27
3.3	Métodos baseados no filtro EKF . . . . .	29
<b>4</b>	<b>Extracção de primitivas geométricas</b>	<b>37</b>
4.1	Formulação do problema . . . . .	38
4.2	Extracção de rectas . . . . .	40
4.3	Extracção de círculos . . . . .	42
4.4	Extracção de elipses . . . . .	45
4.5	Extracção de primitivas com filtros de Kalman . . . . .	46
<b>5</b>	<b>Seguimento de objectos</b>	<b>51</b>
5.1	Representação em espaço-de-estados . . . . .	52
5.1.1	Modelos cinemáticos utilizados . . . . .	53
5.1.2	Covariância dos ruídos do processo e das medidas . . . . .	56
5.2	Associação de dados . . . . .	57
5.2.1	Região de validação elipsoidal . . . . .	58
5.2.2	Região de validação rectangular . . . . .	59
5.2.3	Seguimento de um único objecto . . . . .	59
5.2.4	Seguimento de múltiplos objectos . . . . .	62
5.3	Seguimento de objectos usando dados do LRF . . . . .	64
5.3.1	Pressupostos para os filtros implementados . . . . .	65
5.3.2	Filtro de Kalman . . . . .	66
5.3.3	Filtro de Kalman extendido . . . . .	66
5.3.4	Filtro de Partículas . . . . .	67
5.3.5	Filtro PDAF . . . . .	69
5.3.6	Consistência . . . . .	72

---

5.4	Cooperação sensorial . . . . .	74
<b>6</b>	<b>Classificador-GMM usando dados laser</b>	<b>77</b>
6.1	Função densidade probabilística . . . . .	78
6.2	Modelos de misturas Gaussianas (GMM) . . . . .	78
6.3	Máxima verosimilhança (ML) . . . . .	80
6.4	Breve descrição da regra de Bayes . . . . .	81
6.5	Algoritmo EM . . . . .	82
6.6	Classificador GMM . . . . .	85
6.7	Combinação probabilística de classificadores . . . . .	88
<b>7</b>	<b>Sistema MTDTS e resultados</b>	<b>91</b>
7.1	Arquitectura . . . . .	91
7.2	Segmentação e extracção de características . . . . .	93
7.3	Seguimento e associação de dados . . . . .	99
7.4	Análise de resultados do classificador GMM . . . . .	103
7.5	Combinação entre classificadores . . . . .	108
<b>8</b>	<b>Conclusões e trabalho futuro</b>	<b>115</b>
8.1	Conclusões . . . . .	115
8.2	Trabalho Futuro . . . . .	117
<b>A</b>	<b>Matriz de confusão</b>	<b>119</b>
A.1	Matriz de confusão para o classificador GMM-ML . . . . .	119
A.2	Matriz de confusão para o classificador GMM-MAP . . . . .	121
	<b>Bibliografia</b>	<b>123</b>



# Lista de Figuras

1.1	Diagrama de blocos da arquitectura do sistema de percepção proposto . . .	5
2.1	Princípio de funcionamento de um LRF com tecnologia TOF . . . . .	12
3.1	Representação Cartesiana de um varrimento de dados laser hipotético e algumas das variáveis envolvidas . . . . .	23
3.2	Representação geométrica comum a alguns métodos do tipo SBDE . . . . .	26
3.3	Representação geométrica do método tipo SBDE proposto por (Borges & Aldon, 2004) . . . . .	27
3.4	Segmentação de dados laser num cenário interior usando um método tipo SBKF. (a) Exemplo experimental com onze segmentos detectados após o processo de segmentação; (b) Valores quadráticos da inovação normalizada (NIS) para cada ponto-laser estimado em (a). . . . .	29
3.5	Modelo geométrico que define o método SEGMENTS . . . . .	31
3.6	Exemplo de segmentação de dados-laser num ambiente interior usando o método SEGMENTS . . . . .	32
3.7	Relação entre os pontos-laser numa superfície plana utilizada para o método ADA1 . . . . .	32
3.8	Exemplo de segmentação de dados-laser num ambiente interior usando o método ADA1 . . . . .	33
3.9	Relação entre os pontos-laser numa superfície plana para o método ADA2 . . . . .	35
3.10	Exemplo de segmentação de dados-laser num ambiente interior usando o método ADA2 . . . . .	36

4.1	Exemplo prático da utilização de primitivas geométricas para representar objectos segmentados em dados laser. (a) Sete segmentos detectados no processo de segmentação; (b) Primitivas geométricas extraídas de cada segmento de pontos. . . . .	39
4.2	Modelo geométrico utilizado nos métodos GUI1 e GUI2. . . . .	44
4.3	Relação geométrica dos pontos laser numa dada superfície cilíndrica . . . .	47
4.4	Exemplo de extracção de círculo usando um algoritmo baseado em KF. (a) Círculos extraídos ao longo do processo de estimação; (b) Círculo estimado ao final do processo de estimação usando um KF. . . . .	49
5.1	Regiões de validação elipsoidais utilizadas na associação de dados no seguimento de um único objecto num plano Cartesiano . . . . .	59
5.2	Ilustração gráfica das regiões de validação rectangulares utilizadas na associação de dados no seguimento de um único objecto num plano Cartesiano .	60
5.3	Distribuição Chi-quadrado definida em função da dimensão da variável observada . . . . .	60
5.4	Região de validação em torno do valor estimado. (a) RV com apenas uma medida válida; (b) RV com múltiplas medidas válidas. . . . .	61
5.5	Região de validação para o caso de dois objectos na mesma vizinhança . . .	63
6.1	Exemplo de um modelo multi-modal, com um conjunto de componentes Gaussianas, para uma característica de uma das classes de interesse. . . . .	79
6.2	Representação no plano de um conjunto de pontos que definem um objecto e as variáveis utilizadas para calcular as componentes do vector característico. 87	
7.1	Diagrama de blocos representando o sistema MTDTS . . . . .	93
7.2	Exemplo prático da utilização de círculos na representação geométrica de pontos laser. (a) Extracção de círculos nos pontos que definem um segmento; (b) Segmento com muitos pontos; (c) Segmento com três pontos; (d) Segmento de apenas dois pontos. . . . .	97

---

7.3	Exemplo prático da utilização de elipses para representar objectos segmentados em dados laser. (a) Elipses extraídas num conjunto de pontos laser; (b) Elipses com eixo-principal paralelo ao eixo vertical (ordenadas) do plano laser; (c) Elipses extraídas nos pontos que representam o veículo da <i>Yamaha</i> .	98
7.4	Diagrama sequencial da informação nos subsistemas que compõem o sistema MTDTS. . . . .	102
7.5	Curvas ROC do classificador GMM para a classe1: peões. O número de componentes Gaussianas utilizadas em cada modelo é representado por <b>M</b> .	105
7.6	Curvas ROC do classificador GMM para a classe2: carros/veículos. . . . .	106
7.7	Curvas ROC do classificador GMM para a classe3: árvores/postes. . . . .	107
7.8	Detecção e classificação de objectos usando o classificador <i>AdaBoost</i> . As linhas verticais definem as ROI no plano imagem em função da informação do módulo de cooperação sensorial com o laser. . . . .	111



# Lista de Tabelas

2.1	Características técnicas do LMS200 . . . . .	14
7.1	Resultado experimental para os métodos SBDE . . . . .	94
7.2	Matriz de confusão, onde $f(i)$ é a resposta do classificador . . . . .	104
7.3	Dados utilizados no classificador GMM para as diferentes classes e para a construção das curvas ROC . . . . .	104
7.4	Comportamento dos classificadores GMM-ML e GMM-MAP num caso de falha na associação entre objectos de classes distintas. . . . .	109
7.5	Taxa de classificação (TVP) e número de falsos positivos (FP) . . . . .	111
A.1	Matriz de confusão usando regra ML para a Classe-1: peões . . . . .	119
A.2	Matriz de confusão usando regra ML para a Classe-2: carros/veículos . . . . .	120
A.3	Matriz de confusão usando regra ML para a Classe-3: árvores/postes . . . . .	120
A.4	Matriz de confusão usando regra ML para a Classe-1: peões . . . . .	120
A.5	Matriz de confusão usando regra ML para a Classe-2: carros/veículos . . . . .	120
A.6	Matriz de confusão usando regra ML para a Classe-3: árvores/postes . . . . .	121
A.7	Matriz de confusão usando regra MAP para a Classe-1: peões . . . . .	121
A.8	Matriz de confusão usando regra MAP para a Classe-2: carros/veículos . . . . .	121
A.9	Matriz de confusão usando regra MAP para a Classe-3: árvores/postes . . . . .	121
A.10	Matriz de confusão usando regra MAP para a Classe-1: peões . . . . .	122
A.11	Matriz de confusão usando regra MAP para a Classe-2: carros/veículos . . . . .	122
A.12	Matriz de confusão usando regra MAP para a Classe-3: árvores/postes . . . . .	122





# Lista de Acrónimos e Glossário

CLUTTER . . . . .	Ambiente com elevada densidade de objectos-alvo sob seguimento
EKF . . . . .	Extended Kalman Filter
EM . . . . .	Expectation Maximization
GMM . . . . .	Gaussian Mixture Models
HMM . . . . .	Hidden Markov Models
IEKF . . . . .	Iterated EKF
IEPF . . . . .	Iterative End Point Fit
iid . . . . .	Independente e identicamente distribuído
ITS . . . . .	Intelligent Transportation System
KF . . . . .	Kalman Filter
LRF . . . . .	Laser Range Finder
LS . . . . .	Least Squares
MAP . . . . .	Maximum A Posteriori
ML . . . . .	Maximum Likelihood
MTDTS . . . . .	Multiple Target Detection and Tracking System
MV . . . . .	Múltiplo Votos, Classificador
NIS . . . . .	Normalized Innovation Squared
NN . . . . .	Nearest Neighbour
PDAF . . . . .	Probabilistic Data Association Filter

PDF .....	Probability Density Function
PF .....	Particle Filter
ROI .....	Region Of Interest
RV .....	Região de Validação
SBDE .....	Segmentação Baseada na Distância Euclidiana
SBKF .....	Segmentação Baseada em Filtros de Kalman
SLAM .....	Simultaneous Localization and Map Building
SVM .....	Support Vector Machine
TOF .....	Time Of Flight
TRACKER .....	Filtro estocástico associado a um objecto sob seguimento
UKF .....	Unscented KF
VPE .....	Estimador da Pose do Veículo

# Capítulo 1

## Introdução

### Conteúdo

---

1.1	Motivação . . . . .	2
1.2	Enquadramento do trabalho . . . . .	5
1.3	Objectivos da dissertação . . . . .	6
1.4	Trabalho realizado e contribuições . . . . .	8
1.5	Organização da tese . . . . .	10

---

**E**STE capítulo apresenta, de forma sucinta, as motivações, os principais objectivos, o desenvolvimento, as implementações experimentais, e as contribuições do trabalho realizado ao longo do período que constituiu este projecto de Mestrado em Engenharia Electrotécnica e de Computadores, área de especialização em Automação e Robótica. As primeiras secções expõem a motivação deste trabalho, uma introdução aos conceitos e métodos estudados e o seu respectivo enquadramento no contexto científico actual. As secções seguintes apresentam os objectivos e as contribuições desta dissertação. Finalmente a última secção apresenta a organização da tese.

## 1.1 Motivação

No âmbito europeu há um conjunto significativo de projectos que propõem novas formas de transporte urbano que se pretendem ser economicamente sustentáveis, que assegurem mais qualidade na mobilidade das pessoas, que não agridam o meio-ambiente, e principalmente que ofereçam maior segurança para a população. Estes projectos estão agrupados no chamado *NetMobil cluster* (Parent, 2005). Neste contexto, o principal interesse e motivação desta dissertação está nas tecnologias e inovações relacionadas com os sistemas de transporte inteligentes (ITS) e, mais especificamente, aos sistemas do tipo *Cybercars* (veículos urbanos de transporte inteligente). Essas tecnologias têm por objectivo dar mais eficiência aos veículos autónomos e garantir maior segurança. Pode-se citar alguns projectos de interesse no âmbito dos sistemas ITS e que têm relação com os objectivos desta dissertação: projecto ULTra (ULTra, 2003), projecto ARGO (ARGO, 1999), os projectos *Cybercars*, o projecto CityMobil (CityMobil, 2006), entre outros.

A maioria dos sistemas existentes nos robôs e veículos autónomos possuem uma arquitectura com alguns módulos que são comuns: navegação, planeamento de trajectória, controlo, comunicação, localização, detecção de obstáculos, alerta de colisão, classificação de objectos (ou obstáculos), entre outros; sendo que alguns destes módulos citados são comumente integrados num subsistema de percepção, ou seja, sistema de detecção e seguimento de múltiplos objectos/alvos (MTDTS). Os trabalhos realizados no âmbito dos sistemas de percepção, para aplicações em robótica móvel (Marques et al., 2006), e mais especificamente para o caso dos veículos inteligentes, constituem uma área com destaque nos últimos anos e que já está a ser aplicada, a um nível crescente, nos automóveis e nos veículos inteligentes nalgumas situações em ambientes reais (Bertozzi et al., 2004), (Zhang, Xie, & Adams, 2003), (Dietmayer, Sparbert, & Streller, 2001), (Wijesoma, Kodagoda, & Balasuriya, 2002). Um sistema MTDTS possui módulos cruciais nos veículos autónomos, principalmente por razões de segurança. Isto porque após as tarefas de detecção, seguimento e classificação dos objectos no ambiente que circunda o veículo, é possível a tomada de decisões de controlo: contornar obstáculos, evitar colisões, paragem, redefinição de

trajectória, etc. Desta forma, é imperativo a concepção e desenvolvimento de métodos robustos e altamente fiáveis, que constituem um sistema MTDTS, de modo que se cumpram os requisitos de controlo e segurança dos veículos autónomos.

Um dos temas principais desta dissertação é o da classificação de objectos, para o qual existem um conjunto diversificado de métodos disponíveis. Como exemplo podem-se referir os métodos baseados em múltiplas hipóteses (Streller, Dietmayer, & Sparbert, 2001), em funções de votos (Mendes, Bento, & Nunes, 2004), em classificadores do tipo *AdaBoost* (Viola & Jones, 2001), usando máquina de vector de suporte (SVM), em métodos usando mistura de modelos Gaussiano (GMM) (Premebida & Nunes, 2006) e modelos escondidos de Markov (HMM) (Wolf, Sukhatme, Fox, & Burgard, 2005), entre outros (Duda., Hart, & Stork, 2001). Os classificadores tratam de dados estruturados, pós processados, cuja representação passa por um vector de características, extraídas dos dados sensoriais que, em geral, são provenientes de câmaras, *Laser Range Finder* (LRF), radares, sonares, entre outros.

Uma linha de investigação, em evidência já há alguns anos, consiste na cooperação e fusão sensorial de dados de sensores instalados em veículos ou robôs móveis, com a finalidade de dar maior robustez ao sistema de percepção. São inúmeras as aplicações neste domínio: navegação e construção simultânea de mapas (SLAM) (Gamini, Newman, Clark, Durrant-Whyte, & Csorba, 2001), detecção de faixas nas estradas (Wijesoma et al., 2002), detecção de pessoas (Zhao & Shibasaki, 2005), navegação em terrenos desfavoráveis (e.g. o projecto *DARPA Grand Challenge*), sistemas de mobilidade para pessoas, entre outros (Leonard & Durrant-Whyte, 1991). Recentemente, e dependendo do contexto da aplicação, alguns sistemas tratam, além dos dados vindos dos sensores embarcados, a informação sensorial proveniente do ambiente envolvente (*surrounding smart infrastructure*) e tiram partido desta possibilidade como uma mais valia a acrescentar na cooperação sensorial do sistema. No âmbito desta tese, a cooperação sensorial está restringida à utilização de um LRF e uma câmara montados num suporte comum e embarcados num veículo autónomo.

As técnicas e métodos utilizados neste projecto têm impacto em muitas aplicações

comuns às áreas tecnológicas dos sistemas ITS, nomeadamente nos sistemas anti-colisão e detecção de obstáculos. Em termos mais específicos, o sistema MTDTS aqui proposto tem os seguintes módulos principais (Figura 1.1): a) Aquisição de dados e pré-filtragem b) Segmentação e extracção de características c) Detecção e seguimentos de objectos d) Associação e cooperação sensorial e) Classificação e combinação de classificadores.

Aquisição e pré-filtragem são as etapas primárias do sistema proposto. Têm como principal função enviar os dados já filtrados e tratados, vindos dos diferentes sensores instalados no veículo, e eventualmente da infra-estrutura envolvente, para os demais módulos.

Segmentação destina-se a identificar os segmentos nos dados de entrada e assim reduzir significativamente a informação útil a ser processada. O módulo de extracção de primitivas e de características têm como função enviar dados estruturados ao módulo de classificação. Essa informação também poderá ser utilizada nas tarefas de navegação, identificação de marcos de referência e construção de mapas.

A detecção e seguimento de objectos (ou obstáculos) é realizada por filtros estocásticos, nomeadamente o Filtro de Kalman (KF), com o objectivo de estimar e prever o comportamento dinâmico dos objectos identificados.

O módulo de associação actua de modo integrado com o módulo de detecção e seguimento (associação do tipo “*tracker-observation*”). Este módulo também deverá processar a informação vinda do módulo de segmentação e extracção de características (associação do tipo “*measurement-observation*”), de maneira a juntar (se necessário) segmentos que na verdade pertencem a um único objecto.

O módulo de classificação terá um papel fundamental na robustez e desempenho do sistema. Um classificador GMM (Premebida & Nunes, 2006) com regras de decisão do tipo *Maximum Likelihood* (ML) e *Maximum A Posteriori* (MAP) foi projectado para trabalhar com os dados do laser. Para aumentar a fiabilidade na classificação dos objectos nas áreas próximas ao veículo, propõe-se um módulo de classificação cooperativo para ser utilizado com outros classificadores: classificador múltiplos votos (MV) (Mendes et

al., 2004) (usando dados laser), e classificador *AdaBoost* (Monteiro, Peixoto, & Nunes, 2006) (usando dados de uma câmara). Por fim é utilizada uma regra Bayesiana de soma para combinar os três classificadores com o objectivo de se obter um resultado mais fiável (Monteiro, Premebida, Peixoto, & Nunes, 2006).

Os dados tratados pelo sistema MTDTS servem como informação útil a outros sub-sistemas, tais como: sistema de anti-colisão, módulo SLAM, módulo de estimação da posição e orientação do veículo (navegação), correcção de trajectória, etc. Enfim, a motivação deste trabalho prende-se com o facto do sistema aqui proposto poder ser utilizado, com as devidas adaptações, em múltiplos contextos de robótica móvel e veículos inteligentes.

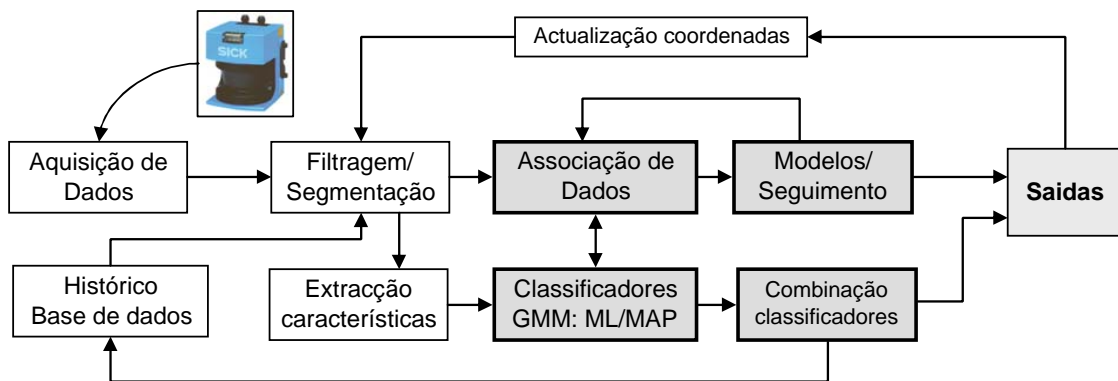


Figura 1.1: Diagrama de blocos da arquitectura do sistema de percepção proposto

## 1.2 Enquadramento do trabalho

Esta dissertação enquadra-se no contexto dos trabalhos que estão a ser desenvolvidos no Instituto de Sistemas e Robótica (ISR-UC), designadamente em projectos de investigação relacionados com sistemas de percepção, navegação, detecção, e classificação de objectos em tarefas de robótica móvel e veículos autónomos, de que são exemplo os projectos internacionais *Cybercars* ([www.cybercars.org](http://www.cybercars.org)) e *CyberC3*, e o *MTDTS04* financiado pela Fundação para a Ciência e Tecnologia. De um modo mais específico, o projecto *MTDTS04*



está relacionado mais explicitamente com as tarefas de detecção, seguimento e classificação de objectos baseado em múltiplos sensores. Dentro deste contexto e considerando alguns dos trabalhos já iniciados no ISR-UC (Mendes, 2004), esta dissertação dá continuidade ao projecto MTDTS04.

Como parte integrante dos projectos já iniciados outrora, os resultados experimentais apresentados neste trabalho utilizam as plataformas já existentes, nomeadamente o robot móvel *Robucar* e o veículo *CyberCab* da *Yamaha*. O sensor laser utilizado é um LMS200 da *Sick* e a câmara digital é uma *Unibrain (Fire-i)*.

### 1.3 Objectivos da dissertação

O objectivo principal desta dissertação passa por desenvolver uma arquitectura flexível e multi-sensorial, nomeadamente utilizando sensores do tipo LRF e câmaras, constituída por um sistema de detecção, seguimento e classificação de múltiplos objectos. Os objectos em estudo estão limitados a três classes de interesse: peões (Classe 1), carros (Classe 2), postes e troncos de árvores (Classe 3). O foco do trabalho possui três componentes principais:

- Módulos de pré-processamento, segmentação e extracção de características: efectua-se a filtragem, a segmentação (definição de segmentos/cluster), e finalmente é realizada a extracção de características de interesse para se compor o vector de características;
- Módulo de classificação de objectos: com base nas características observadas (extraídas) de cada segmento, e nos modelos GMM das categorias de interesse, o classificador GMM, baseado numa regra de decisão ML ou MAP, dá como resultado a classe correspondente ao objecto em análise;
- Combinação de classificadores: efectua combinação dos classificadores GMM, MV e *AdaBoost* de maneira a obter um resultado de classificação final mais robusto e

fiável.

Seguindo o enquadramento do sistema MTDTS apresentado e, para se alcançar os objectivos definidos, outras actividades foram desenvolvidas:

- **Cooperação sensorial:** uma relação de transformação entre o referencial do sensor laser e o da câmara é obtida de maneira que a informação correspondente aos objectos detectados no espaço do laser seja transformada, utilizando os parâmetros de calibração da câmara, para o espaço do plano imagem; essa informação tem por objectivo definir regiões de interesse (ROI) no plano imagem e facilitar o processo de segmentação e detecção dos objectos;
- **Modelos GMM:** cada categoria de objecto que se quer classificar é modelada, com base nos vectores de características extraídos na fase de treino, com um modelo GMM cujos parâmetros são estimados por um algoritmo *Expectation Maximization* (EM);
- **Modelos dinâmicos:** cada objecto é caracterizado por um modelo linear que representa aproximadamente a sua dinâmica. As observações também possuem modelos próprios que podem ser lineares ou não;
- **Técnicas de seguimento:** as técnicas implementadas referem-se a filtros estocásticos independentes: KF, *Extended Kalman Filter* (EKF) e *Probabilistic Data Association Filter* (PDAF);
- **Técnicas de associação de dados:** técnicas de construção de regiões de validação, assumindo modelos Gaussianos, rectangulares ou elipsoidais, em torno da variável estimada, cujas regras de decisão utilizam limiares definidos em função de tabelas de distribuição Chi-quadrado;
- **Regras de decisão Bayesianas:** utilizadas tanto no classificador GMM como na combinação dos classificadores. No caso GMM as regras referem-se à ML e à MAP

calculadas para cada vector-característico, e no caso da combinação de classificadores, trata-se do valor máximo da soma das probabilidades *a posteriori* de cada classificador.

As aplicações e resultados experimentais circunscreveram-se a ambientes de navegação exteriores semi-estruturados (e.g. cenários *cybercars*) utilizando-se plataformas já existentes no ISR-UC: o robot móvel *Robucar* e veículo eléctrico da *Yamaha*.

Num âmbito mais geral, este trabalho tem como objectivo dar continuidade ao projecto MTDTS04, este por sua vez está relacionado com aplicações em sistemas ITS, nomeadamente em cenários *Cybercars* (Cybercars, 2001). Dar continuidade refere-se ao estudo aprofundado do que já foi desenvolvido, de modo a propor possíveis melhorias e acrescentar novos módulos à arquitectura para aumentar, no global, a robustez e fiabilidade de todo o sistema. As principais contribuições deste trabalho foram: o classificador GMM, o módulo de cooperação sensorial entre laser e câmara e o módulo de combinação de classificadores.

## 1.4 Trabalho realizado e contribuições

Numa sequência cronológica distribuída ao longo de vinte e quatro meses contínuos, o trabalho realizado, e apresentado nesta dissertação, pode ser sumariado nas seguintes actividades principais:

- Implementação de vários métodos de segmentação dos dados de um LRF, analisando os pontos fracos e fortes de cada método e comparando-os com base em critérios de complexidade de implementação, esforço computacional e adaptabilidade a dados adquiridos em diferentes ambientes (Premebida & Nunes, 2005);
- Implementação de alguns métodos de detecção e extracção de primitivas geométricas, nomeadamente rectas, círculos e elipses, com base nos dados laser. Esses métodos foram utilizados com os dados no espaço Cartesiano, sendo que alguns são formulados

num contexto matemático geral e outros são específicos para aplicações usando LRF em situações de robótica móvel e veículos autónomos (Premebida & Nunes, 2005), (Premebida, 2005a);

- Análise e experimentação com filtros estocásticos em tarefas de seguimento (predição e estimação) de objectos detectados pelo LRF. Além dos filtros clássicos (KF, EKF), foi também implementado um filtro PDAF e uma versão “básica” de um Filtro de Partículas (PF) (Premebida, 2005b);
- Classificação dos objectos detectados utilizando modelos GMM e com regra de decisão ML e MAP (Premebida & Nunes, 2006);
- Cooperação sensorial (LRF e câmara) e combinação dos classificadores MV (Mendes et al., 2004), *AdaBoost* (Monteiro, Peixoto, & Nunes, 2006) e GMM (Premebida & Nunes, 2006), utilizando uma regra de soma do tipo Bayesiana (Monteiro, Premebida, et al., 2006).

De um modo geral, todos os algoritmos foram implementados utilizando-se o software Matlab<sup>®</sup>, sendo que alguns algoritmos foram implementados em linguagem C para os casos de aplicações experimentais. A análise experimental e a validação dos métodos propostos tiveram em conta os requisitos funcionais do sistema MTDTS proposto e os objectivos do presente projecto. Adicional e paralelamente às actividades mencionadas acima, outras actividades foram desenvolvidas no contexto desta dissertação, cujos resultados e desenvolvimentos estão referenciadas nos relatórios técnicos (Premebida, 2005a)-(Premebida, 2005b).

Uma das principais contribuições deste trabalho foi o desenvolvimento do módulo de classificação de objectos cuja formalização matemática permite uma grande flexibilidade em aplicações práticas, sejam elas utilizando um LRF ou a conjugação de vários sensores. Adicionalmente, a cooperação sensorial e a combinação de vários classificadores (projectados em espaços sensoriais distintos) torna o sistema mais robusto e permite a utilização de classificadores desenvolvidos noutros projectos. No que se refere à detecção, seguimento e

associação de dados no sistema MTDTS, as principais contribuições desta dissertação são:

- Os vários métodos de segmentação e extracção de características utilizados no contexto das aplicações em robótica móvel e em veículos autónomos utilizando um sensor LRF;
- A implementação de vários modelos dinâmicos lineares com filtros estocásticos para detecção de peões, carros, e outros objectos: paredes, postes e árvores;
- A aplicação das técnicas “clássicas” de associação de dados, i.e. a utilização de regiões de validação do tipo rectangular ou elipsoidal num formalismo Gaussiano e com base em limiares definidos em função de uma distribuição Chi-quadrado;

## 1.5 Organização da tese

A dissertação está dividida em três partes essenciais. Os dois primeiros capítulos (**Introdução** e **Aplicações do sensor LRF em sistemas MTDTS**) apresentam a introdução, as metodologias e o enquadramento das situações tratadas. O capítulo 3 (**Segmentação**), capítulo 4 (**Extracção de primitivas geométricas**), e capítulo 5 (**Seguimento de objectos**) apresentam os métodos, as técnicas e as abordagens analisadas no contexto do trabalho. O capítulo 6 (**Classificador-GMM usando dados laser**) trata do projecto e das características do classificador GMM. Os resultados experimentais alcançados estão no capítulo 7 (**Sistema MTDTS e resultados**), e finalmente o capítulo 8 (**Conclusões e trabalho futuro**) apresenta as conclusões do trabalho desenvolvido e os objectivos futuros.

## Capítulo 2

# Aplicações do sensor LRF em sistemas MTDTS

### Conteúdo

---

2.1	Características do sensor LRF . . . . .	13
2.2	Objectos sob seguimento usando LRF . . . . .	15
2.3	Segmentação e extracção de características . . . . .	17
2.4	Seguimento e associação de dados . . . . .	18
2.5	Classificador-GMM e combinação de classificadores . . . . .	18
2.6	Sistema MTDTS . . . . .	19

---

UM LRF é um sensor optoelectrónico de medição de distâncias ou profundidade. As duas principais tecnologias utilizadas nos sensores lasers de medição de distâncias são: tempo de voo (TOF) e deslocamento de fase. Num sensor do tipo TOF, cujo princípio de funcionamento está ilustrado na Figura 2.1, um pulso laser é emitido e o intervalo de tempo  $\Delta t$  passado até que este feixe retorne ao receptor do sensor é medido (basicamente o mesmo princípio utilizado pelos sensores de sonar). A medição da distância  $r$  entre o sensor e o objecto é então calculada com base na seguinte fórmula:

$$r = \frac{c_l \Delta t}{2} \quad (2.1)$$

sendo  $c_l$  a velocidade da luz (aproximadamente  $3 \times 10^8 m/s$ ). Para efectuar o cálculo (2.1), são utilizados sistemas electrónicos de extrema precisão. Com pulsos lasers rápidos e curtos é possível utilizar níveis de potência mais elevados, aumentando-se então a capacidade de detecção a distâncias maiores, mantendo o consumo energético relativamente baixo e os níveis de radiação toleráveis aos olhos humanos.

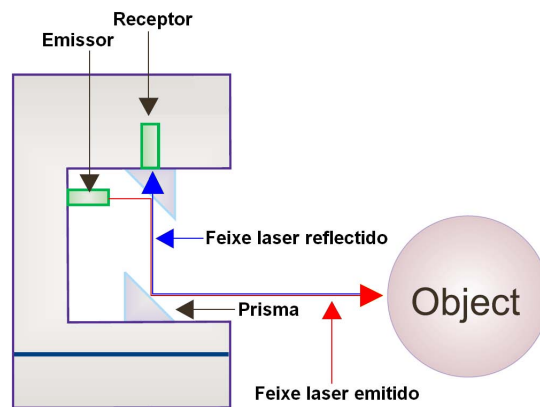


Figura 2.1: Princípio de funcionamento de um LRF com tecnologia TOF

Os sensores do tipo LRF utilizam um sistema mecânico de rotação com um prisma (espelho) de maneira a reflectir uma sequência de pulsos laser, formando assim um plano de medição, i.e. um conjunto de pontos que definem um varrimento. A maioria dos sensores comerciais medem a distância apenas numa direcção, mas há sensores disponíveis no mercado que efectuem o varrimento dos feixes lasers em mais de uma direcção, gerando-se assim imagens do tipo 3D. Outra opção adoptada é a utilização de um dispositivo externo, instalado num LRF-2D convencional, de maneira a fazer girar o sensor no sentido vertical obtendo-se deste modo informação tridimensional.

O sensor utilizado neste trabalho é do tipo 2D LRF, quer dizer, um sensor laser de medição de distância por meio de um varrimento, formando-se assim um plano bidimensional de medição.

## 2.1 Características do sensor LRF

Os sensores do tipo LRF efectuam um conjunto de medidas que correspondem à intersecção dos feixes laser com os objectos no campo de alcance do sensor. Este conjunto de pontos (tratados como uma sequência de dados com uma resolução angular constante) referem-se ao valor da distância, ou profundidade, entre o sensor e os respectivos objectos que foram intersectados. O modelo usado neste trabalho é um LMS200 da *Sick*, cujo funcionamento se baseia na técnica TOF. O feixe laser é deflectido por meio de um espelho (prisma) que está a girar a 4500 rpm (75 rps), resultando assim num “varrimento”. Cada feixe laser é emitido com um dado intervalo de tempo e uma determinada resolução angular. Caso o sinal não retorne num dado tempo limite, o sensor considera a medição deste feixe laser como sendo máxima. Neste sensor a resolução angular  $\Delta\theta$  pode ser seleccionada como  $1^\circ$ ,  $0.5^\circ$ , ou  $0.25^\circ$ , sendo que as resoluções de  $0.5^\circ$  e  $0.25^\circ$  são conseguidas por meio de entrelaçamento durante o varrimento, ou seja, um *frame* completo de dados é obtido através de dois ou quatro varrimentos entrelaçados, respectivamente. A taxa de transferência dos dados pode ser de 9.6, 19.2, 38.4, ou 500 Kbaud, podendo trabalhar com os protocolos RS232 ou RS422. As principais características técnicas do LMS200<sup>1</sup> estão sumariadas na Tabela 2.1.

### Características positivas do LRF

Os principais pontos positivos do sensor LRF são:

- Velocidade de transmissão: o sensor LMS200 pode transmitir até 500 kb/s, o que significa uma elevada taxa útil de transmissão (aproximadamente de 36.5 Hz), permitindo grande fiabilidade em aplicações de “tempo real”. Uma das principais limitações actualmente está no sistema mecânico de rotação cuja velocidade e precisão impedem que o sensor tenha taxas de medição mais elevadas;
- Precisão: Para a maioria das aplicações em robótica móvel, a resolução do sensor

---

<sup>1</sup>Valores tomados de (Ye & Borenstein, 2002), com medições de até 4m



Especificações Técnicas	
Distância Máxima	8m/80m
Resolução	10mm/100mm
Desvio padrão <sup>1</sup>	$0.8 \leq 9.1mm$
Erro sistemático <sup>1</sup>	$\pm 17mm$
Alcance angular	$100^\circ \leq 180^\circ$
Resolução angular	0.25°/0.5°/1°
Interfaces	RS232/RS422
Taxas de transferência	9.6/19.2/38.4/500 Kbaud

Tabela 2.1: Características técnicas do LMS200

LMS200 é mais do que suficiente, e.g. para distâncias até 8m têm-se um desvio padrão aproximado de 10mm;

- Resolução: a resolução angular do LRF LMS200 é alterável, podendo ser de 0.25°, 0.5° ou 1°, e com desvio padrão extremamente baixos;
- Percepção: os dados de distância  $(r, \theta)$  enviados pelo LRF quando intersectam um objecto são de fácil interpretação, de modo a facilitar as tarefas de percepção e tratamento dos dados.

### Limitações do LRF

Algumas das limitações do sensor LMS200 são as seguintes:

- Consumo energético: o LMS200 consome aproximadamente 17.5 Watt (24 Vdc), o que é muito em comparação com outros tipos de sensores como câmaras e sonares;
- Limitação na percepção sensorial: os sensores LRF do tipo 2D, tal como o LMS200, têm a limitação sensorial restringida a um plano. Embora essa situação possa ser colmatada com a utilização de LRF do tipo 3D (mais caros);

- Custo: o preço deste tipo de sensor ainda é elevado;
- Reflectividade: alguns materiais, tais como vidro, acrílico, entre outros, não são detectados pelo LRF devido às propriedades especular desses materiais.

## 2.2 Objectos sob seguimento usando LRF

O seguimento de objectos usando um LRF possui características peculiares, cujas vantagens são:

- Cada ponto-laser representa uma posição directa no plano polar, e indirectamente no plano Cartesiano, e é tão precisa quanto a resolução angular do sensor;
- Os algoritmos de segmentação são relativamente fáceis de implementar, apresentam bons resultados em situações práticas, e com baixo tempo de processamento;
- A quantidade de informação (o conjunto de pontos que definem um varrimento) enviada por um LRF 2D é relativamente pequena, de fácil tratamento e com taxa de transmissão elevada;
- As medições de um LRF são muito robustas, designadamente quanto a interferências externas tais como: luminosidade, nevoeiro pouco denso, variações de temperatura, humidade. Interferências essas que nalguns casos são extremamente prejudiciais no desempenho de alguns tipos de sensores;

No entanto, alguns aspectos menos positivos merecem atenção:

- A correspondência entre pontos detectados num instante corrente com os pontos medidos no instante anterior pode ser uma tarefa difícil, cuja complexidade aumenta consideravelmente quando o objecto e sensor estão em movimento;

- Objectos translúcidos, espelhados ou com superfície muito irregular (e.g. objectos com muitos orifícios) tendem a ser de difícil detecção a longas distâncias, agravando fortemente o problema de identificação e segmentação do objecto;
- Devido à quantidade de pontos que são reflectidos por um objecto, por vezes podem ocorrer ambiguidades na determinação da posição exacta do objecto, i.e. tomando-se a “nuvem” de pontos que representam um objecto, qual ponto característico utilizar?

Ao longo deste trabalho define-se como ponto característico dum segmento o centro de massa (ou centróide), calculado sobre o conjunto de pontos laser segmentados, de modo a representar a posição do objecto num plano Cartesiano com origem no referencial do sensor.

### **Objectos e laser em movimento**

Em aplicações práticas, é comum que o sensor laser esteja instalado num veículo ou robot móvel. Considerando o caso em que o sensor laser esteja em movimento (instalado num veículo por exemplo), os objectos detectados pelo sensor apresentam um movimento “aparente” em relação ao referencial de coordenadas do laser. Essa situação é ainda mais acentuada quando o objecto está em movimento independente do movimento do sensor. Um modo de se resolver, ou minimizar, esse efeito de “velocidade aparente” dos objectos é, através da pose estimada do veículo (ou robot), corrigir a inovação (no ciclo de predição do filtro), com os deslocamentos incrementais do carro no qual o laser está embarcado. Pode-se também implementar uma etapa de transformação de coordenadas, anterior ao processo de segmentação dos dados do laser, como uma solução mais precisa. Há ainda outra dificuldade relacionada com a detecção de objectos em movimento usando laser que é a mudança na visualização geométrica do objecto. A seguir, estão listadas situações importantes a ter em atenção durante o projecto de um sistema MTDTs com LRF:

- um mesmo objecto pode, em diferentes varrimentos, ser composto por mais de um segmento;

- um objecto pode ficar parcial ou totalmente oculto por outro objecto, ou obstáculo;
- um objecto, ao se mover, pode apresentar uma “brusca” mudança na sua aparência (contorno);
- em situações específicas, um objecto pode apresentar um contorno muito complexo e de difícil identificação e reconhecimento pelo laser. Exemplo disso são os arbustos e alguns tipos de veículos;
- alguns objectos apresentam baixa reflectividade na frequência do sinal do laser, e traz consequências indesejadas tais como a dificuldade na detecção e reconhecimento do objecto (este pode estar “invisível” ao laser).

### 2.3 Segmentação e extracção de características

As aplicações que utilizam métodos de segmentação e extracção de características com dados sensoriais estão presentes em diversas áreas científicas, como por exemplo: processamento de sinais, visão computacional, percepção sensorial, robótica móvel, entre outras. No contexto deste trabalho, há especial interesse nos métodos relacionados ao tratamento dos dados vindos de um sensor LRF para aplicações em robótica móvel e veículos autónomos. Os métodos de segmentação de dados usando LRF podem ser classificadas em dois grupos: métodos algébricos baseados na distância entre pontos (Lee, 2001),(Borges & Aldon, 2004),(Streller & Dietmayer, 2004),(Santos, Faria, Soares, Araujo, & Nunes, 2003) e os métodos baseados em filtros estocásticos (a maioria utilizam filtros de Kalman) cujos modelos diferem consoante a aplicação (ambiente interior ou exterior) e o contexto em que são aplicados (Roumeliotis & Bekey, 2000),(Adams, 2001),(Borges & Aldon, 2004),(Zhang et al., 2003). Os métodos de extracção de características geométricas apresentados nesta dissertação foram divididos consoante o tipo de primitiva geométrica: rectas (Nguyen, Martinelli, Tomatis, & Siegwart, 2005), (Borges & Aldon, 2004), (Xavier, Pacheco, Castro, Ruano, & Nunes, 2005), (Roumeliotis & Bekey, 2000), (Vandorpe, Brusel, & Xu, 1996), (Siadat, Kaske, Klausmann, Dufaut, & Husson, 1997), círculos (Amara

et al., 1996), (Gander, Golub, & Strebel, 1994), (Vandorpe et al., 1996), (Siadat et al., 1997) e elipses (Fitzgibbon, Pilu, & Fisher, 1999), (Gander et al., 1994).

## 2.4 Seguimento e associação de dados

Pode-se seguramente dizer que qualquer sistema que trate de eventos dinâmicos, tal como o MTDTS, precisa de algum módulo de seguimento. A tarefa de seguimento abrange, em geral, um conjunto de filtros e estimadores estocásticos, sendo os mais utilizados: KF, EKF, PDAF (Kirubarajan & Bar-Shalom, 2004), PF (Rekleitis, 2004). Para se descrever o comportamento dinâmico dos objectos detectados, alguns modelos estocásticos lineares com representação em espaço de estados, baseados em (Bar-Shalom & Li, 1995), foram utilizados. Indispensável para os algoritmos de seguimento, o módulo de associação de dados abrange as técnicas formuladas em função de uma região de validação (rectangular ou elíptica, sob uma abordagem Gaussiana) e das regras de associação: *Nearest Neighbour* (NN), máxima verosimilhança, entre outras, (Bar-Shalom & Fortmann, 1988), (Blackman, 1986).

No caso do sistema MTDTS discutido nesta dissertação, além dos módulos já citados, tem-se particular interesse na cooperação sensorial entre uma câmara e o laser, de que são exemplos outros trabalhos nesta área: (Neira, Tardós, Horn, & Schmidt, 1999), (Wijesoma et al., 2002), (Bertozzi et al., 2004).

## 2.5 Classificador-GMM e combinação de classificadores

A classificação de objectos, cuja aplicação é vasta, nomeadamente na visão (Duda. et al., 2001), é um dos módulos cruciais no MTDTS apresentado nesta dissertação, para o qual existem diferentes métodos. No âmbito de alguns projectos com sistemas MTDTS desenvolvidos no ISR-UC, pode-se citar dois exemplos de aplicação: um classificador de múltiplos votos (MV) com dados do LRF (Mendes et al., 2004), e um classificador *Ada-*

*Boost* (Monteiro, Peixoto, & Nunes, 2006) com dados de uma câmara a cores. O método GMM, com regra de decisão do tipo ML e MAP, foi implementado experimentalmente apenas com informação sensorial proveniente do sensor LRF (Premebida & Nunes, 2006), mas o método pode ser expandido para incluir outras modalidades sensoriais, e os objectos de interesse foram: peões, carros, e troncos de árvores/postes. De modo a aumentar a robustez do sistema, foi implementado um método de combinação de classificadores baseada na regra de Bayes (Kittler, Hatef, Duin, & Matas, 1998), que combina o resultado dos classificadores GMM, MV e *AdaBoost* por meio de uma regra de soma das probabilidades *a posteriori* de cada classificador.

## 2.6 Sistema MTDTS

Dentro do contexto em que será utilizado ao longo desta tese, o sistema MTDTS abrange quatro subsistemas principais, responsáveis por um conjunto de tarefas sucintamente descritas a seguir:

- Segmentação: realiza a pré-filtragem dos dados-laser e detecta os grupos de pontos (segmentos) que representam os objectos-alvo dentro do espaço sensorial;
- Extração de primitivas geométricas: extrai os parâmetros das primitivas geométricas e calcula o vector característico utilizado nas fases de seguimento e classificação;
- Seguimento de objectos: abrange os modelos cinemáticos, os filtros estocásticos, e o módulo de associação de dados;
- Classificação usando dados laser: utiliza um classificador GMM com regras de decisão do tipo ML e MAP;



## Capítulo 3

# Segmentação

### Conteúdo

---

3.1	Métodos baseados na distância Euclidiana . . . . .	24
3.2	Métodos baseados no filtro de Kalman . . . . .	27
3.3	Métodos baseados no filtro EKF . . . . .	29

---

**S**EGMENTAÇÃO é a primeira etapa, após a aquisição dos dados do LRF, a ser implementada no sistema MTDTS em consideração neste trabalho. No entanto, alguns autores optam por implementar técnicas clássicas de filtragem de dados antes de proceder à segmentação, e.g., filtragem por média, por redução, projecção, etc. Essas técnicas de filtragem são pouco relevantes nas aplicações com sensores que enviam poucos pontos (como é o caso do LMS200) e também são “discutíveis” para as aplicações em ambiente exterior nos quais, devido à natureza dos objectos a serem detectados, os dados laser que representam um objecto são constituídos às vezes por poucos pontos. Em alguns casos o processo de segmentação dos dados de um laser é interpretado como *clustering*, detecção de pontos-de-quebra ou detecção de pontos-de-ruptura. Para evitar ambiguidades, ao longo deste trabalho, as seguintes definições serão consideradas:

- Pontos-de-quebra: correspondem ao primeiro e ao último ponto laser que define um



único segmento, e o distingue dos demais pontos-laser num varrimento;

- Varrimento: refere-se a uma sequência de  $N_S$  pontos-laser recebidos num varrimento de  $180^\circ$  do sensor LRF.
- Segmentação: é o processo de tratamento dos dados laser a fim de distinguir (separar) os pontos-de-quebra presentes num varrimento, i.e. “descontinuidades” no varrimento que indicam muito provavelmente a presença de um objecto no campo de detecção do sensor. A saída desse processo são grupos de pontos, separados entre si, e que constituem o que designamos por segmentos.
- Segmento: é o conjunto de pontos num varrimento laser, próximos uns dos outros de acordo com critérios baseados na distância entre pontos (depende da técnica de segmentação utilizada) e que, devido a essa “proximidade”, provavelmente pertencem a um único objecto. Esse conjunto de pontos caracteriza a silhueta do objecto (correspondente à intersecção do feixe laser com uma das “faces” do objecto).

Considerando um varrimento completo como uma sequência ordenada de  $N_S$  pontos de medição, representado da seguinte forma  $\mathcal{S} = \{(r_l, \theta_l) | l = 1, \dots, N_S\}$ , onde  $(r_l, \theta_l)$  indica as coordenadas polares de cada  $l^{esimo}$  ponto laser que compõem aquele varrimento; um grupo de pontos que constituem um dado segmento  $S_j$ , onde  $S_j$  é um sub-conjunto de  $\mathcal{S}$ , será expresso como:

$$S_j = \{(r_n, \theta_n)\}, \quad n \in [l_{begin}, l_{end}], \quad j = 1, \dots, N_{seg} \quad (3.1)$$

onde  $l_{begin}$  e  $l_{end}$  representam o primeiro e o último ponto-de-quebra de cada segmento, e  $N_{seg}$  é o número total de segmentos. Um segmento pode ser definido em coordenadas Cartesianas, fazendo  $x_n = r_n \cos \theta_n$  e  $y_n = r_n \sin \theta_n$ .

Aplicando, como entrada, uma sequência completa de  $N_S$  pontos de um varrimento, o algoritmo de segmentação identifica, para cada  $j^{esimo}$  segmento (3.1) detectado, um vector

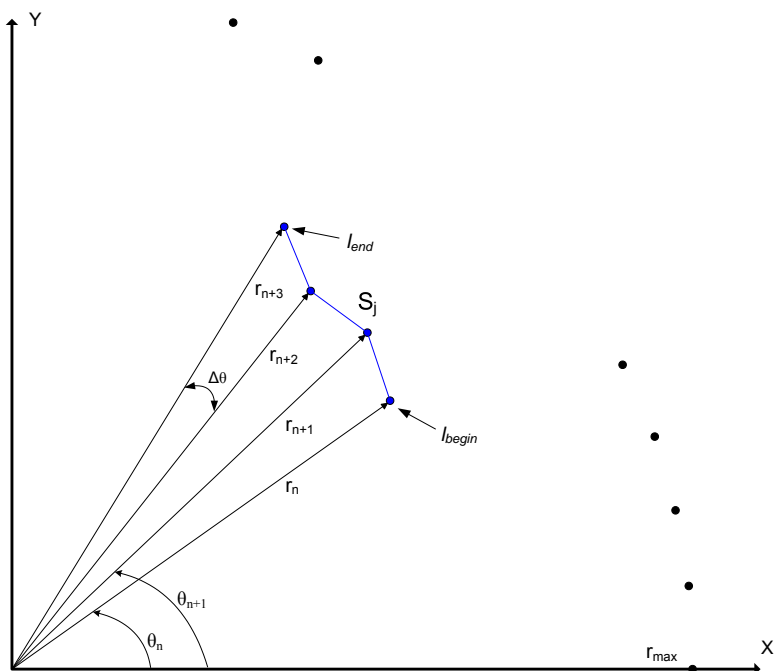


Figura 3.1: Representação Cartesiana de um varrimento de dados laser hipotético e algumas das variáveis envolvidas

de dados que inicia com o primeiro ponto-laser ( $l_{begin}$ ) e termina com o último ponto-laser ( $l_{end}$ ), definido no sentido anti-horário. A Figura 3.1 ilustra as principais variáveis utilizadas e que servem de referência no processo de segmentação dos dados vindos de um varrimento laser.

Quando, durante o varrimento, um determinado ponto-laser excede o alcance máximo do sensor LRF (devido ao facto de não haver objecto dentro do campo de alcance do laser ou por efeitos de dispersão), este ponto-laser é “interpretado” pelo sensor e o seu valor de medição é considerado máximo ( $r_{max}$ ). Durante o processo de segmentação todos os pontos processados que atingem o valor  $r_{max}$  são eliminados, pois não representam nenhum objecto. É importante filtrar os chamados “pontos esporádicos” que apareçam no conjunto de pontos que constituem um dado segmento. Estes pontos esporádicos devem ser eliminados pois não trazem informação útil sobre o segmento. No entanto, nalgumas

situações específicas o número de pontos esporádicos numa sequência de medições que definem um segmento significa que o objecto correspondente (definido pelo segmento) possui áreas (zonas) com forte efeito especular ou de dispersão. De maneira a simplificar e tornar mais clara a discussão dos vários métodos de segmentação apresentados neste trabalho, optou-se por classificar os métodos em duas categorias: **Segmentação Baseada na Distância Euclidiana** (SBDE) e **Segmentação Baseada em Filtros de Kalman** (SBKF). A primeira agrupa os métodos que utilizam a distância Euclidiana entre os pontos como condição de detecção dos pontos-de-quebra, e a categoria SBKF está relacionada com os métodos baseados em filtros de Kalman.

### 3.1 Métodos baseados na distância Euclidiana

Todos os métodos apresentados neste capítulo, e que serão analisados nas secções subsequentes, foram implementados, estudados e testados com dados laser adquiridos em ambientes interiores e exteriores. Os algoritmos da categoria SBDE têm uma forma geral comum, representada no **Algoritmo 1**. Estes métodos utilizam o valor da distância Euclidiana (3.2) entre pontos-laser consecutivos, e uma variável  $D_{thd}$  como limiar, para a condição de detecção dos pontos de quebra.

$$D(r_i, r_{i+1}) = \sqrt{r_i^2 + r_{i+1}^2 - 2r_i r_{i+1} \cos(\Delta\theta)} \quad (3.2)$$

No contexto dos processos de segmentação é comum utilizar uma expressão aproximada para a distância Euclidiana, dada por:

$$D(r_i, r_{i+1}) \approx |r_i - r_{i+1}| \quad (3.3)$$

O primeiro método da categoria SBDE discutido nesta secção é baseado na abordagem proposta por (Sparbert, Dietmayer, & Steller, 2001), cujo valor de limiar ( $D_{thd}$ ) é calculado de acordo com a seguinte expressão:

**Algoritmo 1** Segmentação baseada na distância Euclidiana**Input:** Grupo de pontos provenientes de um varrimento do LRF.**Output:** Conjunto de segmentos:  $S_j$   $j = 1, \dots, N_{seg}$ .

---

```

1:  $N_{seg}$ : número de segmentos detectados
2:  $N_S$ : número de pontos-laser no varrimento
3:  $j = 0$ 
4: for  $i = 1$  to  $N_S$  do
5:   Calcula distância Euclidiana:  $D(r_i, r_{i+1})$ 
6:   if  $D(r_i, r_{i+1}) > D_{thd}$  then
7:      $j++$ 
8:      $S_j \leftarrow TRUE$ : ponto de quebra detectado
9:   else
10:     $S_j \leftarrow FALSE$ 
11:   end if
12: end for

```

---

$$D_{thd} = C_0 + C_1 \cdot \min(r_i, r_{i+1}) \quad (3.4)$$

onde  $C_1 = \sqrt{2(1 - \cos(\Delta\theta))}$ , e  $C_0$  é um parâmetro constante usado para atenuar o efeito do ruído na medição. Inspirado na abordagem anterior, em (Santos et al., 2003) propõe-se um novo parâmetro ( $\beta$ ), com o objectivo de reduzir a dependência da segmentação com a distância entre o sensor LRF e o objecto detectado. O valor de limiar resultante (3.5) é dado por:

$$D_{thd} = C_0 + \frac{C_1 \cdot \min(r_i, r_{i+1})}{\cot(\beta) \cdot (\cos(\Delta\theta/2) - \sin(\Delta\theta/2))} \quad (3.5)$$

sendo  $C_0$  e  $C_1$  são calculados como anteriormente. A Figura 3.2 ilustra uma representação geométrica com as variáveis envolvidas em cada um dos dois métodos.

Um método de segmentação alternativo, apresentado em (Lee, 2001), tem como limiar

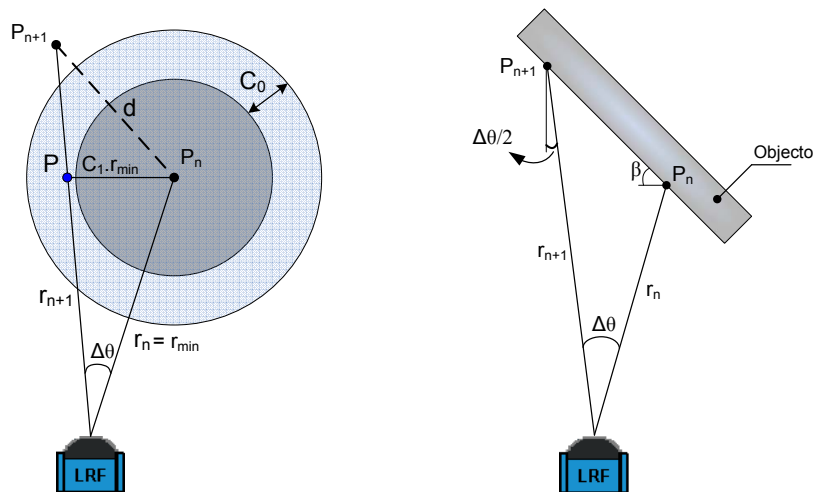


Figura 3.2: Representação geométrica comum a alguns métodos do tipo SBDE

a seguinte expressão:

$$D_{thd} = |r_i + r_{i+1}| \cdot C_e \quad (3.6)$$

onde  $C_e$  é uma constante obtida empiricamente. Outro método tipo SBDE aqui apresentado, proposto por (Borges & Aldon, 2004), tem a representação geométrica ilustrada na Figura 3.3. De acordo com esta representação, este método contempla os seguintes passos:

- Define-se uma linha “virtual” que passa pelo ponto-laser  $p_{n-1}$ ;
- A linha “virtual” faz um ângulo ( $\lambda$ ) com a direcção do varrimento ( $\theta_{n-1}$ );
- O ângulo  $\lambda$  é uma variável livre, definida com valor da ordem dos  $10^\circ$ ;
- A variável  $\Delta\theta$  é a resolução angular do sensor.

Neste método a distância limiar é calculada por:

$$D_{thd} = r_i \cdot \frac{\sin(\Delta\theta)}{\sin(\lambda - \Delta\theta)} + \sigma_r \quad (3.7)$$

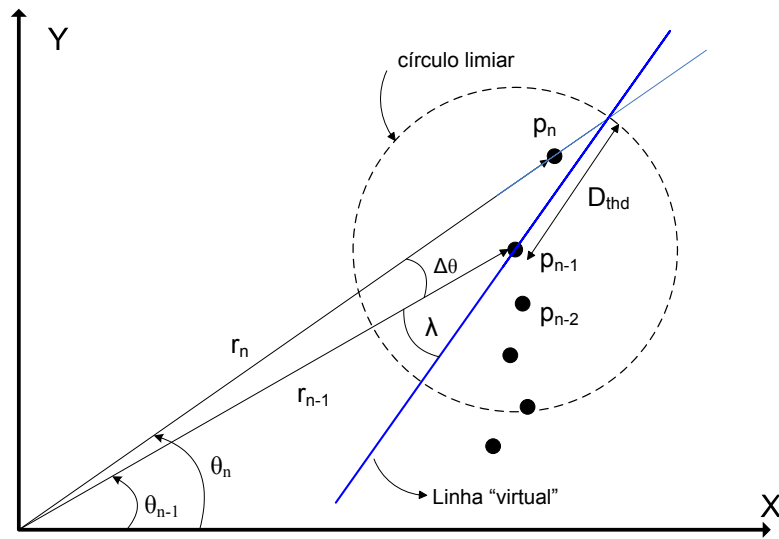


Figura 3.3: Representação geométrica do método tipo SBDE proposto por (Borges & Aldon, 2004)

Em (3.7) a variável  $\lambda$  é um parâmetro auxiliar de ajuste e  $\sigma_r$  é a variância residual que caracteriza o comportamento estocástico da sequência de pontos laser  $S_j$  e o respectivo erro associado a cada ponto-laser do processo.

### 3.2 Métodos baseados no filtro de Kalman

Nesta secção é apresentada uma técnica, utilizando KF, para a segmentação de objectos detectados pelo LRF. O sistema é caracterizado pelo modelo do processo, que define o comportamento estocástico dos pontos-laser durante a medição, e pelo modelo de medição, que são representados no domínio discreto pelas equações:

$$\begin{cases} \mathbf{x}(k+1) = \mathbf{f}(\mathbf{x}(k), \nu(k)) \\ \mathbf{z}(k) = \mathbf{h}(\mathbf{x}(k), w(k)) \end{cases} \quad (3.8)$$

A dinâmica do sistema é definida em função do estado  $\mathbf{x}(k)$  e das perturbações do

processo e das medições, representadas pelas sequências aleatórias  $w(k)$  e  $\nu(k)$  respectivamente.

No método apresentado em (Borges & Aldon, 2004), o comportamento dinâmico dos pontos-laser medidos  $r(k)$  é descrito por um modelo de segunda ordem, com velocidade angular constante:

$$r(k+1) = r(k) + \Delta\theta \cdot \frac{dr(k)}{d\theta} \quad (3.9)$$

cujas variáveis de estados são:

$$\mathbf{x}(k) = [r(k) \quad \frac{dr(k)}{d\theta}]^T \quad (3.10)$$

O modelo discreto em espaço de estados é representado por:

$$\begin{aligned} x_1(k+1) &= x_1(k) + \Delta\theta \cdot x_2(k) \\ x_2(k+1) &= x_2(k) \end{aligned} \quad (3.11)$$

Neste método, o sistema dinâmico estocástico que descreve o comportamento do processo é do tipo linear, com a seguinte representação:

$$\begin{aligned} \mathbf{x}(k+1) &= \mathbf{F}\mathbf{x}(k) + \nu(k) \\ \mathbf{z}(k+1) &= \mathbf{H}\mathbf{x}(k) + w(k) \end{aligned} \quad (3.12)$$

em que  $\mathbf{H} = [1 \ 0]$  e a matriz de transição de estados  $\mathbf{F}$  é dada por:

$$\mathbf{F} = \begin{bmatrix} 1 & \Delta\theta \\ 0 & 1 \end{bmatrix} \quad (3.13)$$

Com este método, a detecção de um ponto de quebra depende de uma região de validação (RV) do tipo Gaussiana e de um limiar calculado com base na dimensão da variável

observada e na distribuição Chi-quadrado. Ou seja, uma RV centrada no valor estimado da medida é construída e, caso o ponto-laser de medição no tempo corrente esteja fora desta região, então este ponto é detectado como ponto de quebra. A Figura 3.4(b) ilustra um exemplo dos valores da distância de *Mahalanobis* calculados para cada valor estimado ao longo de um processo de segmentação num ambiente interior (Figura 3.4(a)); caso esse valor ultrapasse o limiar, escolhido com base numa distribuição probabilística do tipo Chi-quadrado, então um ponto-de-quebra é detectado (um limiar de validação igual a 6.64 foi utilizado, correspondente a 99% de probabilidade de validação).

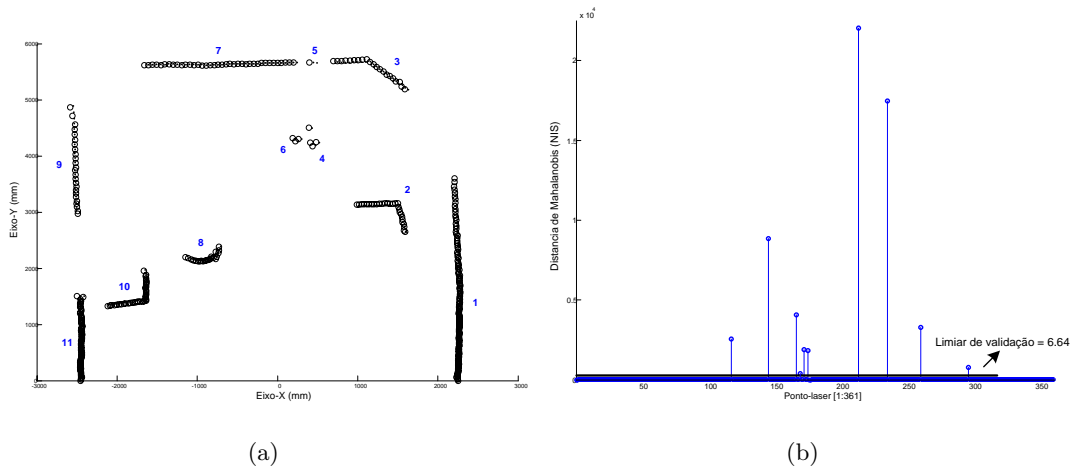


Figura 3.4: Segmentação de dados laser num cenário interior usando um método tipo SBKF. (a) Exemplo experimental com onze segmentos detectados após o processo de segmentação; (b) Valores quadráticos da inovação normalizada (NIS) para cada ponto-laser estimado em (a).

### 3.3 Métodos baseados no filtro EKF

Em sistemas em que a função  $f$  e/ou  $h$  (3.8) são não-lineares, deve ser utilizado um filtro estocástico que trate de processos não-lineares, sendo o EKF o mais conhecido. A seguir serão apresentados alguns exemplos de processos de segmentação utilizando-se o EKF.



### Método *SEGMENTS*

O método chamado *SEGMENTS* (Roumeliotis & Bekey, 2000) usa uma equação não-linear para descrever o modelo do processo. Essa equação do processo tem como modelo uma sequência de três pontos lasers consecutivos numa superfície plana (rectilínea, Figura 3.5), sendo que o vector de estados contém a distância estimada  $r(k)$  e o ângulo interno estimado  $\alpha(k)$  entre dois pontos consecutivos no varrimento laser. O modelo de medição é do tipo linear e depende apenas da distância  $r(k+1)$  observada no instante corrente. As expressões que definem o modelo dinâmico do sistema são:

$$\begin{cases} r(k+1) = \frac{\sin(\theta(k))}{\sin(\theta(k)+\Delta\theta)} r(k) \\ \theta(k+1) = \theta(k) + \Delta\theta \end{cases} \quad (3.14)$$

sendo o vector de estado

$$\mathbf{x}(k) = [r(k) \quad \theta(k)]^T \quad (3.15)$$

Em espaço de estados, o modelo do sistema é representado por:

$$\begin{aligned} x_1(k+1) &= \frac{\sin(x_2(k))}{\sin(x_2(k)+\Delta\theta)} x_1(k) \\ x_2(k+1) &= x_2(k) + \Delta\theta \end{aligned} \quad (3.16)$$

Como em (3.12),  $\mathbf{H} = [1 \ 0]$  mas, por se tratar de um modelo do processo não-linear, a matriz de transição  $\mathbf{F}$  é dada por:

$$\mathbf{F} = \nabla \mathbf{f}(\mathbf{x}(k)) = \begin{bmatrix} \frac{\sin(\theta(k))}{\sin(\theta(k)+\Delta\theta)} & \frac{\sin(\theta(k))}{\sin^2(\theta(k)+\Delta\theta)} \cdot r(k) \\ 0 & 1 \end{bmatrix} \quad (3.17)$$

em que  $\nabla$  é o operador Jacobiano.

A Figura 3.6 ilustra um exemplo de segmentação utilizando este método. Neste caso, assim como para todos os demais discutidos neste trabalho, os filtros de Kalman são implementados de forma a que o algoritmo seja sempre reiniciado quando um ponto de

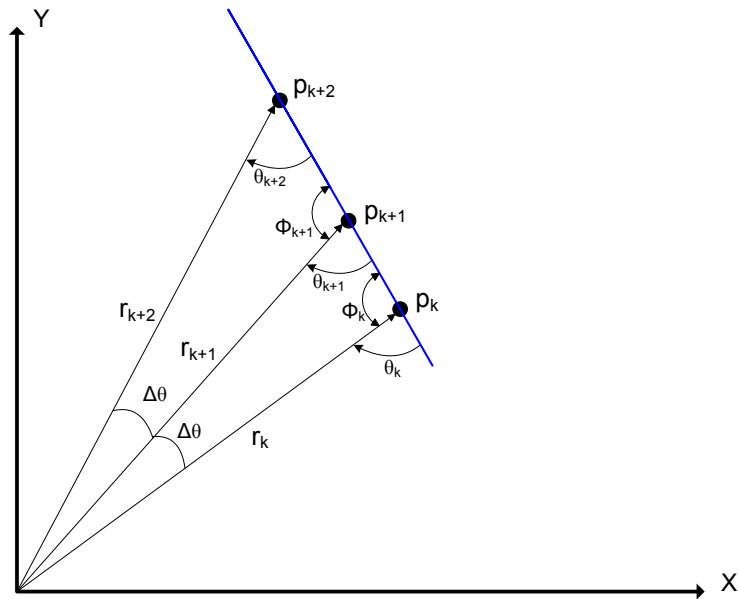


Figura 3.5: Modelo geométrico que define o método SEGMENTS

quebra é detectado, ou seja, o filtro inicia um novo ciclo de predição com base nas condições iniciais de um novo processo.

### Método ADA1

O processo de detecção ADA1 (Zhang et al., 2003) usa uma equação não-linear para descrever o modelo do processo. O processo contempla uma relação geométrica entre três pontos lasers consecutivos que seguem uma trajectória rectilínea (Figura 3.7), sendo que o vector de estados contém a distância estimada por dois pontos consecutivos no varrimento laser:

$$r(k+2) = \frac{r(k) \cdot r(k+1)}{2r(k) \cdot \cos(\Delta\theta) - r(k+1)} \quad (3.18)$$

O modelo de medição é do tipo linear  $z = [r(k+2)]$ , com matriz de transição  $\mathbf{H} = [1 \ 0]$ . O modelo do processo é definido por duas variáveis de estado:

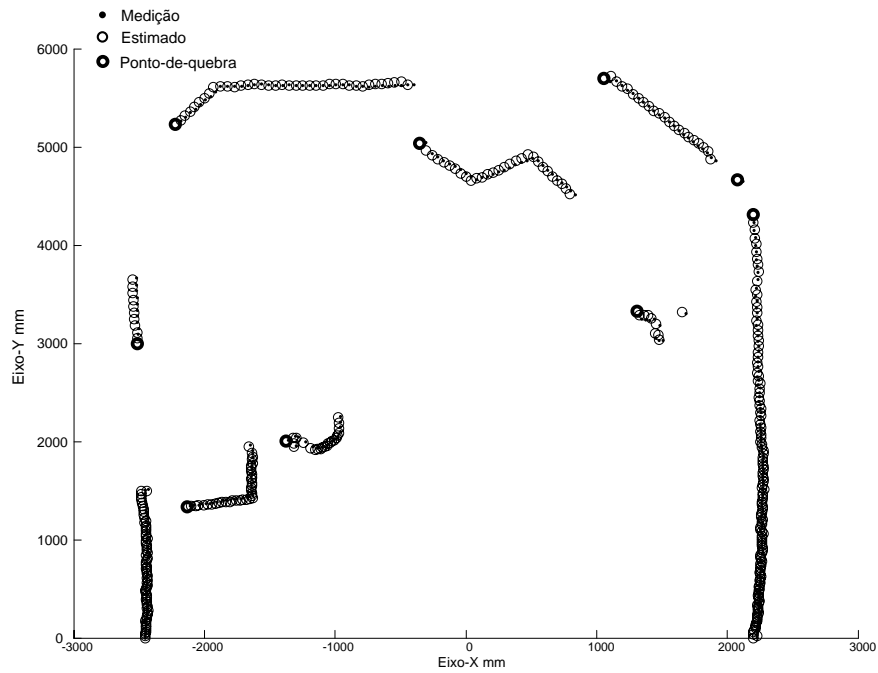


Figura 3.6: Exemplo de segmentação de dados-laser num ambiente interior usando o método SEGMENTS

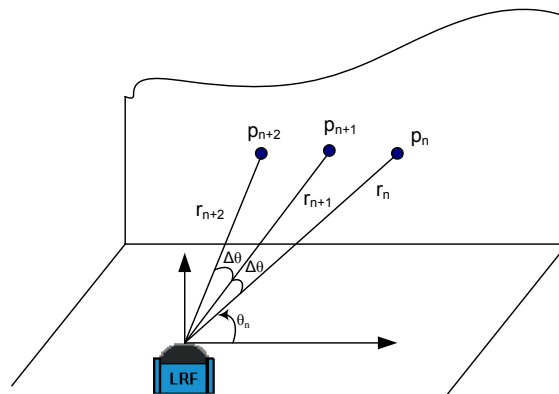


Figura 3.7: Relação entre os pontos-laser numa superfície plana utilizada para o método ADA1

$$\begin{aligned} x_1(k+1) &= r(k+2) \\ x_2(k+1) &= x_1(k) = r(k+1) \end{aligned} \quad (3.19)$$

A função não-linear de transição de estado neste modelo é dada por:

$$\mathbf{f}(\mathbf{x}(k)) = \begin{bmatrix} \frac{x_1(k)x_2(k)}{2x_1(k)\cos(\Delta\theta) - x_2(k)} \\ x_1(k) \end{bmatrix} \quad (3.20)$$

sendo o Jacobiano da matriz  $\mathbf{F}$  definido por:

$$\mathbf{F} = \begin{bmatrix} \frac{-x_2^2(k)}{\vartheta^2} & \frac{2x_1^2(k)\cos(\Delta\theta)}{\vartheta^2} \\ 1 & 0 \end{bmatrix} \quad (3.21)$$

onde  $\vartheta = 2x_1(k)\cos(\Delta\theta) - x_2(k)$ .

Este método apresenta resultados muito similares ao método SEGMENTS, sendo também útil na detecção de superfícies planas (retilíneas). Um exemplo de segmentação utilizando o método ADA1 está ilustrado na Figura 3.8.

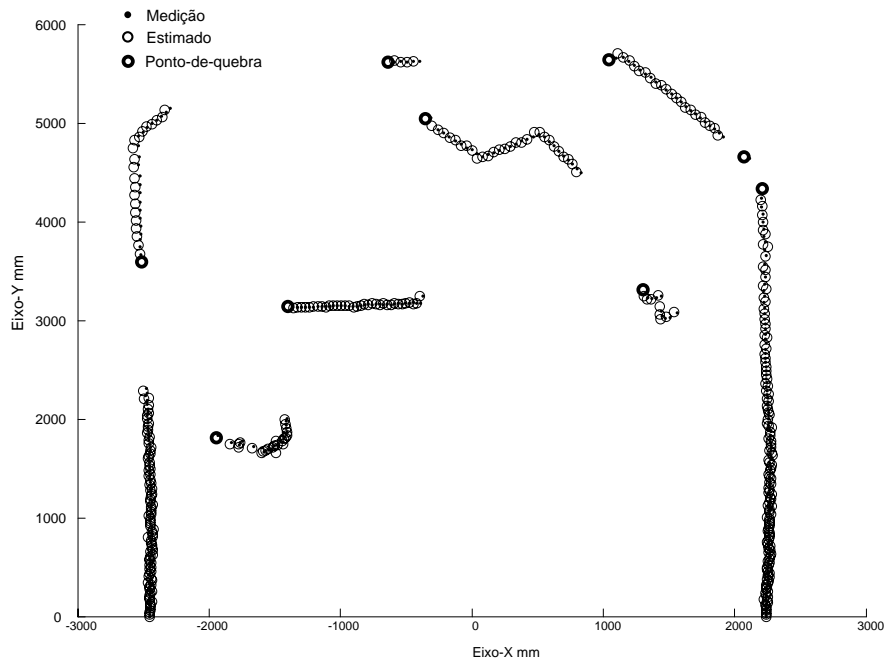


Figura 3.8: Exemplo de segmentação de dados-laser num ambiente interior usando o método ADA1

### Método ADA2

O processo de detecção ADA2 possui um modelo do processo construído com base em quatro pontos no plano laser e tem como condição de detecção o cálculo do gradiente espacial (Adams, 2001). Este método justifica-se nas aplicações em ambiente externo, onde a consideração de modelos que seguem rectas bem definidas (sub-secção 3.3) é pouco realista. Neste método, a técnica para detectar os pontos-de-quebra baseia-se no conceito de regiões de validação como função do valor do gradiente espacial, definido por uma sequência de quatro pontos adquiridos pelo laser. O ponto em que o valor do gradiente espacial exceda um limiar, ou seja, que esteja fora da RV do EKF, é prontamente identificado como “ponto de quebra”. O modelo discreto do processo é expresso pelo seguinte conjunto de equações:

$$\begin{aligned} r(k+2) &= \frac{r_k \cdot r_{k+1}}{2 \cdot r_k \cos \Delta\theta - r_{k+1}} \\ r^-(k+3) &= \frac{(2r_k - r_{k+1})r_{k+2}}{r_k - 2r_{k+1} \cos \Delta\theta + 2r_k \cos 2\Delta\theta} \\ r^+(k+3) &= r_{k+3}^- + \frac{N_1}{D_1} \cdot \frac{N_2}{D_2 + D_3 + D_4 + D_5} \end{aligned} \quad (3.22)$$

em que,

$$\begin{aligned} N_1 &= 2r_{k+2}(r_k^2 + r_{k+1}^2 - 2r_k r_{k+1} \cos \Delta\theta) \\ N_2 &= 2r_k r_{k+2} \cos \Delta\theta - r_k r_{k+1} - r_{k+1} r_{k+2} \\ D_1 &= r_k - 2r_{k+1} \cos \Delta\theta + 2r_k \cos 2\Delta\theta \\ D_2 &= -2r_k r_{k+1}^2 - r_{k+1}^2 r_{k+2} + 2r_k^2 r_{k+1} \cos \Delta\theta + 2r_{k+1}^3 \cos \Delta\theta \\ D_3 &= r_k r_{k+1} r_{k+2} \cos \Delta\theta - r_k r_{k+1}^2 \cos 2\Delta\theta + r_k^2 r_{k+2} \cos 2\Delta\theta \\ D_4 &= r_{k+1}^2 r_{k+2} \cos 2\Delta\theta - r_k^2 r_{k+1} \cos 3\Delta\theta - 2r_k r_{k+1} r_{k+2} \cos 3\Delta\theta \\ D_5 &= r_k^2 r_{k+2} \cos 4\Delta\theta \end{aligned} \quad (3.23)$$

Os sinais “mais” e “menos” presentes nas expressões (3.22) e (3.23) estão relacionados com um valor de gradiente espacial positivo e negativo (Adams, 2001) e (Zhang et al., 2003) respectivamente. A Figura 3.9 ilustra essa situação.

A partir do vector de estados  $x_{(k)} = [r(k) \ r(k+1) \ r(k+2)]^T$ , o modelo do sistema é definido por:

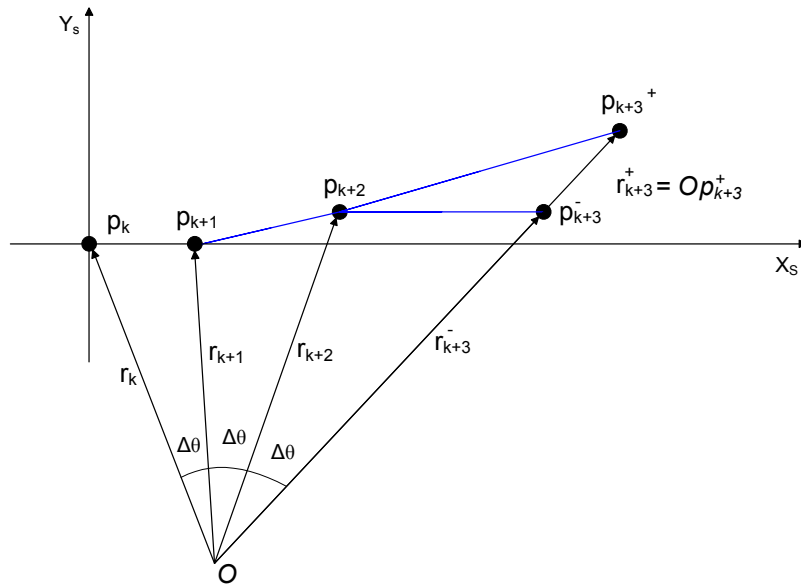


Figura 3.9: Relação entre os pontos-laser numa superfície plana para o método ADA2

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ x_3^-(k+1) \\ x_3^+(k+1) \end{bmatrix} = \begin{bmatrix} x_2(k) \\ x_3(k) \\ f_k^-[x_1, x_2, x_3, \Delta\theta] \\ f_k^+[x_1, x_2, x_3, \Delta\theta] \end{bmatrix} \quad (3.24)$$

sendo  $f^-[x_1, x_2, x_3, \Delta\theta]$  e  $f^+[x_1, x_2, x_3, \Delta\theta]$  calculadas a partir de (3.22).

A título de exemplo, para o caso em que se utiliza o gradiente negativo como “predição” da observação, i.e.  $r^-(k+3)$ , o valor da matriz de transição  $\mathbf{F}$  é dado por:

$$\mathbf{F} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ \frac{2x_3(k) \cdot n_1 - n_2(1+2\cos(2\Delta\theta))}{n_2^2} & \frac{-x_3(k) \cdot n_1 + n_2(2\cos(\Delta\theta))}{n_2^2} & \frac{(2x_1(k) - x_2(k)) \cdot n_2}{n_2^2} \end{bmatrix} \quad (3.25)$$

em que

$$\begin{aligned} n_1 &= 2x_1(k)x_3(k) - x_2(k)x_3(k) \\ n_2 &= x_1(k) + 2x_1(k)\cos(2\Delta\theta) - 2x_2(k)\cos(\Delta\theta) \end{aligned} \quad (3.26)$$

A Figura 3.10 ilustra um exemplo de segmentação utilizando a abordagem apresentada anteriormente (ADA2). Este método tem o inconveniente de ser mais complexo e de difícil implementação, particularmente para manter o filtro consistente com o modelo. Pela análise de resultados experimentais, conclui-se que este método de segmentação não traz vantagens em relação aos demais.

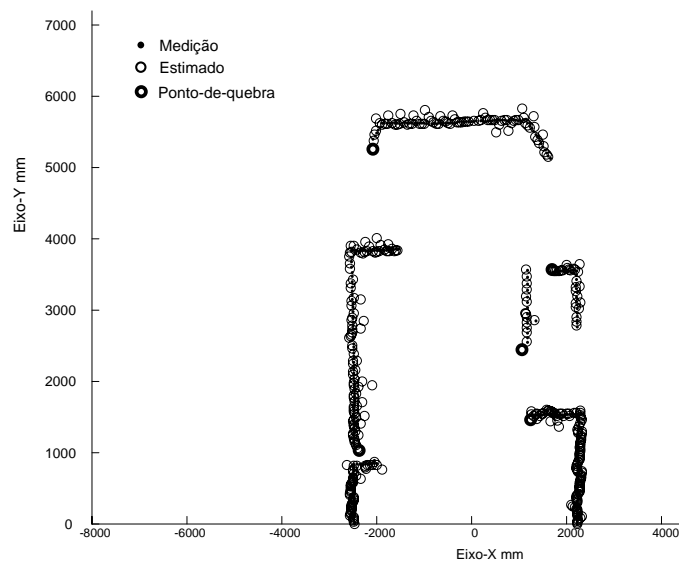


Figura 3.10: Exemplo de segmentação de dados-laser num ambiente interior usando o método ADA2

## Capítulo 4

# Extracção de primitivas geométricas

### Conteúdo

---

4.1	Formulação do problema . . . . .	38
4.2	Extracção de rectas . . . . .	40
4.3	Extracção de círculos . . . . .	42
4.4	Extracção de elipses . . . . .	45
4.5	Extracção de primitivas com filtros de Kalman . . . . .	46

---

**E**M robótica móvel ou, para ser mais abrangente, em várias áreas tecnológicas que envolvam percepção e interpretação sensorial, é comum a utilização de primitivas geométricas como forma concisa e de fácil interpretação dos dados extraídos de sensores (LRF, sonares, visão, entre outros). Em situações mais próximas do contexto deste trabalho, podem-se citar diversos casos da utilização de primitivas geométricas, como por exemplo na localização e navegação em ambientes interiores (Borges & Aldon, 2004) e exteriores (Vandorpe et al., 1996), (Siadat et al., 1997), em SLAM (Zhang, Xie, & Adams, 2005), na classificação de objectos (Fuerstenberg & Willhoeft, 2001), para previsão de colisões (Mendes, 2004), entre outros. O principal contributo da utilização de primitivas



geométricas está relacionado com a facilidade de interpretação e a simplificação no tratamento dos dados. Isso fica muito evidente quando, por exemplo, um peão é detectado pelo LRF como um segmento de vários pontos (nuvem de pontos) e que, após a extracção de um círculo, este mesmo objecto é representado por três parâmetros escalares (centro e raio), sem perder consideravelmente a informação “útil” que o caracteriza.

No âmbito desta tese, a utilização de primitivas geométricas está restrita a círculos, segmentos de rectas e elipses (Figura 4.1). Uma vez que os objectos são detectados e classificados, um vector de parâmetros característicos (construído como uma “identidade” de cada objecto) é utilizado para caracterizar o objecto. Nos elementos deste vector, incluem-se os parâmetros da primitiva geométrica seguindo o seguinte critério:

- Círculos: utilizados para representar, geometricamente, peões, postes e troncos de árvores;
- Rectas: utilizadas para representar paredes e carros (objecto nº 3 na Figura 4.1);
- Elipses: utilizadas em situações particulares, como por exemplo na representação geométrica do veículo *CyberCab* da *Yamaha* (objecto nº 2 na Figura 4.1).

Como exemplo prático podemos referir o módulo de navegação Estimador da Pose do Veículo (VPE) (Bento & Nunes, 2005), que utiliza o laser para detectar algumas *landmarks* que são representadas por círculos, cujos parâmetros são estimados pelo módulo de extracção de primitivas do MTDTS. Nas aplicações em ambiente interior, é comum a utilização de rectas para caracterizar o ambiente (Roumeliotis & Bekey, 2000) devido à elevada ocorrência de paredes.

## 4.1 Formulação do problema

A extracção de primitivas geométricas pode ser dividida em duas técnicas: *clustering* (e.g. métodos baseados na transformada de Hough) e métodos do tipo *Least Squares* (LS). Neste trabalho, será abordada unicamente a segunda categoria. Devido ao facto de linhas,

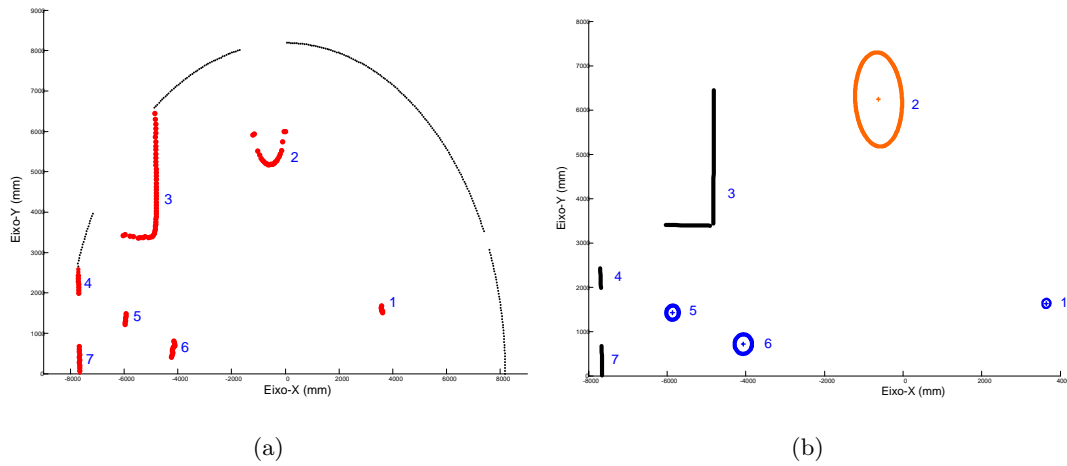


Figura 4.1: Exemplo prático da utilização de primitivas geométricas para representar objectos segmentados em dados laser. (a) Sete segmentos detectados no processo de segmentação; (b) Primitivas geométricas extraídas de cada segmento de pontos.

círculos e elipses poderem ser descritos como casos particulares de cónicas, a formulação do problema de extracção de cada primitiva geométrica pode ser feita de forma genérica usando-se a formulação de uma cónica cuja equação num plano é dada por:

$$ax^2 + 2bxy + cy^2 + 2dx + 2ey + f = 0 \quad (4.1)$$

onde  $a$  e  $c$  são diferentes de zero. Temos então as seguintes condições para se definir um tipo particular de cónica:

- Elipse:  $(b^2 - ac) < 0$ ;
- Círculo:  $a = c$  e  $b = 0$ ;
- Recta:  $a = b = c = 0$ .

## 4.2 Extracção de rectas

De maneira a simplificar a notação, a forma geral de uma recta pode ser expressa por:

$$ax + by + c = 0, \quad (4.2)$$

com a respectiva forma padrão:

$$y = mx + q \quad (4.3)$$

O processo de extracção de rectas tem por objectivo substituir o conjunto de pontos, no espaço Cartesiano, que definem um determinado segmento  $S_j$  por um, ou mais, segmentos de rectas que aproximem aqueles pontos. A primeira abordagem aqui apresentada é o método de regressão linear. Provavelmente, este é o método mais comum e conhecido para extracção de rectas, cujo objectivo é minimizar o erro quadrático médio ao longo de um dado eixo (neste caso o *eixo vertical*). Por motivos de comodidade, neste capítulo o termo *eixo vertical* refere-se a  $Y$  e *eixo horizontal* refere-se a  $X$ . O método usando regressão linear apresentado por (Vandorpe et al., 1996), é definido por um conjunto parâmetros de regressão:

$$T_1 = S_{xy} \cdot n - S_x S_y \quad T_2 = S_{xx} \cdot n - S_x^2 \quad (4.4)$$

em que

$$S_x = \sum_{i=1}^n x_i \quad S_y = \sum_{i=1}^n y_i \quad S_{xx} = \sum_{i=1}^n x_i^2 \quad S_{yy} = \sum_{i=1}^n y_i^2 \quad S_{xy} = \sum_{i=1}^n x_i y_i \quad (4.5)$$

Os parâmetros da recta, na forma padrão, são:

$$\begin{cases} m = \frac{T_1}{T_2} \\ q = \frac{(S_y - mS_x)}{n} \end{cases} \quad (4.6)$$

Em (Siadat et al., 1997) o problema da regressão linear para a recta expressa por (4.2), tem a seguinte solução:

$$\begin{cases} a = S_x S_{yy} - S_y S_{xy} \\ b = S_y S_{xx} - S_x S_{xy} \\ c = S_{xy}^2 - S_{xx} S_{yy} \end{cases} \quad (4.7)$$

em que  $S_x$ ,  $S_y$ ,  $S_{xy}$ ,  $S_{xx}$  e  $S_{yy}$  são calculados como em (4.5). Daqui obtêm-se os parâmetros na forma padrão:  $m = -a/b$  e  $q = -c/b$ . Considerando ainda a equação (4.2), pode-se formular o mesmo problema matricialmente, tal que:

$$\mathbf{B}\mathbf{u} = \mathbf{0} \quad (4.8)$$

sendo

$$\mathbf{u} = [a \ b \ c]^T \quad (4.9)$$

e

$$\mathbf{B} = \begin{bmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ \vdots & \vdots & \vdots \\ x_n & y_n & 1 \end{bmatrix} \quad (4.10)$$

As expressões (4.9) e (4.10) definem o problema clássico de resolução dum sistema linear sobre-determinado (4.8) em função dos coeficientes  $\mathbf{u}$  e de uma matriz de dados  $\mathbf{B}$ . O sistema (4.8) traduz uma equação linear de parâmetros desconhecidos  $\mathbf{u}$  a estimar,

cuja solução passa por técnicas já conhecidas, e.g. decomposição  $QR$ , decomposição em valores singulares (SVD) ou um algoritmo do tipo LS, sendo conhecida a matriz  $\mathbf{B}$ . A título de exemplo, utilizando a função *svd* do software Matlab<sup>®</sup>, apresenta-se de seguida uma solução para o problema:

1) Constrói-se a matriz  $\mathbf{B}$  (4.10) com dados do laser  $[x_i \ y_i]$ , e a terceira coluna com valores unitários;

2) Obtém-se a decomposição *svd* da matriz  $B$ . O vector correspondente ao menor valor singular é a solução para  $\mathbf{u}$ . Especificamente para o caso da função *svd* em Matlab<sup>®</sup>, se  $\mathbf{B}=\mathbf{U}\mathbf{D}\mathbf{V}'$  sendo  $\mathbf{D}$  a matriz positiva diagonal de entradas, e ordenada de modo descendente, o vector solução  $u$  é a última coluna da matriz  $\mathbf{V}$ .

Em situações particulares, como por exemplo na identificação de cantos em ambientes interiores ou na caracterização de objectos rectangulares (e.g. carros), é conveniente que o conjunto de pontos que definem um segmento  $S_j$  seja representado por, pelo menos, dois segmentos de rectas; então torna-se necessário detectar o ponto de intersecção das rectas. Para colmatar esta situação, optou-se por utilizar o método IEPF (*Iterative End-Point Fit*) (Duda & Hart, 1973) no qual, primeiramente, uma recta é extraída entre os pontos extremos de um segmento de pontos e, de maneira recursiva, o algoritmo calcula o ponto do segmento com a máxima distância até a recta. Se esta distância for superior a um limiar então a recta é “dividida” em dois intervalos, e o algoritmo IEPF reinicia recursivamente, a partir daquele ponto, o mesmo processo, até que a distância entre os pontos restantes e a nova recta esteja abaixo do limiar.

### 4.3 Extracção de círculos

A expressão geral de um círculo com centro  $(x_c, y_c)$  é

$$(x - x_c)^2 + (y - y_c)^2 = r_c^2 \quad (4.11)$$

em que o raio é dado por:

$$r_c = \sqrt{(x - x_c)^2 + (y - y_c)^2} \quad (4.12)$$

O método proposto por (Vandorpe et al., 1996) toma como base as variáveis de regressão (4.5), e os parâmetros que definem o círculo são calculados por:

$$\begin{cases} x_c = S_x/n \\ y_c = S_y/n \\ r_c = \sqrt{\rho_x^2 + \rho_y^2} \end{cases} \quad (4.13)$$

em que  $S_x$  e  $S_y$  são dados em (4.5) e

$$\begin{cases} \rho_x^2 = \sum_{i=1}^n \frac{(x_i - x_c)^2}{n-1} \\ \rho_y^2 = \sum_{i=1}^n \frac{(y_i - y_c)^2}{n-1} \end{cases} \quad (4.14)$$

Em situações específicas, como por exemplo quando se quer extrair um círculo a partir de poucos pontos-laser (e.g. para objectos de pequena dimensão e distantes do sensor), o método de (Vandorpe et al., 1996) é extremamente útil. Outro método para determinar os parâmetros de um círculo, proposto por (Guivant, Nebot, & Durrant-Whyte, 2000) e descrito em (Lee, 2001), consiste no cálculo do diâmetro do círculo a partir da média dos pontos-laser e da abertura angular (ângulo definido entre o primeiro  $r_i$  e o último ponto  $r_f$  de um segmento com  $n$  pontos), resultando no valor do diâmetro dado pela expressão:

$$2r_c = \Delta\theta \sum_{n=i}^f r_n \quad (4.15)$$

onde  $r_n$  é o conjunto de pontos que define um segmento, e  $\Delta\theta$  é a resolução angular do sensor laser.

Assumindo-se que o centro do círculo está colinear com a recta centrada na abertura

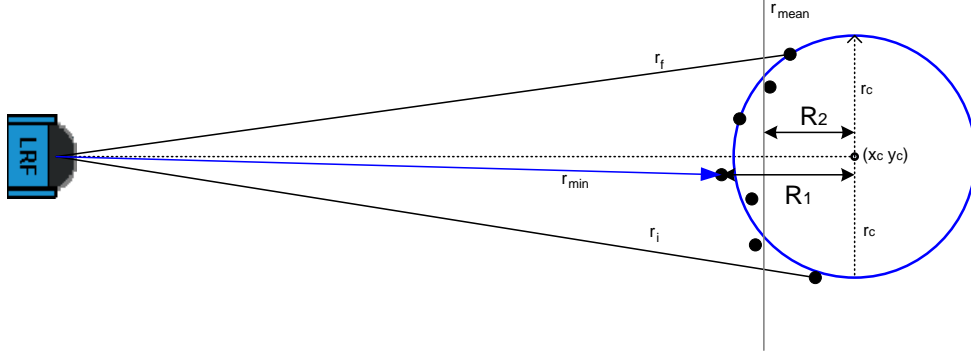


Figura 4.2: Modelo geométrico utilizado nos métodos GUI1 e GUI2.

angular que define um segmento (Figura 4.2), as coordenadas que definem o centro  $(x_c, y_c)$  podem ser determinadas, de forma aproximada, considerando-se duas suposições:

GUI1: Assume-se que a distância entre as coordenadas do centro e do ponto-laser mais próximo  $r_{min}$  seja igual ao raio  $R1$  do círculo. Isto equivale a:

$$\begin{cases} x_c = (r_{min} + r_c) \cdot \cos(\theta_{min}) \\ y_c = (r_{min} + r_c) \cdot \sin(\theta_{min}) \\ \text{onde } r_{min} = \min(r_i), \quad i \in [n, k] \end{cases} \quad (4.16)$$

GUI2: Assume-se que a distância entre as coordenadas do centro e da média dos pontos-laser ( $r_{mean}$ ) seja de um valor  $R2 = r_c/2$  (Figura 4.2). Isto é equivalente a:

$$\begin{cases} x_c = (r_{mean} + r_c/2) \cdot \cos(\theta_{mean}) \\ y_c = (r_{mean} + r_c/2) \cdot \sin(\theta_{mean}) \\ \text{onde } r_{mean} = \text{mean}(r_i), \quad i \in [n, k] \end{cases} \quad (4.17)$$

Uma outra abordagem para extrair círculos, ou elipses, apresentada em (Gander et al., 1994), tem por base minimizar a soma do quadrado das distâncias entre pontos num plano de modo a calcular os parâmetros da primitiva geométrica de interesse. Usando a equação geral das cónicas, um círculo pode ser expresso matematicamente por:

$$a(x^2 + y^2) + b_1x + b_2y + c = 0 \quad (4.18)$$

Matricialmente, como em (4.8), o sistema de equações é dado por:

$$\mathbf{B}\mathbf{u} = \begin{bmatrix} x_1^2 + y_1^2 & x_1 & y_1 & 1 \\ x_2^2 + y_2^2 & x_2 & y_2 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ x_n^2 + y_n^2 & x_n & y_n & 1 \end{bmatrix} \begin{bmatrix} a \\ b_1 \\ b_2 \\ c \end{bmatrix} = \mathbf{0} \quad (4.19)$$

Desta forma o cálculo dos parâmetros do círculo reduz-se ao problema clássico de solucionar  $\|\mathbf{B} \cdot \mathbf{u} = \mathbf{0}\|$ , sujeito à restrição  $\|\mathbf{u}\| = 1$ . Finalmente, o centro e o raio do círculo são obtidos usando as expressões:

$$\begin{cases} x_c = -b_1/2a \\ y_c = -b_2/2a \\ r_c = \sqrt{\frac{\|\mathbf{b}\|^2}{4a^2} - \frac{c}{a}} \end{cases} \quad (4.20)$$

onde  $\mathbf{b} = [b_1 \ b_2]^T$ . Novamente, uma decomposição do tipo *svd* ou um algoritmo *LS* podem ser utilizados para resolver (4.19).

## 4.4 Extracção de elipses

Nesta secção apresenta-se, de uma maneira concisa, dois métodos para extracção de elipses. A forma standard da equação que define uma elipse, com centro  $(x_e, y_e)$ , é dada por:

$$\frac{(x - x_e)^2}{a^2} + \frac{(y - y_e)^2}{b^2} = 1 \quad (4.21)$$

Mais especificamente, os parâmetros a serem obtidos e que definem uma elipse são: o centro, os eixos menor e maior, e a orientação. Um trabalho publicado por (Fitzgibbon



et al., 1999) faz a comparação entre vários métodos de extracção de elipses (entre eles os métodos publicados por (Gander et al., 1994)), e propõe um método relativamente simples e eficiente para extracção de elipses, impondo-se a seguinte condição (também comum na maioria das abordagens):  $4\mathbf{ac} - \mathbf{b}^2 = 1$ , reduzindo assim o problema à minimização da seguinte expressão:  $\mathbf{E} = \|\mathbf{D}\mathbf{a}\|^2$ , onde  $\mathbf{a}$  é o vector de parâmetros da elipse, e  $\mathbf{D}$  é dada por:

$$\mathbf{D} = \begin{bmatrix} x_1^2 & x_1y_1 & y_1^2 & x_1 & y_1 & 1 \\ x_2^2 & x_2y_2 & y_2^2 & x_2 & y_2 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_n^2 & x_ny_n & y_n^2 & x_n & y_n & 1 \end{bmatrix} \quad (4.22)$$

Tratando o problema de extracção de elipse de forma mais geral, cuja formulação é semelhante à descrita nos métodos do tipo LS, (Gander et al., 1994) propõe vários algoritmos do tipo LS e do tipo “minimização do erro quadrático” para extrair elipses. Alguns resultados de extracção de elipses usando os métodos discutidos são apresentados na secção 7.2.

## 4.5 Extracção de primitivas com filtros de Kalman

Abordagens para extracção de primitivas baseadas no filtro de Kalman são descritas em (Roumeliotis & Bekey, 2000), (Zhang et al., 2003) e (Adams, 2001). O cerne deste tipo de método está no modelo do processo que deve ter em consideração as equações que descrevam a primitiva geométrica de interesse. Para o caso do círculo, o método descrito em (Zhang et al., 2003) tem como modelo de estado:

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ x_3(k+1) \\ x_4(k+1) \end{bmatrix} = \begin{bmatrix} x_1(k) = x_c \\ x_2(k) = y_c \\ x_3(k) = r \\ d_{n+2} = g[x_1(k), x_2(k), x_3(k), x_4(k)] \end{bmatrix} \quad (4.23)$$

O modelo geométrico está representado na Figura 4.3, e é expresso pelas equações:

$$\begin{cases} r^2 = d_{n+1}^2 + a^2 - 2d_{n+1}a \cos(\Delta\theta - \theta) \\ r^2 = d_{n+2}^2 + a^2 - 2d_{n+2}a \cos(2\Delta\theta - \theta) \\ a^2 = x_c^2 + y_c^2 \end{cases} \quad (4.24)$$

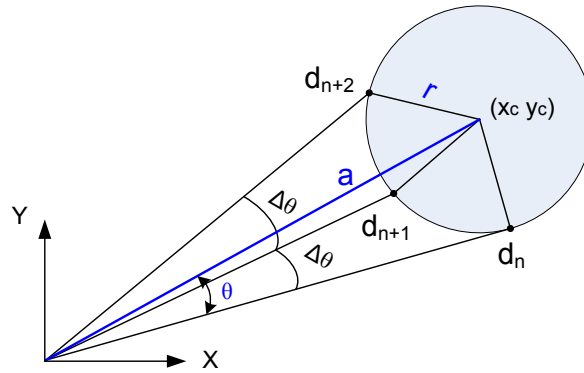


Figura 4.3: Relação geométrica dos pontos laser numa dada superfície cilíndrica

Esse método foi inspirado em métodos similares utilizados para o caso da detecção de rectas, como nos trabalhos de (Roumeliotis & Bekey, 2000) e (Zhang et al., 2003). Este tipo de abordagem tem a vantagem de que, além de detectar e extrair primitivas geométricas, os pontos de quebra também são identificados, i.e. este método efectua a segmentação dos pontos e, em simultâneo, detecta a primitiva geométrica em questão. Uma desvantagem deste tipo de método está no modelo do sistema que, na maioria das vezes, é do tipo não-linear e, no caso de se usar um EKF, o cálculo do Jacobiano é uma tarefa complexa. Uma maneira de ultrapassar aquela dificuldade pode passar pelo uso de um UKF (*Unscented Kalman Filter*) como em (Julier, Uhlmann, & DurrantWhyte, 2000).

Em (Amara et al., 1996) descreve-se um método para extrair rectas e círculos, baseado no KF, tendo por base a equação geral (4.2) e (4.11). Este método utiliza aproximações para se obter modelos lineares tanto no processo como na observação, facilitando a sua implementação. Os resultados são muito satisfatórios e os modelos são relativamente simples. Como exemplo, para o caso de uma círculo, a equação de estado é definida directamente por:

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ x_3(k+1) \end{bmatrix} = \begin{bmatrix} x_1(k) = x_c \\ x_2(k) = y_c \\ x_3(k) = r \end{bmatrix} \quad (4.25)$$

Com base em (4.11) e nos dados de medida, em coordenadas Cartesianas  $(x(k) \ y(k))$ , e fazendo-se uma aproximação no sistema para torná-lo linear, chega-se à matriz de medidas:

$$H = \begin{bmatrix} 2(x(k) - \hat{x}_1(k-1)) \\ 2(y(k) - \hat{x}_2(k-1)) \\ 2(\hat{x}_3(k-1)) \end{bmatrix} \quad (4.26)$$

Nesta abordagem a inovação assume a expressão (Amara et al., 1996):

$$v = (x(k) - x_1(k-1))^2 + (y(k) - x_2(k-1))^2 - x_3(k)^2 \quad (4.27)$$

Neste método, o processo de estimação dos parâmetros do círculo é recursivo (a Figura 4.4(a) ilustra as estimativas no processo), com o resultado final da última estimação apresentado na Figura 4.4(b).

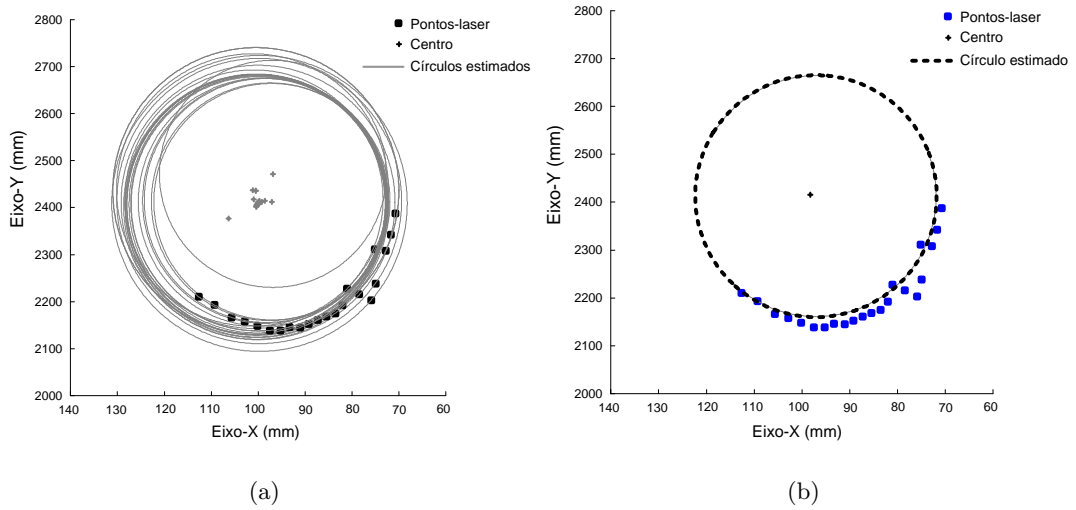


Figura 4.4: Exemplo de extracção de círculo usando um algoritmo baseado em KF. (a) Círculos extraídos ao longo do processo de estimação; (b) Círculo estimado ao final do processo de estimação usando um KF.



## Capítulo 5

# Seguimento de objectos

### Conteúdo

---

5.1	Representação em espaço-de-estados . . . . .	52
5.2	Associação de dados . . . . .	57
5.3	Seguimento de objectos usando dados do LRF . . . . .	64
5.4	Cooperação sensorial . . . . .	74

---

**E**STE capítulo trata do módulo de seguimento de objectos (modelos e filtros), e do módulo de associação de dados do sistema MTDTS. Serão discutidos brevemente alguns filtros estocásticos, nomeadamente o filtro de Kalman (e a sua variante EKF), o filtro de Partículas (na sua forma padrão) e, com mais ênfase, o filtro probabilístico de associação de dados (PDAF); ainda neste mesmo contexto, são discutidos os modelos cinemáticos que representam a dinâmica dos objectos, bem como as matrizes de covariância do ruído do processo e das observações. Directamente relacionado com os filtros estocásticos, duas técnicas de associação de dados serão discutidas: regiões de validação do tipo rectangular e elipsoidal. Na última secção, é apresentado um método de cooperação sensorial entre um LRF e uma câmara, utilizado no processo de correspondência entre os objectos detectados pelo laser e aqueles detectados pela câmara, cujo resultado é utilizado na combinação de classificadores.

## 5.1 Representação em espaço-de-estados

A representação em espaço de estados de um sistema dá-se em termos de um conjunto de parâmetros escalares, chamados variáveis de estado, de modo que o sistema seja convenientemente definido em termos de um vector de estados  $\mathbf{x} = [x_1, \dots, x_n]^T$  e das entradas associadas. Em aplicações práticas, não há uma condição de certeza absoluta, ou determinística, do vector de estados que define um dado sistema, por isso uma condição estocástica é utilizada como abordagem na caracterização das incertezas de cada variável de estado. Se essas incertezas têm uma distribuição do tipo Gaussiana, então cada variável pode ser totalmente definida em termos do seu valor esperado  $\mu_i$  (média) e do desvio padrão. O estado do sistema é representado pelo vector de estados estimado, definido como o seu valor esperado ou médio, e a respectiva matriz de covariância:

$$\hat{\mathbf{x}} = \begin{bmatrix} \hat{x}_1 \\ \vdots \\ \hat{x}_n \end{bmatrix} \triangleq E[\mathbf{x}] \quad (5.1)$$

$$P_x = \begin{bmatrix} \sigma_{11}^2 & \dots & \sigma_{1n}^2 \\ \vdots & \ddots & \vdots \\ \sigma_{1n}^2 & \dots & \sigma_{nn}^2 \end{bmatrix} \triangleq E[(\mathbf{x} - \hat{\mathbf{x}})(\mathbf{x} - \hat{\mathbf{x}})^T] \quad (5.2)$$

sendo  $E[\mathbf{x}]$  definido como o valor esperado ou operador valor médio de  $\mathbf{x}$ , cuja expressão é dada por:  $E[\mathbf{x}] \triangleq \int_{-\infty}^{\infty} \mathbf{x}p(\mathbf{x})d\mathbf{x}$ , onde a função  $p(\mathbf{x})$  representa a função densidade probabilística (PDF) do vector  $\mathbf{x}$ .

Independente do filtro estocástico e da técnica de associação a ser utilizada num sistema MTDTS, há determinadas considerações de projecto que são comuns:

- Definir o modelo do processo;
- Definir o modelo de medição/observação;

- Caracterizar o ruído: é comum, e conveniente, definir o ruído do processo e da medição como ruído branco de média nula e Gaussiano;

Uma vez definidos e caracterizados os modelos do processo e de observação, interessamos escolher um filtro que apresente o melhor desempenho. No entanto, o filtro *per se* (e.g. KF, EKF, PF, ou PDAF) não é geralmente o módulo crítico, sendo por sua vez o módulo de associação de dados e a técnica escolhida para implementá-lo uma das principais dificuldades em sistemas MTDTs. A secção 5.2 está dedicada a algumas das principais técnicas de associação de dados.

### 5.1.1 Modelos cinemáticos utilizados

O seguimento dos objectos detectados pelo LRF é realizado com base em modelos dinâmicos que caracterizam o comportamento cinemático dos objectos. Nesta dissertação são unicamente considerados modelos lineares. As observações da posição do objecto referem-se ao centro de massa (ponto característico) dos pontos-laser que definem um segmento  $S_k$  (3.1). Neste caso, estamos a tratar o problema de seguimento com base apenas nos centro de massa dos objectos, i.e. o vector de estados  $\mathbf{x}(k)$  tem como componentes: a posição Cartesiana do centro de massa do objecto (variáveis observáveis), a velocidade e, eventualmente, a aceleração estimadas (também num referencial Cartesiano). Cada classe de objecto de interesse é então caracterizada pelos seguintes modelos lineares dinâmicos:

1. Modelo *quase*-estático: objectos com velocidade nula, conseqüentemente com posição estacionária;
2. Modelo de aceleração como ruído branco: também conhecido como modelo cinemático de segunda ordem, no qual a velocidade é considerada constante;
3. Modelo de aceleração como processo *Wiener*: ou modelo de terceira ordem, no qual considera-se que o objecto possui aceleração constante.



Todos estes modelos podem ser descritos pelo modelo linear

$$\begin{aligned}\mathbf{x}(k+1) &= \mathbf{F}(k)\mathbf{x}(k) + \Gamma(k)\nu(k) \\ \mathbf{z}(k+1) &= \mathbf{H}(k)\mathbf{x}(k) + \omega(k)\end{aligned}\tag{5.3}$$

com vector de estados

$$\mathbf{x}(k+1) = [p_x \ v_x \ a_x \ p_y \ v_y \ a_y]^T\tag{5.4}$$

em que  $p = [p_x \ p_y]$  define a posição Cartesiana do centro de massa do objecto,  $v$  a sua velocidade e  $a$  a sua aceleração.

### Modelo de posição constante

Neste modelo a velocidade é considerada nula  $x_2(k) = v(k) = 0$ , mas, como na prática a posição não se mantém estritamente constante, as incertezas no estado (posição) são caracterizadas por ruído branco de média nula adicionado ao modelo do processo. Em termos discretos, este modelo é descrito pelas equações:

$$\begin{cases} p(k+1) = p(k) + \nu(k) \\ v(k+1) = 0 \\ a(k+1) = 0 \end{cases}\tag{5.5}$$

com as matrizes de transição (para cada uma das coordenadas) dadas por:

$$F = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}\tag{5.6}$$

$$\Gamma = [1 \ 0 \ 0]^T\tag{5.7}$$

**Modelo de velocidade constante**

No modelo de velocidade constante,  $a(k) = 0$ , as incertezas na velocidade são modeladas por ruído branco de média nula. Neste caso a velocidade é um processo *Wiener*, i.e. a integral do ruído branco, e o modelo do processo é descrito por:

$$\begin{cases} p(k+1) = p(k) + \Delta T v(k) + \Delta T^2/2 \nu(k) \\ v(k+1) = v(k) + \Delta T \nu(k) \\ a(k+1) = 0 \end{cases} \quad (5.8)$$

sendo  $\Delta T$  o período de amostragem. Uma vez mais as matrizes de transição são iguais para as coordenadas  $x$  e  $y$ , sendo neste caso definidas por:

$$F = \begin{bmatrix} 1 & \Delta T & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (5.9)$$

$$\Gamma = [\Delta T^2/2 \quad \Delta T \quad 0]^T \quad (5.10)$$

**Modelo de aceleração constante**

Na prática a aceleração não é estritamente constante, de modo que o modelo é descrito pelas equações:

$$\begin{cases} p(k+1) = p(k) + \Delta T v(k) + \Delta T^2/2 a(k) + \Delta T^2/2 \nu(k) \\ v(k+1) = v(k) + \Delta T a(k) + \Delta T \nu(k) \\ a(k+1) = a(k) + \nu(k) \end{cases} \quad (5.11)$$

As matrizes de transição são dadas por (para uma das dimensões Cartesianas):

$$F = \begin{bmatrix} 1 & \Delta T & \Delta T^2/2 \\ 0 & 1 & \Delta T \\ 0 & 0 & 1 \end{bmatrix} \quad (5.12)$$

$$\Gamma = [\Delta T^2/2 \quad \Delta T \quad 1]^T \quad (5.13)$$

### 5.1.2 Covariância dos ruídos do processo e das medidas

Em conformidade com os modelos definidos na sub-secção anterior, optou-se por estabelecer uma abordagem comum na definição das matrizes de covariância que representam o ruído adicionado ao modelo do processo. Para o modelos (5.5), (5.8) e (5.11), têm-se respectivamente as seguintes matrizes de covariância:

$$Q_1 = \Gamma \sigma_\nu^2 \Gamma^T = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \sigma_\nu^2 \quad (5.14)$$

sendo  $\sigma_{min} \leq \sigma_\nu^2 \leq \sigma_{max}$ , onde  $\sigma_{max}$  é o valor de incerteza máximo permitido na posição do objecto.

$$Q_2 = \Gamma \sigma_\nu^2 \Gamma^T = \begin{bmatrix} \frac{\Delta T^4}{4} & \frac{\Delta T^3}{2} & 0 \\ \frac{\Delta T^3}{2} & \Delta T^2 & 0 \\ 0 & 0 & 0 \end{bmatrix} \sigma_\nu^2 \quad (5.15)$$

sendo  $\sigma_{min} \leq \sigma_\nu^2 \leq \sigma_{max}$ , onde  $\sigma_{max}$  é o valor de incerteza máximo permitido na velocidade do objecto.

$$Q_3 = \Gamma \sigma_\nu^2 \Gamma^T = \begin{bmatrix} \frac{\Delta T^4}{4} & \frac{\Delta T^3}{2} & \frac{\Delta T^2}{2} \\ \frac{\Delta T^3}{2} & \Delta T^2 & \Delta T \\ \frac{\Delta T^2}{2} & \Delta T & 1 \end{bmatrix} \sigma_\nu^2 \quad (5.16)$$

sendo  $\sigma_{min} \leq \sigma_v^2 \leq \sigma_{max}$ , onde  $\sigma_{max}$  é o valor de incerteza máximo permitido na aceleração do objecto.

Para o modelo de observação, com as medidas em coordenadas polares, a matriz de covariância do ruído de medida é dada por:

$$R = \begin{bmatrix} \sigma_r^2 & \sigma_{r,\theta}^2 \\ \sigma_{\theta,r}^2 & \sigma_\theta^2 \end{bmatrix} \quad (5.17)$$

sendo  $\sigma_r^2$  a variância relativa à distância medida pelo LRF e  $\sigma_\theta^2$  a variância que representa a incerteza angular da medição. A matriz  $R$  definida em (5.17) traduz que as medidas são feitas directamente na forma polar (tal como originalmente enviadas pelo LRF), sendo comum considerar, na prática, nula a covariância das variáveis  $r$  e  $\theta$ . No entanto, como as variáveis de estado foram definidas em coordenadas Cartesianas, esta forma de  $R$  só pode ser utilizada com um filtro que trate as observações como não-lineares (e.g. EKF). Para o caso em que o modelo de observação é linear, as medidas são pré-convertidas da forma polar (original) para a forma Cartesiana: ( $x_1 = r_n \cos \theta_n$ ,  $x_4 = r_n \sin \theta_n$ ), assumindo a matriz  $R$  a seguinte forma (Bar-Shalom & Fortmann, 1988):

$$R = \begin{bmatrix} r_n^2 \sigma_\theta^2 \sin^2 \theta_n + \sigma_r^2 \cos^2 \theta_n & (\sigma_r^2 - r_n^2 \sigma_\theta^2) \sin \theta_n \cos \theta_n \\ (\sigma_r^2 - r_n^2 \sigma_\theta^2) \sin \theta_n \cos \theta_n & r_n^2 \sigma_\theta^2 \cos^2 \theta_n + \sigma_r^2 \sin^2 \theta_n \end{bmatrix} \quad (5.18)$$

## 5.2 Associação de dados

No seguimento de múltiplos objectos assume-se, na maioria dos casos, que os objectos se movem independentemente uns dos outros, e as associações observações-objectos são processadas considerando-se as incertezas das medidas, as incertezas nos modelos dinâmicos dos objectos e na *pose* do veículo. Das técnicas de associação de dados, a mais comum consiste na utilização de uma região de validação definida pelo valor quadrático da inovação normalizada (técnica NIS), conhecida por distância de *Mahalanobis*. Essa região é definida no espaço das medidas e assume que as incertezas envolvidas no processo são do tipo

Gaussianas.

### 5.2.1 Região de validação elipsoidal

Considerando que o filtro de seguimento, e.g. KF, já tenha sido inicializado, e assumindo que as medidas possuem incertezas cujo comportamento segue uma distribuição Gaussiana, pode-se então definir uma região no espaço das observações na qual a medida  $z(k+1)$  (cuja estimativa é  $\hat{z}(k+1|k)$ ) pertence com elevada probabilidade:

$$\tilde{V}_{k+1} = z : v^T(k+1)S^{-1}(k+1)v(k+1) \leq \gamma_n \quad (5.19)$$

sendo  $v$  a inovação,  $S$  é a matriz de covariância da inovação, e  $\gamma_n$  é um parâmetro que depende da dimensão  $n$  do vector de observações. A definição do valor de  $\gamma_n$  é baseada na distribuição Chi-quadrado, e o seu valor define o limiar da região de validação. A região  $\tilde{V}_{k+1}$  definida por (5.19) é um elipsóide (elipse num plano), e as medidas que pertencem a esta região são consideradas “medidas validadas”. Em muitas aplicações, há sempre alguma probabilidade de um objecto sob seguimento não ser detectado num determinado instante, e então nenhuma das medidas correntes correspondam ao objecto em seguimento. Também pode ocorrer que mais de uma medida, correspondente a um dado objecto, sejam validadas para um único objecto-alvo em seguimento, ou ainda que dois ou mais objectos tenham uma “intersecção” em suas respectivas RV para um ou mais valores de medidas. São causas comuns desses problemas: objectos que se “cruzam”, ruído de fundo, perda de sinal de medida, objectos “próximos” e ou com dinâmicas similares, etc. Quando não há associação válida dentro da RV, o filtro estocástico prediz o estado sem no entanto entrar no ciclo de correcção, tendo como consequência o aumento na incerteza do estado. Essa incerteza aumenta ao longo do tempo, como se ilustra na Figura 5.1, na qual a partir do instante  $(k+8)$  não há mais validação da variável de estado observada e o filtro segue sem o ciclo de correcção.

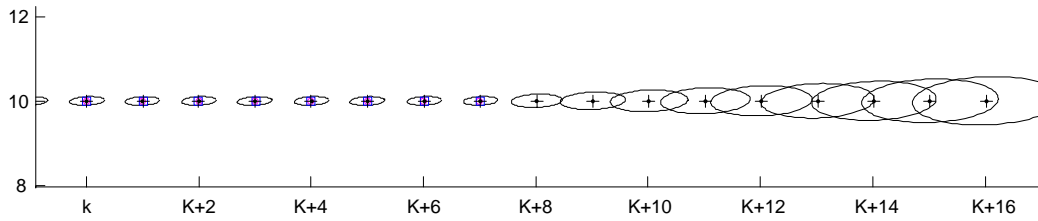


Figura 5.1: Regiões de validação elipsoidais utilizadas na associação de dados no seguimento de um único objecto num plano Cartesiano

### 5.2.2 Região de validação rectangular

A validação de dados usando uma RV rectangular utiliza como condição de aceitação ou rejeição apenas o valor da inovação. Se este valor estiver abaixo do limiar definido, então a medição é válida:

$$|z - \hat{z}| = |v| \leq K_G \sigma_G \quad (5.20)$$

O desvio padrão residual  $\sigma_G = \sqrt{\sigma_o^2 + \sigma_p^2}$  depende dos valores da variância da observação  $\sigma_o^2$  e do seu valor predito  $\sigma_p^2$ . O limiar  $K_G$  é obtido, tal como em (5.19), de uma tabela de distribuição Chi-quadrado e depende da dimensão da variável observada. Um exemplo de utilização de regiões de validação rectangulares está ilustrado na Figura 5.2. A forma da distribuição Chi-quadrado depende da dimensão da variável observada (Figura 5.3), e a área definida pelo valor limiar representa a probabilidade do objecto pertencer à região de validação.

### 5.2.3 Seguimento de um único objecto

Nesta secção será discutido o problema caracterizado pela ocorrência de múltiplas medidas pertencentes à RV relativa a um único objecto sob seguimento. Esse conjunto de medidas válidas consistem:

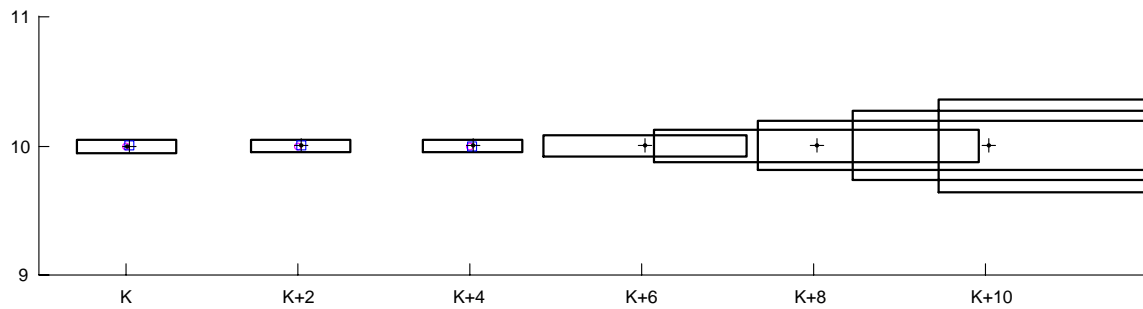


Figura 5.2: Ilustração gráfica das regiões de validação retangulares utilizadas na associação de dados no seguimento de um único objecto num plano Cartesiano

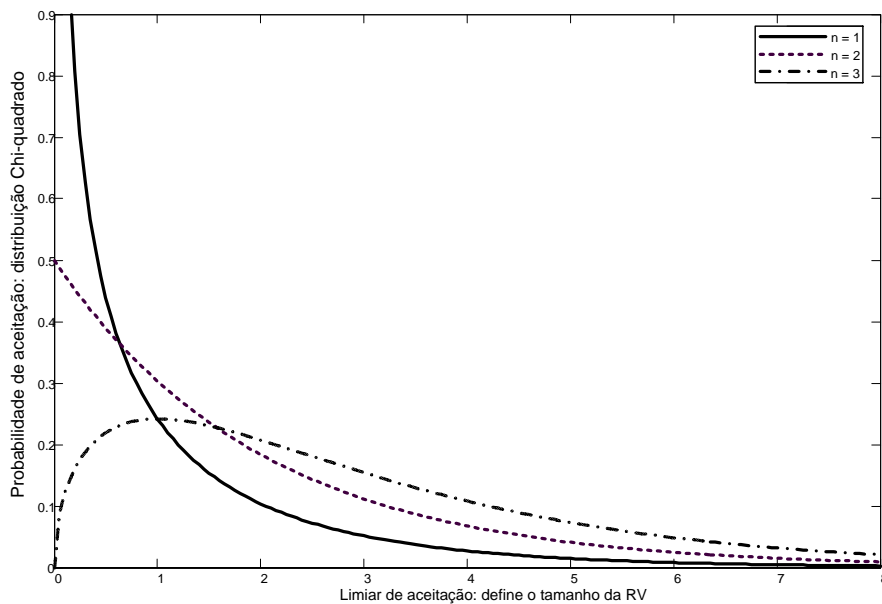


Figura 5.3: Distribuição Chi-quadrado definida em função da dimensão da variável observada

- Medida correcta: considera que o valor correcto foi detectado e encontra-se na RV, como ilustrado na Figura 5.4(a);
- Demais medidas: interferências, ruídos na observação, medidas dispersas, etc, como indicado na Figura 5.4(b).

A elipse está centrada no valor estimado da medida e a sua forma e tamanho são determinadas pela inovação, sua covariância e pelo valor do limiar  $\gamma_n$ . Dentre todas as medidas “válidas”, um algoritmo ou técnica de associação de dados precisa actuar de forma a seleccionar, de uma maneira adequada, apenas a medida mais “correcta” (ou *more likely*) para assegurar consistência no filtro. Uma implicação relacionada ao seguimento de um único objecto é que as medidas indesejáveis, e que no entanto sejam válidas, constituem, em geral, uma interferência de carácter aleatório e precisam ser ignoradas. Este tipo de situação é comumente referenciado como seguimento de um único objecto em ambiente *clutter*.

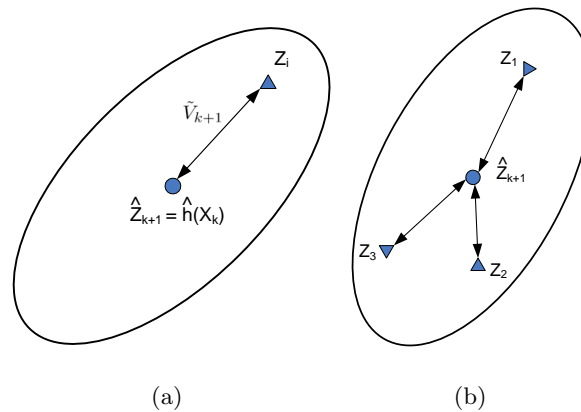


Figura 5.4: Região de validação em torno do valor estimado. (a) RV com apenas uma medida válida; (b) RV com múltiplas medidas válidas.

Apesar do seguimento de um único objecto em ambiente ruidoso parecer simples, as ambiguidades que podem ocorrer no processo de associação são as mesmas para o caso de múltiplos objectos (apesar de, neste último, serem mais complexas). Uma técnica, possivelmente a mais simples, que tem sido utilizada para minimizar as ambiguidades no processo de associação de dados é reduzir o limiar (ou dimensão da RV) de modo a diminuir a probabilidade de associação e assim evitar o seu no filtro de medidas indesejáveis. Outro método para gerir essas ambiguidades na associação é conhecido como NN (*Nearest Neighbour data association*). Este método é definido com base numa função de



verosimilhança. Dado um conjunto de observações válidas  $\mathbf{z} = z_1, \dots, z_n$  para uma RV definida pelo objecto-alvo  $\mathbf{x}_i$ , a verosimilhança para cada medida válida é calculada, no caso Gaussiano, por:

$$\Lambda_i = \frac{1}{(2\pi)^{n/2} |S_i|^{1/2}} \exp\left(-\frac{1}{2} v_i^T S_i^{-1} v_i\right) \quad (5.21)$$

onde  $n$  é a dimensão do vector da inovação. Uma outra métrica possível, é tomar o logaritmo de (5.21):

$$\ln \Lambda_i \approx v_i^T S_i^{-1} v_i + \ln |S_i| \quad (5.22)$$

A associação do tipo NN selecciona a observação que maximiza o valor de (5.21), ou minimiza (5.22). Em ambiente cuja densidade de interferências é elevada, a técnica NN mostra-se pouco eficiente (Kirubarajan & Bar-Shalom, 2004). Um método que incorpora a informação de todas as medidas validadas para reduzir as ambiguidades é o filtro PDA (Fortmann, Bar-Shalom, & Scheffe, 1983). Também conhecido como *all-neighbours*, o filtro PDA combina, de uma forma ponderada, a informação de todas as medidas validadas  $\mathbf{z}$  de modo a propagar as incertezas como uma “média” ponderada de cada observação validada individualmente, i.e. trata-se de propagar uma combinação (mistura) da PDF de cada medida válida. Esse método será discutido com mais detalhe na secção 5.3.5.

#### 5.2.4 Seguimento de múltiplos objectos

A tarefa de seguimento de múltiplos objectos-alvos num ambiente *clutter* (com elevada densidade de objectos, ruído de fundo e perturbações) pode ser significativamente complicada. Considerando o exemplo na Figura 5.5, que represente vários objectos a serem seguidos numa vizinhança, as considerações possíveis são:

- A medida  $z_1$  pode ter sido originada do objecto 1 ou de algum “falso” objecto;

- A medida  $z_2$  pode estar relacionada ao objecto 1 ou 2 ou a alguma “interferência”;
- As medidas  $z_3$  e  $z_4$  seguem as mesmas considerações de  $z_1$  relativamente ao objecto 2.

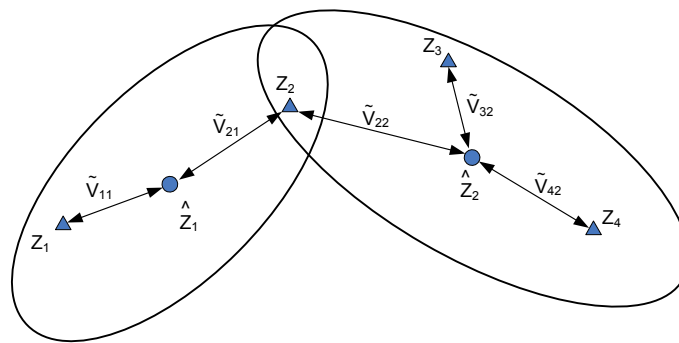


Figura 5.5: Região de validação para o caso de dois objectos na mesma vizinhança

Este simples exemplo ilustra a interdependência do processo de associação de dados quando há interferência persistente e interferência aleatória. Neste contexto, a interferência aleatória está relacionada a “falsos objectos” (ruídos, perturbações, etc) e a persistente relacionada a objectos “verdadeiros”, de interesse, mas que estão relativamente próximos uns dos outros e cujas RV se intersectam. A associação de medidas num ambiente onde há vários objectos-alvo próximos precisa ser realizada considerando-se todos os objectos simultaneamente, isto porque caso uma medida tenha sido considerada como tendo origem em mais que um objecto, o cálculo da probabilidade de associação torna-se altamente correlacionado e esses objectos não podem ser então considerados separadamente. Um filtro JPDA (Joint Probabilistic Data Association) (Fortmann et al., 1983), ou um filtro de Múltiplas Hipóteses (MHF), entre outros, são algumas das técnicas plausíveis para solucionar esse problema.

### 5.3 Seguimento de objectos usando dados do LRF

O seguimento de objectos com base nas medidas de um sensor LRF pode ser efectuado com os dados laser na forma polar ou Cartesiana. Neste trabalho o seguimento é realizado no referencial Cartesiano, motivo pelo qual os dados fornecidos pelo LRF (na forma polar) são convertidos para a forma Cartesiana. Os erros de medida do LRF são de natureza:

- angular: devido à rotação e sincronismo do sistema mecânico, e depende também do tamanho (diâmetro) do pulso laser;
- radial: dependente da distância ao objecto e da precisão do sistema opto-electrónico do sensor. Em geral este erro é caracterizado por um desvio-padrão relativamente constante e proporcional à distância entre o sensor e os objectos detectados.

É fácil perceber que, quanto maior a distância entre o sensor laser e o objecto, mais a incerteza dependerá da resolução angular. Num caso prático, um objecto a 70 metros de distância a ser detectado pelo sensor, a trabalhar com uma resolução angular de  $0.5^\circ$ , terá uma incerteza na distância de aproximadamente 25 mm e em termos angulares pode ser de até 300 mm. Uma vez detectado um objecto, interessa-nos estimar a sua posição com base nas medições do sensor LRF, estimar a sua velocidade e, dependendo do caso, a sua aceleração (dinâmicas de maior ordem) a cada instante de amostragem, i.e. de uma maneira *on-line*. O propósito principal no seguimento de um objecto é justificado em dois pontos:

1. Estabelecer a evolução no tempo dos objectos detectados e estimar a dinâmica de cada objecto separadamente;
2. Prever o comportamento “futuro” dos objectos (predição);

De maneira geral, o seguimento é um processo de três passos fundamentais:

1. Predição: prediz o novo estado (posição e velocidade por exemplo) do objecto baseado no modelo e no histórico probabilístico (matrizes de covariância);
2. Associação: no caso mais usual, associa uma observação a cada objecto sob seguimento. Na verdade este ciclo de associação não faz parte da maioria dos métodos de predição e estimação, sendo necessário adicionar esta etapa no algoritmo;
3. Estimação: estima o novo estado do objecto baseado na última observação disponível.

### 5.3.1 Pressupostos para os filtros implementados

Nesta dissertação, considera-se que o modelo dinâmico do processo (cinemática dos objectos detectados) é descrito pela expressão:

$$\mathbf{x}(k+1) = \mathbf{f}[\mathbf{x}(k)] + \nu(k) \quad (5.23)$$

onde, para simplicidade, foi assumido que a entrada de controlo é nula e o sistema é excitado (governado) por ruído branco de média nula:

$$\begin{aligned} E[\nu(k)] &= 0 \\ E[\nu(k)\nu(\tau)^T] &= Q_{(\tau)}\delta_{k\tau} \end{aligned} \quad (5.24)$$

O modelo de medida é expresso por:

$$\mathbf{z}(k) = \mathbf{h}[\mathbf{x}(k)] + \omega(k) \quad (5.25)$$

onde o ruído de medida é aditivo, branco e de média nula:

$$\begin{aligned} E[\omega(k)] &= 0 \\ E[\omega(k)\omega(\tau)^T] &= R_{(\tau)}\delta_{k\tau} \end{aligned} \quad (5.26)$$

As sequências que definem os ruídos do processo e das medições são assumidas como não-correlacionadas entre si, ou seja

$$E[\nu_{(k)}\omega_{(\tau)}^T] = 0 \quad \forall k, \tau \quad (5.27)$$

### 5.3.2 Filtro de Kalman

Largamente utilizado, o filtro de Kalman (KF) é um algoritmo do tipo predição e estimação baseado na forma recursiva do estimador LS. É aplicado a sistemas lineares e com incertezas do tipo Gaussiana. Basicamente o algoritmo KF (tal como a maioria dos filtros, e.g. PF, PDAF, EKF) é composto por um ciclo de predição e outro de correcção (filtração), de modo que as variáveis de estado são recursivamente estimadas sob um critério de minimização do erro quadrático médio.

### 5.3.3 Filtro de Kalman extendido

Sendo a equação do processo e ou de medição não-linear, e conseqüentemente a propagação da PDF variável, é necessário recorrer a outros algoritmos de estimação de estados. Em equações do processo ou das observações não-lineares, um método “não-óptimo” muito conhecido é o EKF. Considerando um sistema dinâmico descrito por (5.23), onde  $\mathbf{f}$  é do tipo não-linear, para se obter a predição do estado  $\hat{\mathbf{x}}(k+1|k)$ , a função não-linear  $\mathbf{f}$  é expandida numa série de Taylor, em torno de  $\hat{\mathbf{x}}(k|k)$ , e considera-se os elementos de primeira ou segunda ordem (EKF de 1ª ou 2ª ordem). As equações são assim “linearizadas” em torno do último estado estimado. O filtro EKF de segunda ordem envolve, além do Jacobiano, o cálculo do Hessiano, por isso é mais comum considerar o EKF de primeira ordem por ser menos complexo. Por ser um filtro de “aproximação” (não óptimo), as matrizes  $P(k|k)$  e  $P(k+1|k)$  não representam a covariância (verdadeira) do vector de estados estimados. Ao contrário do KF, as matrizes  $P$  e o ganho  $K$  do EKF não podem ser calculadas de maneira *off-line*, pois dependem de estados anteriores e das observações. Outra diferença

é que o EKF pode divergir, principalmente se as linearizações consecutivas não forem uma “boa” aproximação do modelo não-linear em todo o domínio de incertezas associadas. A não consideração de termos de ordem superior àquela definida pelo filtro EKF, introduz erros nas estimativas. Existem técnicas para minimizar os erros devidos à linearização, de que são exemplos os filtros *Iterated-EKF* (IEKF) e o filtro *Unscented-KF* (UKF) (Bar-Shalom & Fortmann, 1988).

#### 5.3.4 Filtro de Partículas

O PF é uma técnica para implementação de um filtro recursivo Bayesiano por meio do método de Simulação Monte Carlo, (Gustafsson et al., 2002), (Rekleitis, 2004) e (Arulampalam, Maskell, Gordon, & Clapp, 2001). Essa técnica tem como objectivo a representação da PDF por um conjunto aleatório de amostras e respectivos pesos associados (partículas). Dessa forma, o FP estima o vector de estados com base nessas partículas e nos respectivos pesos. Neste contexto, a abordagem Bayesiana é uma ferramenta que serve, de maneira geral, para a estimação de parâmetros (estados) de um sistema na forma de uma PDF, baseada em todas as informações disponíveis. Para o caso da estimação de estados para sistemas cujos modelos são lineares e Gaussianos (função de distribuição probabilística do ruído é Gaussiana), a PDF mantém-se Gaussiana e então um KF é uma das técnicas mais recomendadas. Em sistemas Gaussianos não-lineares, um EKF pode ser empregado com sucesso. Para o caso de sistemas não-lineares e com modelos não-Gaussianos, surgem algumas dificuldades:

- Como representar uma PDF geral usando uma capacidade computacional finita de armazenamento;
- Como realizar as integrações envolvidas no estágio de actualização da PDF quando novos dados são adquiridos (observados).

O formalismo básico do método que define um PF consiste em construir uma representação amostrada (*sample-based*) de toda a PDF que representa o estado do sistema

proposto. O algoritmo de um FP é recursivo por natureza, e opera em duas fases: predição e actualização. Há algumas técnicas para se estimar, com base nas partículas, a variável de interesse (estados), entre elas:

- A melhor partícula (aquela com peso máximo);
- Média ponderada;
- Média “robusta” com as partículas seleccionadas (define-se um limiar e seleccionam-se aquelas partículas cujo módulo da diferença entre os respectivos pesos e o peso máximo não seja superior a esse limiar pré-definido);
- Uso de todas as partículas.

A cada período de amostragem, cada partícula é modificada de acordo com o modelo do processo (estágio de predição), incluindo a adição de ruído aleatório para simular o efeito do ruído nas variáveis de estado do processo. Então, o peso de cada partícula é reavaliado com base na última informação sensorial (estágio de actualização). No entanto, há que realizar um processo de *resampling* daquelas partículas cujos pesos sejam “infinitesimalmente” pequenos. Isso significa que, após algumas iterações, ocorre uma perda de diversidade, ou seja, um determinado número de partículas acaba por possuir um peso “desprezível” no contributo da estimação da PDF. Ou seja, os pontos críticos na implementação de um FP são:

1. Definir o algoritmo para tratar do problema da degenerescência;
2. Definir o método para a re-amostragem das partículas;
3. Escolher a função da densidade de importância.

### **Vantagens**

- Habilidade em representar funções de densidade probabilística PDF arbitrárias (e.g. não-Gaussianas, multi-modais);

- Trabalha com ruídos não-Gaussianos;
- É viável em aplicações que incluem múltiplos modelos (seguimento de multi-alvos).

#### Desvantagens

- Certa dificuldade em determinar o número ótimo de partículas;
- Número de partículas aumenta proporcionalmente à dimensão do modelo;
- Problemas de degenerescência (*degeneracy*) e perda de diversidade (*loss of diversity*);
- A escolha da função de densidade de importância é crucial.

#### 5.3.5 Filtro PDAF

O filtro PDAF é um algoritmo Bayesiano sub-ótimo em que se assume que há apenas um objecto-alvo de interesse sob seguimento. A cada período de amostragem uma nova RV é criada e, entre as medidas “validadas”, uma precisa ser identificada como originada do objecto-alvo sob seguimento, e as demais são tratadas como “falsos alarmes”. No filtro PDAF, todas as “falsas” medidas (não associadas ao objecto de interesse) são consideradas como interferências aleatórias, com uma distribuição espacial uniforme. O algoritmo PDAF calcula uma associação probabilística para cada medida validada, i.e. que esteja dentro da RV, para o objecto-alvo de interesse no instante corrente em que a medição está disponível. Essa informação probabilística (Bayesiana) é utilizada no ciclo de correcção do PDAF.

#### Pressupostos:

- há apenas um objecto-alvo de interesse (múltiplos objectos devem ser tratados com o filtro *Joint-PDAF*);
- a informação passada do objecto é sumariada sob a condição de uma densidade probabilística condicional do tipo Gaussiana e depende, aproximadamente, do valor do



estado no instante corrente, do valor do estado estimado (no instante anterior) e da PDF *a priori*;

- a cada período de amostragem uma RV é construída;
- dentre as medidas validadas, pelo menos uma delas é considerada como sendo efetivamente originada do objecto;
- as medidas validadas restantes são consideradas como “falsos alarmes” (ruídos, perturbações) e são modeladas como variáveis independentes e identicamente distribuídas (i.i.d.) aleatórias e com distribuição espacial uniforme;
- a detecção do objecto ocorre independentemente ao longo do tempo com uma probabilidade conhecida  $P_D$ .

O filtro PDAF usa uma decomposição, para estimar a variável de estado, com respeito à origem de cada elemento do conjunto de medidas validadas:

$$Z(k) \triangleq z_i(k)_{i=1}^{m_k} \quad (5.28)$$

ou seja, o conjunto de  $z_i(k)$  medidas no instante  $k$  é  $Z(k)$ , onde  $m_k$  (variável aleatória) é o número de medidas dentro da região (i.e. validadas) no instante de tempo  $k$ . A informação do “passado” do filtro é sumariada sob a seguinte consideração:

$$p[x(k)|Z^{k-1}] = \mathcal{N}[x(k); \hat{x}(k|k-1), P(k|k-1)] \quad (5.29)$$

Para cada medida uma RV é construída, tal como em (5.19), sendo o volume da RV calculado por:

$$V(k) = c_{nz} |\gamma S(k)|^{1/2} = c_{nz} \gamma^{\frac{n_z}{2}} |S(k)|^{1/2} \quad (5.30)$$

sendo que o coeficiente  $c_{nz}$  depende da dimensão da medida. Isto é o volume de uma hipersfera unitária de dimensão  $n_z$ , tal que:  $c_1 = 2$ ,  $c_2 = \pi$ ,  $c_3 = 4\pi/3$ , etc. O filtro PDAF decompõe a estimação com respeito à origem de cada elemento do último conjunto de medidas válidas (5.28). Sabendo que  $Z(k)$  contém as medidas que estão dentro da região elíptica de validação  $\tilde{V}_k(\gamma)$ , define-se os eventos:

$\theta_i(k) \triangleq z_i(k)$ ,  $i = 1, \dots, m_k$ ; sendo  $z_i(k)$  as medidas originadas do objecto;

$\theta_0(k) \triangleq \{ \text{nenhuma medida no instante } k \text{ é originada do objecto/alvo} \}$

cujas probabilidades estão condicionadas a  $Z(k)$ :

$$\beta_i(k) \triangleq P\{\theta_i(k)|Z(k)\}, i = 0, 1, \dots, m_k \quad (5.31)$$

Assumindo que esses eventos são mutuamente exclusivos, considerando o teorema de probabilidade total  $\sum_{i=0}^{m_k} \beta_i(k) = 1$ , e ainda tendo em conta outras considerações e manipulações matemáticas (Kirubarajan & Bar-Shalom, 2004), tem-se que a equação de actualização de estados do Filtro PDAF é dada por:

$$\hat{x}(k|k) = \hat{x}(k|k-1) + K(k)v_C(k) \quad (5.32)$$

onde  $K$  é o ganho do filtro, e  $v_C(k)$  é chamada inovação combinada:

$$v_C(k) \triangleq \sum_{i=0}^{m_k} \beta_i(k)v_i(k) \quad (5.33)$$

sendo a inovação  $v_i(k)$ , de cada medida validada, calculada por:

$$v_i(k) = z_i(k) - \hat{z}_i(k|k-1) \quad (5.34)$$

É de notar que (5.32) é não-linear pois a inovação combinada  $v_C(k)$  depende das probabilidades  $\beta_i(k)$ , que por sua vez dependem das observações. A covariância associada

com o estágio de actualização de estado é:

$$P(k|k) = \beta_0(k)P(k|k-1) + [1 - \beta_0(k)]P^C(k|k) + \tilde{P}(k) \quad (5.35)$$

onde

$$\tilde{P}(k) = K(k) \left[ \sum_{i=1}^{m_k} \beta_i(k) v_i(k) v_i^T(k) - v_C(k) v_C^T(k) \right] K^T(k) \quad (5.36)$$

e

$$P^C(k|k) \triangleq [I - K(k)H(k)]P(k|k-1) \quad (5.37)$$

é a covariância de actualização de estados com a medida “correcta”, i.e. sem as medidas “incertas”. De forma diferente a que ocorre no KF, onde a equação de covariância é independente das medidas (valores observados  $z_i(k)$ ), a precisão na estimação da covariância de estados (5.35) depende das medições. Essa característica é típica de filtros não-lineares. As demais equações do filtro PDAF são similares às de um KF. O Filtro PDAF é uma opção interessante a ser utilizada quando há mais de um sensor a medir uma mesma variável de estado sob seguimento pois, de um modo probabilístico e pesado, estima o estado com base em todas as medidas disponíveis e validadas.

### 5.3.6 Consistência

A distribuição Chi-quadrado é muitas vezes usada para a verificação da consistência dos estimadores de estado, i.e., se os erros considerados no modelo das variáveis de estado estão consistentes com as variâncias calculadas pelo estimador. A informação sobre o estado dum processo está contida no seu valor estimado  $\hat{\mathbf{x}}(k|k)$  num instante  $k$  e na respectiva matriz de covariância associada  $\mathbf{P}(k|k)$  ou seja, na prática não se conhece com rigor o estado real

dum processo, mas o seu valor estimado e a incerteza a ele associada. O fenómeno da divergência num estimador pode ter como causas:

- Incertezas no modelo;
- Erros numéricos;

No caso do KF, EKF ou PDAF, devido ao facto do ganho ( $K$ ) do filtro ser baseado na covariância do erro de estado, então a consistência é necessária para manter o filtro coerente, i.e., erros no cálculo das matrizes de covariância induzem a erros nos ganhos do filtro. Algumas técnicas para a verificação da consistência de estimadores estão descritas em (Bar-Shalom & Li, 1995).

De modo a minimizar os erros numéricos no cálculo da matriz de covariância do estado de um sistema linear, pode-se adoptar uma das seguintes expressões matemáticas durante o ciclo de correcção do filtro:

$$P(k+1|k+1) = [I - K(k+1)H(k+1)]P(k+1|k) \quad (5.38)$$

$$P(k+1|k+1) = P(k+1|k) - K(k+1)S(k+1)K^T(k+1) \quad (5.39)$$

$$P(k+1|k+1) = [I - K(k+1)H(k+1)]P(k+1|k)[I - K(k+1)H(k+1)]^T + K(k+1)R(k+1)K(k+1)^T \quad (5.40)$$

Ou seja, de forma a preservar as duas características da matriz de covariância, simetria e a propriedade de ser positiva definida, convém analisar as desvantagens de cada equação apresentada acima. A equação (5.38) é muito sensível a erros de aproximação, induzindo a perdas na simetria e  $P(k+1|k+1)$  tende a deixar de ser positiva definida. A equação (5.39) tende a manter a simetria mas, a covariância pode deixar de ser positiva definida.

Já a equação (5.40), conhecida como covariância na forma *Joseph*, é muito menos sensível aos erros de aproximação e, se a implementação do produto das três matrizes for adequada, essa expressão tende a preservar as propriedades da matriz de covariância, apesar de exigir um esforço computacional maior (Bar-Shalom & Fortmann, 1988).

## 5.4 Cooperação sensorial

Em muitas aplicações, em especial naquelas levadas a cabo em ambientes exteriores, a utilização de câmaras para detecção de objectos pode ser uma tarefa extremamente complicada, devido aos seguintes factores: inconstância na qualidade de iluminação, condições climatéricas, ruídos e trepidações, etc. A utilização da cooperação sensorial (Wijesoma et al., 2002) ou fusão (Neira et al., 1999) com os dados de um sensor laser e de uma câmara tem como objectivo primordial, neste trabalho, definir a posição da região de interesse (ROI) no plano imagem e desse modo facilitar o processo de detecção dos objectos na imagem. Essa cooperação sensorial foi implementada sob o pressuposto que o sensor laser e a câmara estejam montados num suporte mecânico fixo e que o referencial da câmara  $\{C\}$  e do laser  $\{L\}$  estejam “aproximadamente” alinhados.

Um ponto-objecto no sistema de coordenadas do referencial externo  $\{E\}$  (mundo)  $(X, Y, Z)^T$  é detectado pelo sensor laser em coordenadas polares  $(r, \theta)$ , cuja transformação para o sistema de coordenadas Cartesianas do laser  $({}^L X, {}^L Y, {}^L Z)^T$  é obtido directamente para duas dimensões:

$$\begin{cases} Z = {}^L Y = r \sin(\theta) \\ X = {}^L X = r \cos(\theta) \end{cases} \quad (5.41)$$

A equação (5.41) tem como pressuposto que o ângulo *pitch* ( $\alpha_L$ ) é igual a  $0^\circ$ , i.e. o plano de base do laser está paralelo ao solo. Isso permite conhecer a variável  $Y$ : igual à altura do laser em relação ao solo  ${}^L Z$ . Interessa-nos obter uma transformação de coordenadas que relacione pontos no referencial do laser  $\{L\}$  com o referencial da câmara  $\{C\}$  e, por

meio dos parâmetros intrínsecos, obter as coordenadas do ponto no plano imagem  $\{I\}$ . Considerando o ponto no referencial Cartesiano da câmara  $({}^cX, {}^cY, {}^cZ)^T$ , a relação de transformação entre um ponto-objecto no plano laser  $\{L\}$  e o referencial da câmara  $\{C\}$  é obtido pela expressão:

$$\begin{bmatrix} {}^cX \\ {}^cY \\ {}^cZ \\ 1 \end{bmatrix} = {}^C M_L = \begin{bmatrix} {}^lX \\ {}^lY \\ {}^lZ \\ 1 \end{bmatrix}; {}^C M_L = [ {}^C R_L ] [ {}^C T_L ] = \begin{bmatrix} r_{11} & r_{12} & r_{13} & T_1 \\ r_{21} & r_{22} & r_{23} & T_2 \\ r_{31} & r_{32} & r_{33} & T_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5.42)$$

Sendo que cada elemento da matriz de rotação é dado por:

$$\left\{ \begin{array}{l} r_{11} = \cos \beta \cos \psi \\ r_{12} = \cos \beta \sin \psi \\ r_{13} = \sin \beta \\ r_{21} = \sin \alpha \sin \beta \cos \psi - \cos \alpha \sin \psi \\ r_{22} = \sin \alpha \sin \beta \sin \psi + \cos \alpha \cos \psi \\ r_{23} = \sin \alpha \cos \beta \\ r_{31} = \cos \alpha \sin \beta \cos \psi + \sin \alpha \sin \psi \\ r_{32} = \cos \alpha \sin \beta \sin \psi - \sin \alpha \cos \psi \\ r_{33} = \cos \alpha \cos \beta \end{array} \right. \quad (5.43)$$

Com o pressuposto que o referencial  $\{L\}$  e  $\{C\}$  estão alinhados, isso significa que matriz de rotação  ${}^C R_L$  é igual à matriz de identidade, ou seja:

1.  ${}^cZ$  e  ${}^lY$  estão paralelos: o ângulo de inclinação *pitch* é nulo ( $\beta=0$ );
2. O ângulo entre  ${}^cX$  e  ${}^lX$  é nulo ( $\alpha=0$ ): não há variação no ângulo *yaw*;
3.  ${}^lZ$  e  ${}^cY$  estão paralelos: o ângulo *roll* é nulo ( $\psi=0$ ) entre o laser e a câmara;

Todos os três pontos são considerados pressupostos “ideais” que, na prática, tendem a ser coerentes para objectos que estejam relativamente próximos de ambos os sensores. Para minimizar os erros destes pressupostos, o processo de “calibração” entre os sensores foi realizado com várias imagens tomadas em distâncias que variaram de 1 a 4 metros. Isso obrigou à utilização de um alvo de calibração (tudo xadrez) relativamente grande:  $729 \times 567$  mm. Basicamente, o processo de obtenção da relação de transformação de coordenadas entre  $\{L\}$  e  $\{C\}$  segue os seguintes passos:

1. Tomar as imagens, tanto da câmara quanto do laser, do alvo de calibração para diferentes distâncias e posições;
2. Usar as imagens da câmara e obter os parâmetros intrínsecos e extrínsecos (Strobl, Sepp, Fuchs, Paredes, & Arbter, 2005);
3. Com as leituras do laser (distâncias) e o valor estimado da posição do alvo no referencial da câmara, utilizar um algoritmo do tipo LS e encontrar a matriz de translação<sup>1</sup> entre os dois sensores;
4. Na prática é conveniente que os valores da matriz  ${}^C T_L$  se adaptem, variem, consoante a distância entre os objectos e os sensores.

Ou seja, ao final têm-se os valores estimados de  $T_1, T_2, T_3$ , os parâmetros extrínsecos e intrínsecos da câmara e as incertezas (desvios padrão) dessas variáveis; valores esses que variam consoante o aumento da distância entre o objecto e os sensores.

---

<sup>1</sup>Devido às restrições impostas ao modelo de transformação de coordenadas, a matriz de rotação  ${}^C R_L$  é uma matriz identidade

## Capítulo 6

# Classificador-GMM usando dados laser

### Conteúdo

---

6.1	Função densidade probabilística . . . . .	78
6.2	Modelos de misturas Gaussianas (GMM) . . . . .	78
6.3	Máxima verosimilhança (ML) . . . . .	80
6.4	Breve descrição da regra de Bayes . . . . .	81
6.5	Algoritmo EM . . . . .	82
6.6	Classificador GMM . . . . .	85
6.7	Combinação probabilística de classificadores . . . . .	88

---

**E**STE capítulo apresenta sucintamente conceitos básicos ao projecto de um classificador GMM com regra de decisão do tipo ML e MAP. Serão abordados a função densidade probabilística com misturas Gaussianas, o algoritmo EM, os modelos GMM, e o cálculo do ML e do MAP para este tipo de classificador.



## 6.1 Função densidade probabilística

A função densidade probabilística (PDF) de uma variável escalar aleatória  $x$ , que assume valores contínuos, num dado instante  $x = \xi$  é definida como:

$$p_x(\xi) = \lim_{d\xi \rightarrow \infty} \frac{P\{\xi d\xi < x \leq \xi\}}{d\xi} \geq 0 \quad (6.1)$$

sendo  $P\{\cdot\}$  a probabilidade de ocorrência do evento  $\{\cdot\}$ . A PDF de uma variável unidimensional que segue uma distribuição Gaussiana, ou PDF Normal, é caracterizada pela curva geométrica com aparência de “sino” e é definida por dois parâmetros, ou momentos, a média  $\mu$  e a variância  $\sigma^2$ . No caso de uma variável multidimensional ( $\mathbf{x} \in \mathbb{R}^D$ ), a PDF Gaussiana é definida em termos matriciais,

$$\mathcal{N}(\mathbf{x}; \mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^D |\Sigma|}} \exp\left[-\frac{1}{2}(\mathbf{x} - \mu)^T \Sigma^{-1}(\mathbf{x} - \mu)\right] \quad (6.2)$$

onde  $\mu$  é o vector de médias e  $\Sigma$  é a matriz de covariância.

A distribuição Gaussiana é uma função com caracterização matemática largamente conhecida, completamente definida por dois parâmetros, e com propriedades que facilitam o tratamento matemático mesmo em múltiplas dimensões. No contexto da classificação, em certas situações é suficiente um modelo unimodal para a caracterização das classes de interesse mas, em geral, na prática os modelos são multi-modais e a utilização de uma única Gaussiana não é suficiente para representar a classe.

## 6.2 Modelos de misturas Gaussianas (GMM)

Um modelo do tipo GMM é definido como uma combinação (Figura 6.1) de várias distribuições Gaussianas e pode representar modelos mais precisos de uma dada classe de interesse. A PDF é definida por uma soma ponderada de  $M$  componentes de distribuições

Gaussianas:

$$p(x; \Theta) = \sum_{m=1}^M \alpha_m \mathcal{N}(x; \mu, \Sigma) \quad (6.3)$$

onde  $\alpha_m$  é o peso da componente  $m$ , tal que  $0 < \alpha_m < 1$  para todos os componentes, e  $\sum_{m=1}^M \alpha_m = 1$ . O vector de parâmetros que define completamente a função densidade probabilística de uma GMM particular é dada por:

$$\Theta = \{\alpha_1, \mu_1, \Sigma_1, \alpha_2, \mu_2, \Sigma_2, \dots, \alpha_M, \mu_M, \Sigma_M\} \quad (6.4)$$

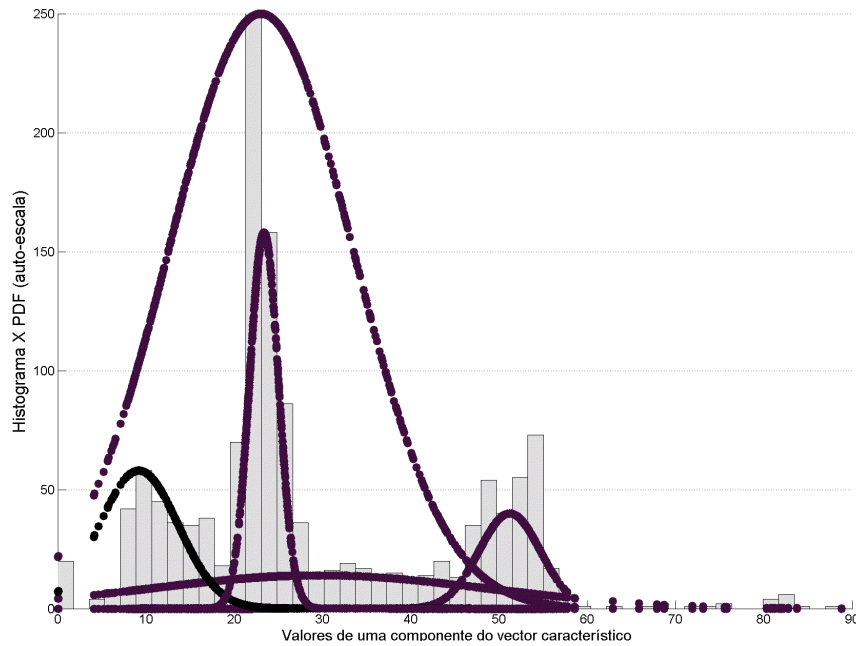


Figura 6.1: Exemplo de um modelo multi-modal, com um conjunto de componentes Gaussianas, para uma característica de uma das classes de interesse.

A estimação dos parâmetros GMM de uma classe pode ser considerado como um problema de classificação do tipo aprendizagem não-supervisionada. Ou seja, considera-se que as amostras são oriundas de componentes individuais da distribuição multi-modal mas, neste caso, não há o conhecimento prévio de qual componente da distribuição foi originada

cada amostra dos dados de treino. Na verdade, o processo de estimação dos parâmetros  $\Theta$  que caracterizam o modelo GMM é uma aproximação da distribuição que melhor caracteriza as amostras utilizadas na fase de aprendizagem. Neste trabalho utiliza-se o algoritmo EM para estimar os componentes que definem o modelo GMM de cada classe de objecto a ser classificado. Antes de descrever o algoritmo EM é conveniente abordar o conceito de máxima verosimilhança.

### 6.3 Máxima verosimilhança (ML)

Considere-se um conjunto de  $N$  vectores de amostras obtidos para cada classe de modo supervisionado, i.e.  $\mathcal{X} = \mathbf{x}_1, \dots, \mathbf{x}_N$ , em que cada vector é caracterizado por uma função densidade probabilística  $p(\mathbf{x}_i|\Theta)$ , e assumindo que esses vectores de dados são i.i.d. para cada classe, a densidade probabilística resultante para esse conjunto de amostras é:

$$p(\mathcal{X}|\Theta) = \prod_{i=1}^N p(\mathbf{x}_i|\Theta) = \mathcal{L}(\Theta|\mathcal{X}) \quad (6.5)$$

$\mathcal{L}(\Theta|\mathcal{X})$  é chamada função verosimilhança (*likelihood*) dos parâmetros que caracterizam as amostras. O problema de maximização da verosimilhança traduz-se na determinação do conjunto de parâmetros  $\Theta$  que maximizam  $\mathcal{L}$ . Ou seja, a solução passa por encontrar o valor  $ML$  tal que:

$$ML = \arg \max_{\Theta} \mathcal{L}(\Theta|\mathcal{X}) \quad (6.6)$$

É comum que as técnicas de maximização da verosimilhança utilizem a função logarítmica  $\ln(\mathcal{L}(\Theta|\mathcal{X}))$ , porque trás grandes vantagens. O esforço computacional é favorável pois o produto em (6.5) transforma-se num somatório:

$$\ln(p(\mathcal{X}|\Theta)) = \ln\left(\prod_{i=1}^N p(\mathbf{x}_i|\Theta)\right) = \sum_{i=1}^N \ln(p(\mathbf{x}_i|\Theta)) \quad (6.7)$$

Outra vantagem está no cálculo das componentes do somatório (6.7) que, por ser do tipo Gaussiano (6.2), evita-se o cálculo da função exponencial, resultando em:

$$\ln(p(\mathbf{x}_i|\Theta)) = -\frac{d}{2}\ln(2\pi) - \frac{1}{2}\ln(|\Sigma|) - \frac{1}{2}(\mathbf{x}_i - \mu)^T \Sigma^{-1}(\mathbf{x}_i - \mu) \quad (6.8)$$

Além disso, como a função logarítmica é monotonamente crescente, o valor logaritmo da verosimilhança tem a mesma relação de ordem que a verosimilhança:

$$p(\mathbf{x}|\Theta_1) > p(\mathbf{x}|\Theta_2) \iff \ln(p(\mathbf{x}|\Theta_1)) > \ln(p(\mathbf{x}|\Theta_2)) \quad (6.9)$$

Esta última propriedade permite usar o valor de ln-verosimilhança directamente como uma condição suficiente para se classificar as classes envolvidas no problema. A dificuldade no cálculo da *ML* é directamente proporcional à complexidade das distribuições  $p(\mathbf{x}_i|\Theta)$ . No nosso caso as PDF são misturas de Gaussianas, cujos parâmetros são estimados pelo algoritmo EM.

## 6.4 Breve descrição da regra de Bayes

Considerando as categorias (classes) de interesse designadas por  $q_i$ ,  $i = 1, 2, \dots, N_C$ , onde  $N_C$  é o número total de classes, caracterizadas por modelos GMM cujos parâmetros são  $\Theta_i$ , e sabendo que  $X$  representa o vector de características utilizado na tarefa de classificação, as seguintes funções probabilísticas serão utilizadas:

- Probabilidade *a priori*  $P(q_i, \Theta_i)$ : reflecte o conhecimento que se tem sobre a probabilidade de uma dada classe ( $q_i$ ) ser verdadeira sem no entanto ter sido observada ou realizada;
- Função densidade probabilística  $p(X|q_i, \Theta_i)$ : condicionada à classe, ou ao modelo que caracteriza a classe, também chamada de likelihood. Esta função caracteriza a probabilidade da característica observada sendo a classe (ou modelo) conhecida.

- Probabilidade *a posteriori*  $p(q_i|X, \Theta_i)$ : a probabilidade da classe  $q_i$  dada a observação de  $X$  e conhecida  $P(q_i, \Theta_i)$ ;

Neste trabalho assume-se que a PDF de cada classe tem uma forma paramétrica conhecida, e que é determinada unicamente por este conjunto (vector) de parâmetros  $\Theta_i$ . Um dos grandes desafios é determinar, ou conhecer, exactamente qual a expressão que define a função densidade para cada classe. É comum, e extremamente prático, recorrer a funções normais multi-variáveis ou de densidade Gaussiana. No classificador-GMM as funções de densidade são consideradas misturas de PDF do tipo Normal, já discutidas na secção 6.1. Sabendo ainda que o vector de características  $X$  é do tipo discreto, as funções densidade probabilística  $p(X|q_i, \Theta_i)$  tornam-se singulares (assumem valores discretos); neste contexto, a regra de Bayes envolve probabilidades em vez de densidades probabilísticas (Duda. et al., 2001), cuja expressão, para cada classe, é dada por:

$$P(q_i|X, \Theta_i) = \frac{P(X|q_i, \Theta_i)P(q_i, \Theta_i)}{P(X)} \quad (6.10)$$

sendo  $P(X) = \sum_{i=1}^{N_c} P(X|q_i, \Theta_i)P(q_i, \Theta_i)$  um factor de normalização designado “evidência”, e  $P(q_i|X, \Theta_i)$  é a probabilidade *a posteriori*.

## 6.5 Algoritmo EM

O algoritmo EM é um método iterativo usado na estimação de parâmetros, otimizando-se o valor máximo da verosimilhança (ML), ou o valor ln-ML, com base nas amostras, mesmo que incompletas, de um dado conjunto de amostras que seguem uma dada distribuição (e.g. do tipo Gaussiana). Este método também pode ser utilizado em casos onde uma abordagem analítica para o cálculo do valor máximo da verosimilhança é inviável, como por exemplo em modelos de misturas Gaussianas (GMM), ou na estimação dos parâmetros de um modelo oculto de *Markov* (Bilmes, 1997), (McLachlan & Krishnan, 1997).

O algoritmo EM, aqui apresentado, para a estimação dos parâmetros de uma mistura

Gaussiana, é baseado no trabalho de (Paalanen, Kamarainen, Ilonen, & Kälviäinen, 2005). Suponha que o conjunto de dados característicos conhecidos  $\mathcal{X}$  é interpretado como um conjunto de dados de amostras incompletas. A parte desconhecida é dada por  $\mathcal{Y}$  e reflecte o “conhecimento” de qual componentes, da distribuição Gaussiana, produziram cada amostras  $x_n$ . Para cada  $x_n$  há um vector binário  $\mathbf{y}_n = y_{n,1}, \dots, y_{n,c}$ , onde  $y_{n,c} = 1$ , se a amostra em questão foi produzida pela componente  $c$ , ou igual a zero caso contrário. O logaritmo da verosimilhança completa dos dados é dado por:

$$\ln \mathcal{L}(\mathcal{X}, \mathcal{Y}; \Theta) = \sum_{n=1}^N \sum_{c=1}^C y_{n,c} \ln(\alpha_c p(x_n|c; \Theta)) \quad (6.11)$$

O *E-step* é utilizado para calcular o valor condicional da esperança da ln-verosimilhança do conjunto de dados completos (função-Q), tendo  $\mathcal{X}$  e o valor corrente dos parâmetros estimados  $\Theta^i$ . Como o valor ln-verosimilhança dos dados completos  $\ln \mathcal{L}(\mathcal{X}, \mathcal{Y}; \Theta)$  é linear com respeito às amostras desconhecidas  $\mathcal{Y}$ , a esperança condicional  $\mathcal{W} \equiv E[\mathcal{Y}|\mathcal{X}, \Theta]$  precisa simplesmente ser calculada e inserida na expressão (6.11). Consequentemente:

$$Q(\Theta, \Theta^i) \triangleq E[\ln \mathcal{L}(\mathcal{X}, \mathcal{Y}; \Theta)|\mathcal{X}, \Theta^i] = \ln \mathcal{L}(\mathcal{X}, \mathcal{W}; \Theta) \quad (6.12)$$

onde os elementos de  $\mathcal{W}$  são definidos como:

$$\omega_{n,c} \triangleq E[y_{n,c}|\mathcal{X}, \Theta^i] = Pr[y_{n,c} = 1|x_n, \Theta^i] \quad (6.13)$$

A probabilidade pode ser calculada por meio da lei de Bayes:

$$\omega_{n,c} = \frac{\alpha_c^i p(x_n|c; \Theta^i)}{\sum_{j=1}^C \alpha_j^i p(x_n|j; \Theta^i)} \quad (6.14)$$

onde  $\alpha_c^i$  é a probabilidade *a priori* (da estimativa  $\Theta^i$ ), e  $\omega_{n,c}$  é a probabilidade *a posteriori* de  $y_{n,c} = 1$ , após a observação de  $x_n$ . Por outras palavras,  $\omega_{n,c}$  é a probabilidade de  $x_n$  ter sido gerada pela componente  $c$  (Paalanen et al., 2005).

Aplicando o *M-step* para o problema da estimação da distribuição dos parâmetros para a componente  $c$  da mistura Gaussiana com matrizes de covariâncias arbitrárias, as expressões para a iteração resultante são dadas por:

$$\alpha_c^{i+1} = \frac{1}{N} \sum_{n=1}^N \omega_{n,c} \quad (6.15)$$

$$\mu_c^{i+1} = \frac{\sum_{n=1}^N x_n \omega_{n,c}}{\sum_{n=1}^N \omega_{n,c}} \quad (6.16)$$

$$\Sigma_c^{i+1} = \frac{\sum_{n=1}^N \omega_{n,c} (x_n - \mu_c^{i+1})(x_n - \mu_c^{i+1})^T}{\sum_{n=1}^N \omega_{n,c}} \quad (6.17)$$

As novas estimativas são agrupadas no vector de parâmetros  $\Theta^{i+1}$ . Se os critérios de convergência não são satisfeitos (Paalanen et al., 2005), então  $i \leftarrow i + 1$  e o conjuntos de equações (6.14), (6.15), (6.16) e (6.17) são calculadas novamente com as novas estimativas.

Para se interpretar o método exposto no conjunto de equações (6.15 - 6.17), deve-se adoptar o raciocínio a seguir. O peso  $\alpha_c$  de uma componente é a porção de amostras pertencentes àquela amostra. Esta componente é calculada pela aproximação da componente condicional da PDF com o valor do parâmetro estimado anteriormente e tomando a probabilidade *a posterior* de cada ponto-amostra pertencente à componente  $c$  (6.14). O vector de médias  $\mu_c$  e a matriz de covariância  $\Sigma_c$  são estimadas de maneira similar. As amostras são “ponderadas” com as suas probabilidades de pertencerem a uma dada componente, e então as respectivas média e covariância são calculadas.

A abordagem apresentada assume que o número  $C$  de componentes Gaussianas é conhecido, ou pré-estabelecido. Apesar de que algumas técnicas também estimam esse parâmetro, o esforço computacional é aliviado se o utilizador estabelecer esse valor como condição inicial ao algoritmo.

## 6.6 Classificador GMM

No classificador GMM a probabilidade condicional (verossimilhança ou *likelihood*) é calculada como uma combinação linear de  $M$  PDF, onde cada PDF é caracterizada por um vector de parâmetros Gaussianos  $\theta_m$  e por um peso  $\alpha_m$  ( $\sum_{m=1}^M \alpha_m = 1$ ), i.e. cada classe  $q_i$  tem um conjunto de parâmetros  $\Theta_i = (\alpha_1, \theta_1, \dots, \alpha_M, \theta_M)$ , tal que  $\theta_m = [\mu_m, \Sigma_m]$ , sendo  $\mu_m$  um vector de médias,  $\Sigma_m$  a matriz de covariância e  $\alpha_m$  é o peso de cada distribuição Gaussiana que caracteriza a sua probabilidade condicional. A probabilidade condicional de cada classe tem a seguinte expressão:

$$P(X|q_i, \Theta_i) = \sum_{m=1}^M \alpha_m P(X|q_i, \theta_m) \quad (6.18)$$

O classificador GMM desenvolvido neste trabalho utiliza duas regras de decisão, a primeira baseada no valor máximo da verossimilhança e a segunda baseada na probabilidade *a posteriori*. Para cada instante de tempo  $k$ , presumindo que o sistema terá uma observação corrente  $X(k)$ , as regras são implementadas da seguinte forma:

1. Máxima verossimilhança:  $X(k) \in q_i \text{ SE } P(X(k)|q_i, \Theta_i) = \max[P(X(k)|q_u, \Theta_u)]$
2. Máxima a posteriori:  $X(k) \in q_i \text{ SE } P(q_i|X(k), \Theta_i) = \max[P(q_u|X(k), \Theta_u)]$

onde  $u = 1, 2, \dots, N_C$ , e  $N_C$  é o número de classes.

Para o caso da regra ML, utiliza-se o Algoritmo 2:

Para o caso da regra MAP, o algoritmo precisa incorporar o valor da probabilidade *a priori*  $P(q_i, \Theta_i)$  e da probabilidade condicional  $P(X(k)|q_i, \Theta_i)$  para se calcular o valor da probabilidade *a posteriori*  $P(q_i|X, \Theta_i)$ . Neste trabalho optou-se por utilizar uma técnica “dinâmica” no cálculo da *a priori*, i.e. em vez de se utilizar uma *a priori* com valor constante para cada classe (caso mais simples), utiliza-se como probabilidade *a priori* (num dado instante  $k$ ) o valor da probabilidade *a posteriori* calculado no instante anterior  $k - 1$ . O Algoritmo 3 representa essa situação.



**Algoritmo 2** Classificador GMM usando regra de decisão ML**Input:** Vector de features  $X(k)$  de um dado objecto.

- 1:  $N_C$ : número de classes
- 2:  $\Theta_i$ : modelo GMM de cada classe
- 3:  $k$ : instante de tempo corrente
- 4: **for**  $i : 1$  to  $N_C$  **do**
- 5:    $\mathcal{L}_i(k) \leftarrow P(X(k)|q_i, \Theta_i)$ ; /\* Calcula a probabilidade condicional para cada classe \*/
- 6: **end for**
- 7:  $ML(k) \leftarrow \max[\mathcal{L}_i(k)]$ ; /\* Escolhe o valor máximo da probabilidade condicional \*/

**Output:** Classe com o valor máximo da probabilidade condicional.**Vector característico para o classificador GMM**

Para que o classificador trabalhe adequadamente, é necessário definir um conjunto de características que representem, de maneira inequívoca tanto quanto possível, os diferentes tipos (categorias) de objectos que se deseja distinguir. Para o caso do classificador GMM usando dados laser, o vector característico (*feature vector*) será calculado com base no conjunto de pontos que definem um segmento  $S_j$ , que por sua vez representa um objecto que esteja dentro do campo de alcance do sensor. Ou seja, esta informação sensorial está limitada à face do objecto que reflecte os feixes laser num plano. Com esse conjunto bi-dimensional de pontos, poucas são as características possíveis de utilizar para identificar inequivocamente cada classe de objecto. Após análises com dados experimentais obtidos com um LRF montado num veículo a percorrer ambientes exteriores, cujos objectos detectados foram carros, peões, árvores e postes, definiu-se o vector característico com as seguintes características (veja Figura 6.2):

- Dimensão Cartesiana normalizada do objecto:  $\sqrt{\Delta X^2 + \Delta Y^2}$ ;
- Número de pontos dividido pela distância mínima ao sensor laser:  $\frac{\sum_{i=1}^n (i)}{r_{min}}$
- Desvio padrão entre o centro de massa CM e as coordenadas dos pontos  $\{P_1, \dots, P_n\}$  que definem o objecto;

**Algoritmo 3** Classificador GMM usando regra de decisão MAP**Input:** Vector de features  $X(k)$  de um dado objecto.

- 1:  $k$ : instante de tempo corrente
- 2:  $N_C$ : número de classes
- 3:  $\Theta_i$ : modelo GMM de cada classe
- 4:  $P(q_i, \Theta_i)_{k-1}$ : probabilidade a priori inicial ou do instante anterior
- 5: **for**  $i : 1$  to  $N_C$  **do**
- 6:  $P(q_i, \Theta_i)_k \leftarrow P(q_i|X(k-1), \Theta_i)$ ; /\* A priori igual a posteriori anterior \*/
- 7:  $\mathcal{L}_i(k) \leftarrow P(X(k)|q_i, \Theta_i)$ ; /\* Probabilidade condicional \*/
- 8:  $P(q_i|X(k), \Theta_i) \leftarrow \frac{P(X(k)|q_i, \Theta_i)P(q_i, \Theta_i)_k}{P(X(k))}$ ; /\* Calcula a posteriori actual \*/
- 9: **end for**
- 10:  $MAP(k) \leftarrow \max[P(q_i|X(k), \Theta_i)]$ ; /\* Escolhe o valor máximo da a posteriori \*/

**Output:** Classe com o valor máximo da probabilidade a posteriori.

- Velocidade estimada do objecto (oriunda de um filtro, KF por exemplo).

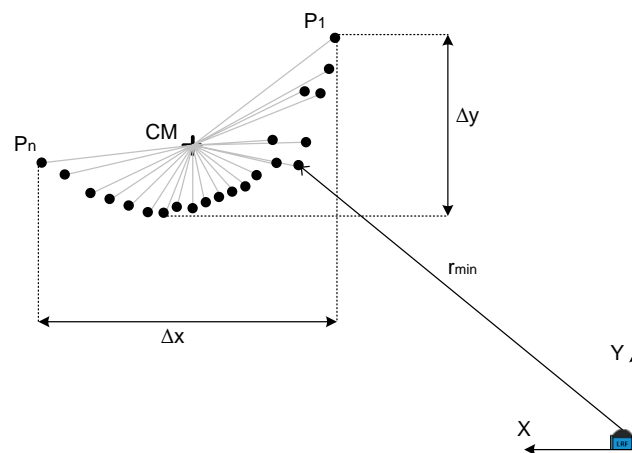


Figura 6.2: Representação no plano de um conjunto de pontos que definem um objecto e as variáveis utilizadas para calcular as componentes do vector característico.

## 6.7 Combinação probabilística de classificadores

Uma abordagem que “combine” alguns, ou vários, classificadores de maneira a se tirar partido das potencialidades de cada classificador tem demonstrado ser uma estratégia adequada para se obter melhor desempenho na tarefa de classificação (Kittler et al., 1998). No âmbito do trabalho aqui apresentado, há interesse em se combinar três tipos de classificadores distintos, dois deles a trabalhar com dados laser, que são o classificador GMM apresentado na secção 6.6 e um classificador múltiplos votos (Mendes et al., 2004), e o terceiro, um classificador AdaBoost, que utiliza os dados de uma câmara (Monteiro, Peixoto, & Nunes, 2006). Motivado pelo trabalho de (Kittler et al., 1998) e pela experiência adquirida com o classificador GMM, optou-se por utilizar a abordagem Bayesiana para combinar o resultado dos três classificadores, cuja regra de decisão final é baseada na “soma” pesada das probabilidades *a posteriori* que cada classificador envia ao módulo de combinação.

Considerando  $N = 3$  o número de classificadores, o objectivo é classificar os objectos detectados numa das  $N_C$  classes  $(q_1, \dots, q_{N_C})$ , tendo como pressuposto que cada classificador envia (como saída) ao módulo de combinação o valor estimado da probabilidade *a posteriori*  $P_{n=1:3}(q_j|X(k))$  de classificação dos objectos detectados. Os resultados apresentados em (Kittler et al., 1998) mostram que a regra de soma é aquela que apresenta melhor desempenho; sendo assim, neste trabalho a regra final de combinação dos classificadores tem em consideração o valor máximo da soma das probabilidades *a posteriori* de cada classificador para cada classe:

$$X(k) \in q_i \longrightarrow (1 - N)P(q_i) + \sum_{n=1}^N P_n(q_i|X(k)) = \max_{j=1}^{N_C} [(1 - N)P(q_j) + \sum_{n=1}^N P_n(q_j|X(k))] \quad (6.19)$$

Ou seja, um objecto (cujo vector característicos é  $X(k)$ ) pertence à classe  $q_i$  se o valor da soma (6.19) das probabilidades for máximo. Convém salientar que, em alguns momentos, a saída do classificador AdaBoost (Monteiro, Peixoto, & Nunes, 2006) não estará disponível

devido ao facto que o campo de visão da câmara é menor que o do LRF. Sendo assim, é natural que os dois classificadores baseados nos dados do sensor laser tenham uma contribuição maior nos resultados.



## Capítulo 7

# Sistema MTDTS e resultados

### Conteúdo

---

7.1	Arquitectura . . . . .	91
7.2	Segmentação e extracção de características . . . . .	93
7.3	Seguimento e associação de dados . . . . .	99
7.4	Análise de resultados do classificador GMM . . . . .	103
7.5	Combinação entre classificadores . . . . .	108

---

**E**STE capítulo apresenta a arquitectura final do sistema MTDTS utilizado em testes experimentais realizados com os veículos *Robucar* e *CyberCab* em ambiente exterior semi-estruturado.

### 7.1 Arquitectura

Os módulos (sub-sistemas) principais do sistema MTDTS proposto estão representados na Figura 7.1. Trata-se de um sistema dinâmico cujas tarefas são efectuadas a cada período de tempo  $k$ , seguindo uma lógica sequencial. Cada sub-sistema efectua um conjunto de

tarefas cuja descrição, para o caso dos dados laser <sup>1</sup>, é a seguinte:

1. **Aquisição e segmentação:** os dados do LRF são recebidos a uma taxa de aproximadamente 36 Hz, através de uma ligação CAN, e uma técnica de segmentação é executada de modo a detectar os dados correspondentes a objectos e aqueles correspondentes ao “fundo” da imagem (que são eliminados).
2. **Seguimento e associação de dados:** para cada segmento detectado  $S_j(k)$ , o centro de massa (CM) é utilizado como ponto-característico nas fases de seguimento: formação, manutenção e finalização do filtro (*tracker*<sup>2</sup>). A associação dos dados é realizada em três etapas: associação do tipo observação-segmento, associação segmento-*tracker* e associação segmento-objecto.
3. **Classificação:** para cada segmento  $S_j(k)$  extrai-se um vector de características  $X(k)$ , utilizado pelos módulos de seguimento e classificação. O classificador GMM calcula a probabilidade condicional, para cada  $X(k)$ , e utiliza uma regra de decisão (ML ou MAP) para classificar o segmento numa das classes de interesse.
4. **Saídas:** o módulo MTDTS envia como saídas, para cada objecto detectado, a informação sobre o estado (posição, velocidade e aceleração) do objecto consoante a classe, os parâmetros da primitiva geométrica que o caracteriza (circulo, rectas ou elipse), e a classe a que pertence. Essa informação é útil para sistemas de anti-colisão, navegação, construção de mapas geométricos, decisão e controlo, entre outros.

Os módulos de actualização de coordenadas (em geral usando informação odométrica) é necessário para evitar que os objectos detectados tenham velocidades “aparentes” (como discutido na secção 2.6). O bloco chamado “repositório” tem como função manter e gerir o histórico dos objectos detectados e as variáveis de estado dos objectos sob seguimento.

---

<sup>1</sup>Os módulos de processamento e de tratamento dos dados de visão (inclusivé o classificador *Adaboost*) foram realizados por outros membros do ISR-UC (Monteiro, Peixoto, & Nunes, 2006).

<sup>2</sup>*Tracker* significa, neste contexto, um filtro associado a um segmento (ou grupo de segmentos) que ainda não foi classificado pelo módulo de classificação.

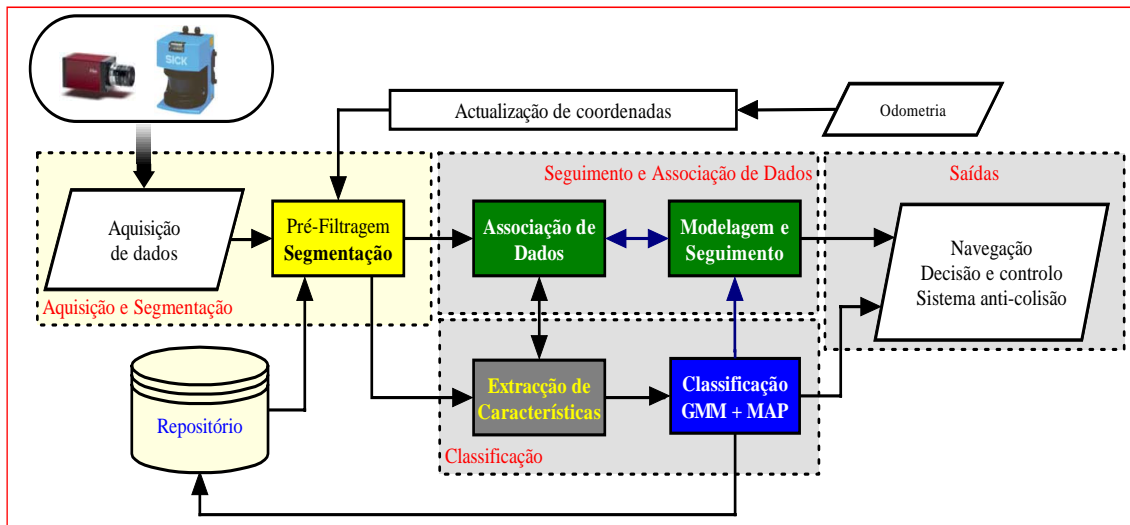


Figura 7.1: Diagrama de blocos representando o sistema MTDTS

## 7.2 Segmentação e extracção de características

Os métodos de segmentação de dados laser 2D, discutidos neste trabalho, são basicamente de dois tipos: métodos baseados na distância Euclidiana entre os pontos (SBDE), e os métodos baseados no filtro de Kalman (SBKF), cujos modelos geométricos caracterizam o comportamento estocástico dos pontos-laser. A primeira categoria tem como vantagens a facilidade na implementação e rapidez de processamento. No entanto, tem como pontos fracos a “rigidez” no mecanismo de detecção do ponto de quebra, pois utiliza limiares fixos, e a baixa flexibilidade porque, ao contrário de alguns métodos do tipo SBKF, não extrai primitivas geométricas. A Tabela 7.2 mostra alguns resultados experimentais obtidos com os métodos da categoria SBDE; essas experiências foram realizadas com dados oriundos de ambientes interiores e exteriores. A segunda coluna apresenta o valor médio <sup>3</sup> das variáveis utilizadas em cada método, sendo que estes valores foram calculados tendo em conta o número de segmentações bem sucedidas.

Os métodos baseados no filtro de Kalman, cujas equações do modelo do processo podem

<sup>3</sup>Calculado com base nos resultados correctos de segmentação



Segmentação SBDE: análise dos métodos		
Variável (equação)	Valor médio	Tempo médio de CPU
$C_0$ (3.4)	246.85	0.0915
$\beta$ (3.5)	0.697	0.0842
$C_e$ (3.6)	0.0478	0.0829
$\lambda$ (3.7)	10.7°	0.0904

Tabela 7.1: Resultado experimental para os métodos SBDE

ser do tipo linear ou não-linear, tem como vantagens, em relação aos métodos SBDE, a não dependência rígida com parâmetros que são ajustados em modo *off-line*. Ou seja, esta abordagem “adapta-se” ao comportamento dos pontos enviados pelo LRF dependendo de como foi definido o modelo. No entanto, como é comum, há sempre a necessidade de “ajustar” os valores da matriz de covariância do processo,  $Q$ , de modo a maximizar o número de sucessos na extracção de segmentos. Outra vantagem da abordagem baseada em KF consiste na possibilidade de implementar um modelo do processo ajustado ao tipo de primitiva geométrica, tais como rectas e círculos. No entanto, nem sempre é fácil a definição das equações que caracterizam o comportamento do processo, “geométrico” neste caso, e na manutenção da coerência do filtro. Nos métodos SBKF, o tempo de processamento é superior ao dos métodos SBDE, sendo de aproximadamente 0.1 a 0.3 segundos consoante o método <sup>4</sup>.

Os métodos de extracção de primitivas geométricas abordados nesta dissertação representam apenas uma pequena parte dos métodos e técnicas descritas em publicações científicas. No entanto, o interesse aqui foi avaliar qualitativamente alguns métodos no contexto da percepção e representação geométrica em tarefas de robótica móvel, com especial interesse para as aplicações com dados laser e na extracção de rectas, círculos e elipses. Os métodos implementados são, deste ponto em diante, designados consoante as seguintes abreviações:

---

<sup>4</sup>Utilizando uma máquina P4, 1.5 GHz, 832 MB-RAM, Matlab<sup>®</sup> 7.0

- GAN = métodos Gander (Gander et al., 1994);
- VAN = método Vandorpe (Vandorpe et al., 1996);
- GUI = método Guivant (Guivant et al., 2000);
- SIA = método Siadat (Siadat et al., 1997);
- LS = métodos tipo *least squares*;
- IEPF = método utilizado para extrair múltiplas rectas num conjunto de pontos;
- FITZ = método Fitzgibbon (Fitzgibbon et al., 1999).

### Extracção de rectas

Em ambiente interior, as rectas são, sem dúvidas, a principal primitiva geométrica utilizada (devido à constante ocorrência de paredes). Em ambiente exterior, e mais especificamente no contexto deste trabalho, restringiu-se a utilização de rectas para representar carros (geralmente dois segmentos de rectas) e eventuais paredes/muros.

Foram implementado os métodos LS, VAN, IEPF e SIA, de cujos resultados se sumariam as seguintes conclusões:

1. No caso de se utilizarem os dados em coordenadas Cartesianas, é necessário impor uma restrição no processo de extracção quando as rectas são colineares com o eixo vertical (referencial do plano Cartesiano). Essa condição é discutida em (Vandorpe et al., 1996), e uma técnica muito simples é utilizada para definir o eixo (sentido) em que a regressão deve ser implementada.
2. É praticamente imprescindível utilizar algum algoritmo ou técnica para detectar a ocorrência de mais de uma recta por segmento, como por exemplo no caso dos veículos ou “cantos” em paredes onde é comum extrair-se duas rectas. Neste trabalho optou-se por usar o algoritmo IEPF, sendo também uma opção interessante o método proposto em (Sarfraz, Rasheed, & Muzaffar, 2005).

3. O método VAN e SIA são praticamente idênticos nos resultados, sendo portanto indiferente a escolha. Os métodos “clássicos” do tipo LS apresentam resultados tão satisfatórios quanto os dois primeiros métodos, mas os dois primeiros são mais simples de implementar.

### Extracção de círculos

Com base nas experiências realizadas com os métodos apresentados na secção 4.3, sobretudo em ambiente exterior, obtiveram-se as seguintes conclusões:

1. Os métodos do tipo LS, e.g. GAN, são impraticáveis em situações nas quais se tem menos de três pontos-laser de entrada; um exemplo disso está ilustrado na Figura 7.2(d).
2. O método de extracção de círculos VAN não apresenta os resultados satisfatórios aos requisitos deste trabalho; sendo justificável apenas quando o número de pontos que definem um segmento for muito pequeno.
3. O método GUI1 apresenta, em termos de facilidade de implementação e tempo de processamento, o melhor desempenho na situação de interesse (ambiente exterior com percepção sensorial baseada em dados laser).

### Extracção de elipses

Apesar de ser útil em muitas situações, a extracção de elipses esteve, nesta dissertação, restrita à aproximação geométrica do veículo *CyberCab* da *Yamaha*. Alguns resultados usando o métodos de FITZ e GAN estão ilustrados na Figura 7.3.

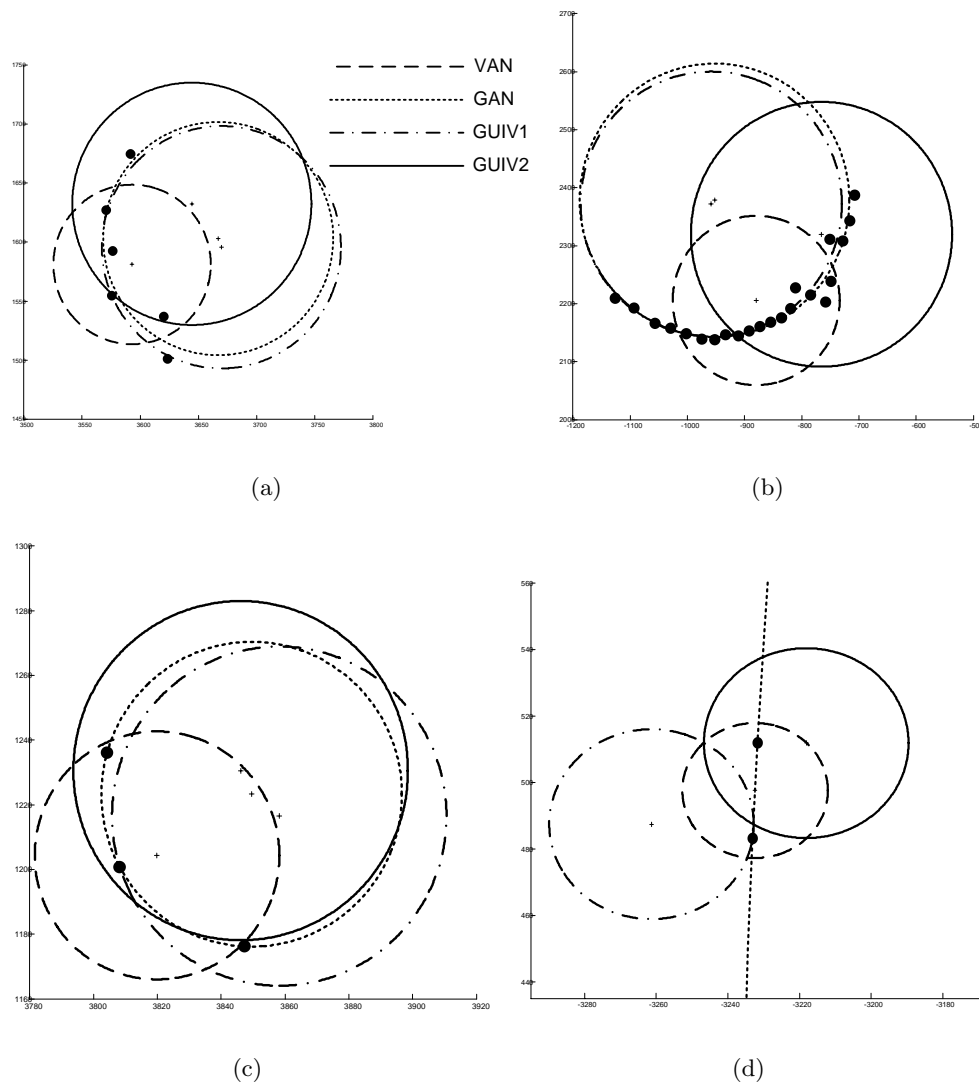


Figura 7.2: Exemplo prático da utilização de círculos na representação geométrica de pontos laser. (a) Extração de círculos nos pontos que definem um segmento; (b) Segmento com muitos pontos; (c) Segmento com três pontos; (d) Segmento de apenas dois pontos.

## Conclusão

Como o objectivo do sistema MTDTS proposto é trabalhar em ambiente exterior e, como a extração de primitivas geométricas não constitui uma necessidade peremptória para

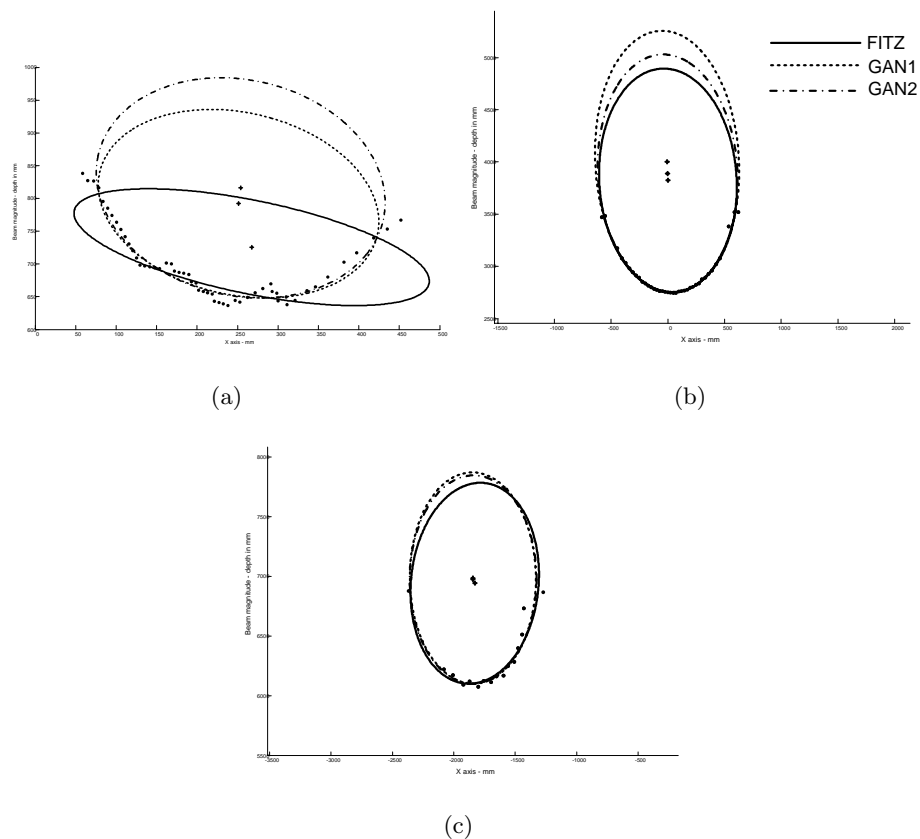


Figura 7.3: Exemplo prático da utilização de elipses para representar objectos segmentados em dados laser. (a) Elipses extraídas num conjunto de pontos laser; (b) Elipses com eixo-principal paralelo ao eixo vertical (ordenadas) do plano laser; (c) Elipses extraídas nos pontos que representam o veículo da *Yamaha*.

o sistema, optou-se assim por utilizar, na segmentação, um método do tipo SBDE (o método proposto por (Borges & Aldon, 2004)). As primitivas geométricas utilizadas para caracterizar os objectos limitaram-se a círculos e segmentos de recta, cujos métodos escolhidos foram GUI1 (Guivant et al., 2000), descrito na Secção 4.3, e VAN (Vandorpe et al., 1996), respectivamente. A extracção de elipses foi utilizada apenas para fins de visualização gráfica, sendo os métodos de GAN (Gander et al., 1994) mais precisos, porém menos fáceis de implementar.

## 7.3 Seguimento e associação de dados

Para o sistema MTDTS implementado e apresentado nesta dissertação optou-se por utilizar o filtro de Kalman como solução mais adequada aos cenários experimentais em estudo. A principal motivação dessa escolha prende-se ao facto que o seguimento dos objectos é realizado no espaço Cartesiano com modelos lineares, e porque adoptou-se como pressuposto que o ruído adicionado ao processo e ao modelo das medidas é Gaussiano, branco e de média nula. Outra condição que motivou essa escolha é que os cenários eram pouco *cluttered*.

### Associação de dados

Pode-se seguramente dizer que um dos pontos mais críticos do sistema MTDTS aqui apresentado está relacionado ao módulo de associação de dados (*data association*). É fácil de perceber que, caso haja algum erro na associação da informação entre os segmentos (ou objectos) detectados, os módulos de seguimento e classificação são negativamente comprometidos e em consequência o sistema torna-se menos fiável e robusto. Como discutido na secção 5.2, as três falhas possíveis podem ocorrer numas das seguintes etapas de associação:

1. segmento-segmento: associar segmentos detectados no instante anterior com os actuais;
2. segmento-tracker: associar um, ou mais, segmentos a um único tracker (objecto não classificado) e assim formar apenas um vector característico  $X(k)$  para este *tracker*;
3. segmento-objecto: associar um, ou mais, segmentos aos objectos existentes e já classificados.

A Figura 7.4 ilustra, esquematicamente, o fluxo da informação no sistema MTDTS aqui discutido, que está subdividido em três partes:

**1. Aquisição, segmentação e extracção de características:**

1.1 Os dados do laser estão constantemente a ser enviados ao módulo de filtragem e segmentação a taxa aproximada de 36 Hz;

1.2 Um conjunto  $S_j$  de segmentos é extraído a cada intervalo de tempo, e esse grupo de pontos é enviado ao módulo de extracção de características;

1.3 Para cada segmento extrai-se um vector de características geométricas  $X(k)$  (para o módulo de classificação); o centro de massa (CM) é o ponto-característico utilizado como valor observado  $Z_j$  para o módulo de seguimento e associação de dados;

**2. Seguimento e associação de dados:**

2.1 São utilizados filtros de Kalman (KF) independentes para cada “entidade” (objecto, *tracker*, *tracker* latente) existente no sistema MTDTS;

2.2 São efectuados dois ciclos de predição: um para os objectos sob seguimento e outro para os *trackers* existentes;

2.3 Para cada objecto já classificado, uma região de validação (RV) é construída consoante a classe do objecto;

2.4 Para cada *tracker*, é construída uma RV elipsoidal. Para os *trackers* em estado latente é utilizada uma RV rectangular;

2.5 Os CM calculados para cada segmento  $S_j$  detectado são agrupados num vector de observações  $Z_j$ ;

2.6 O processo de validação segue uma ordem hierárquica:

2.6.1 Os valores de  $Z_j$  que passam, primeiramente, no teste de validação de cada objecto sob seguimento é utilizado no ciclo de correcção do KF associado ao objecto validado;

2.6.2 Os valores “não-associados” a objectos são utilizados nos testes de validação dos *trackers* existentes;

2.6.3 As observações que, nesta terceira fase, não foram associadas aos *trackers* são então utilizadas para actualizar *trackers* latentes ou para criar um novo *tracker* (módulo de formação);

2.7 OS KF associados aos objectos e *trackers* que não foram validados, nos ciclos de associação de dados, são analisados nos módulos de manutenção e finalização: podem ser mantidos sob seguimento sem o ciclo de correcção (apenas predição) ou podem ser eliminados caso o número de não-associações ultrapasse o limite estipulado;

### 3. Classificação:

3.1 O vector de características  $X(k)$  associado a cada *tracker* existente é actualizado com o valor da velocidade estimada pelo respectivo KF;

3.2 Cada vector  $X(k)$  é enviado ao classificador GMM;

3.3 Regras de decisão do tipo ML/MAP são utilizadas, com base nos valores probabilísticos calculados pelo classificador GMM, para classificar, temporariamente, os *trackers* numa das classes de interesse;

3.4 Uma classificação mais fiável e estável é obtida após, pelo menos, cinco ciclos de seguimento dum *tracker*;

3.5 O módulo gestor da informação mantém a lógica sequencial e a coerência do algoritmo.

Sucintamente, o objectivo principal do sistema MTDTS é detectar e classificar objectos que, ao longo do tempo, assumem diferentes “identidades”: **tracker latente** (segmento com um filtro “sem histórico”), **tracker** (segmento com um filtro associado mas, ainda não classificado), e finalmente o **objecto** (segmento já classificado como pertencente a uma das classes de interesse: pessoas, postes/árvores, veículos). O Algoritmo 4 ilustra as principais fases envolvidas no processo de associação de dados e “gestão” do seguimento de objectos. No sistema MTDTS apresentado, cada “identidade” tem um processo próprio de associação, ou seja:



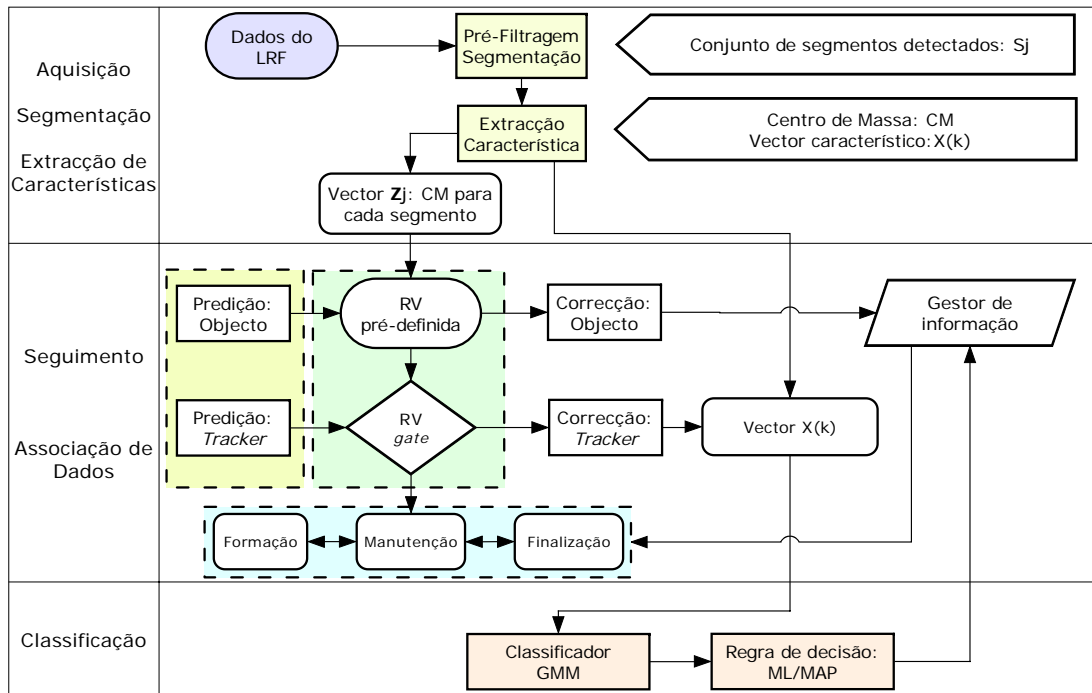


Figura 7.4: Diagrama sequencial da informação nos subsistemas que compõem o sistema MTDTS.

- **tracker latente**: usa uma RV do tipo rectangular (5.20) pois é um segmento cujo KF não tem histórico suficiente para calcular uma RV do tipo elipsoidal (que depende de valores “consistentes” das matrizes de covariância do filtro estocástico associado);
- **tracker**: usa uma RV elipsoidal (5.19) pois tem histórico suficiente para garantir a estabilidade do filtro e, conseqüentemente, assegurar a construção coerente da RV;
- **objecto**: usa RV específica para cada classe: uma RV circular para árvores/postes; uma RV rectangular com dimensões fixas em função do tamanho do veículo ou carro classificado; uma elipse com dimensões também fixas (eixo maior: 120 cm; eixo menor: 50 cm) para os peões classificados.

A gestão do processo de seguimento e associação de dados passa por três fases ou módulos (secção 5.2): criação, manutenção (*update* do filtro) e finalização. Sendo que

dependem basicamente do tempo de vida (número de associações), e do tempo de não-associação. Conclui-se que, se o módulo de associação de dados por ventura falhar, todo o processo de gestão, predição e actualização dos filtros é negativamente comprometido.

## 7.4 Análise de resultados do classificador GMM

As experiências com o classificador GMM foram realizadas em ambiente exterior com um sensor LRF montado nos veículos *Robucar* e *CyberCab*, a uma altura de aproximadamente 64.5 cm e 66.5 cm respectivamente. A Tabela 7.4 apresenta o número de dados utilizados na fase de treino (supervisionado) e na fase de construção das curvas ROC (Receiver Operating Characteristic), para a análise do classificador. Além disso, apresenta o número das amostras verdadeiras (positivos) e “não-verdadeiras” para cada classe (negativos). A análise de desempenho de um classificador passa, na maioria dos casos, pela construção de uma matriz de confusão (Tabela 7.4), cujas variáveis serão utilizadas na curva ROC. As matrizes de confusão utilizadas na análise dos resultados dos classificadores GMM-ML e GMM-MAP são fornecidas no Apêndice A. Representando a resposta  $f(i)$  do classificador a uma dada ocorrência/instância  $\mathcal{I}$ , as principais variáveis (métricas) utilizadas na análise de desempenho são:

- Verdadeiro positivo (VP): se  $\mathcal{I}$  é positiva (verdadeira) e a resposta do classificador  $f(i)$  é também positiva;
- Falso positivo (FP): se  $\mathcal{I}$  é negativa e  $f(i)$  é positiva;
- Verdadeiro negativo (VN): se  $\mathcal{I}$  é negativa e  $f(i)$  é negativa;
- Falso negativo (FN): se  $\mathcal{I}$  é positiva e  $f(i)$  é negativa;
- Taxa de verdadeiro positivo (TVP):  $TVP = \frac{VP}{P}$ , onde P é o número total de amostras positivas (verdadeiras);
- Taxa de falso positivo (TFP):  $TFP = \frac{FP}{N}$ , onde N é o número total de amostras negativas;

Com base nestas variáveis, a curva ROC é construída tendo por base a métrica TVP (eixo das ordenadas) e a TFP (abscissas). No entanto, nesta dissertação o número de FP será utilizado no eixo das abscissas, em vez da TFP. Este procedimento justifica-se porque o número de amostras na fase de análise não é suficiente para que o desempenho do classificador atinja a escala máxima (valor unitário) na métrica TFP e também para estar mais claro a variação na resposta do classificador com o valor absoluto de FP.

		Positivos	Negativos
$f(i)$	Classe	VP	FP
	Não-classe	FN	VN

Tabela 7.2: Matriz de confusão, onde  $f(i)$  é a resposta do classificador

O processo de construção das ROC envolveu uma variação de um limiar (com valor entre 0.0 e 1.0, com uma variação incremental de 0.002), correspondente à resposta probabilística do classificador com base num conjunto de amostras oriundas de ambientes exteriores semi-estruturados. O Algoritmo 5 foi utilizado para gerar as ROC com base nas amostras correspondentes a cada uma das classes de interesse. Como temos dois tipos de regras de decisão, ML e MAP discutidas na secção 6.6, o algoritmo foi utilizado para o classificador GMM-ML e GMM-MAP.

Classificador GMM: número de amostras utilizadas				
Classe	Treino	ROC	Positivos	Negativos
Class1: pessoas	1053	288	96	192
Class2: carros	1357	288	96	192
Class1: postes/árvores	660	288	96	192

Tabela 7.3: Dados utilizados no classificador GMM para as diferentes classes e para a construção das curvas ROC

Uma curva ROC é constituída por um conjunto de pontos que, de um modo sequencial, são posicionados no espaço ROC e representam um “balanço” entre os benefícios (verda-

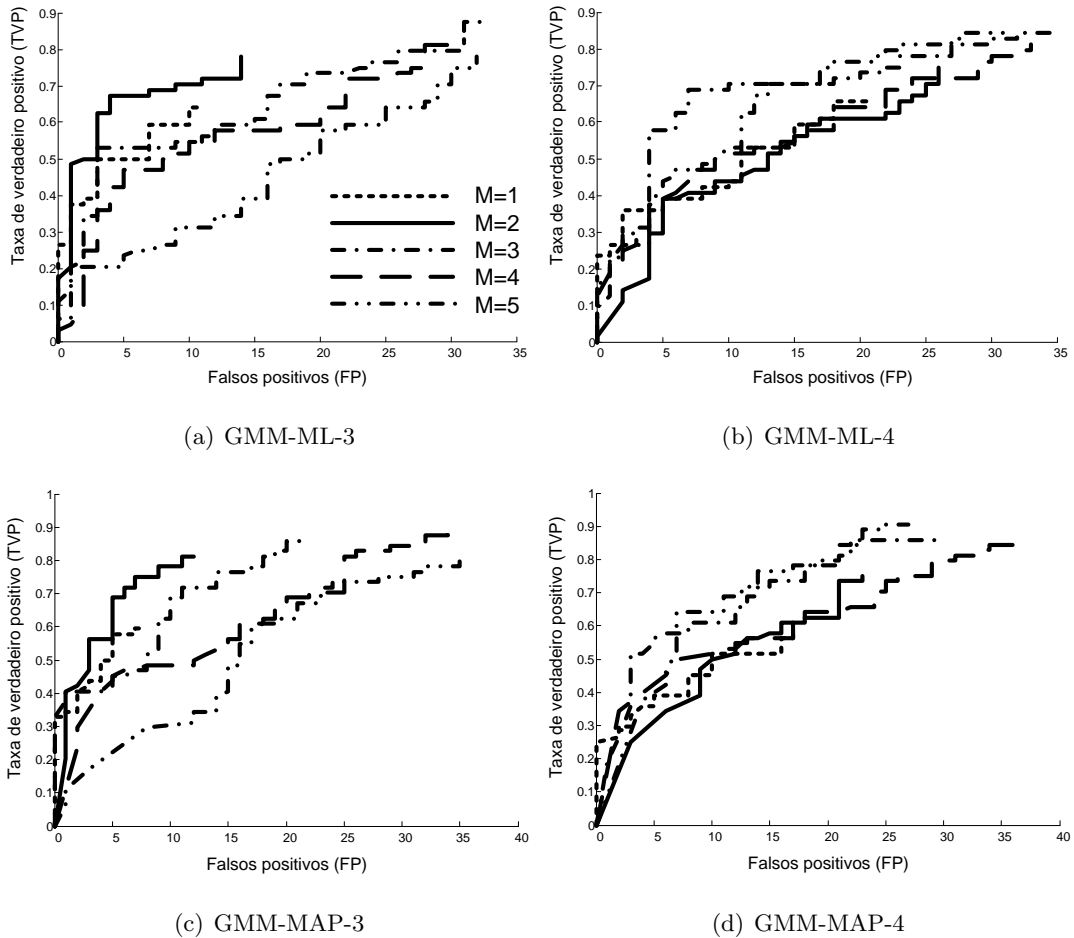


Figura 7.5: Curvas ROC do classificador GMM para a classe1: peões. O número de componentes Gaussianas utilizadas em cada modelo é representado por  $M$ .

deiros positivos) e os custos (falsos positivos). É conveniente interpretar alguns pontos de interesse no espaço ROC:

- Ponto (0,0): representa a resposta de um classificador sem falsos positivos mas, ao mesmo tempo, sem nenhum resultado positivo na classificação;
- Ponto (1,1): é o oposto do ponto anterior, e representa uma classificação a 100% mas, com uma taxa máxima de falsos positivos;

- Ponto (0,1): é a resposta ideal (ou perfeita), na qual o classificador atinge a taxa máxima de acertos mantendo a taxa mínima de falsos positivos.

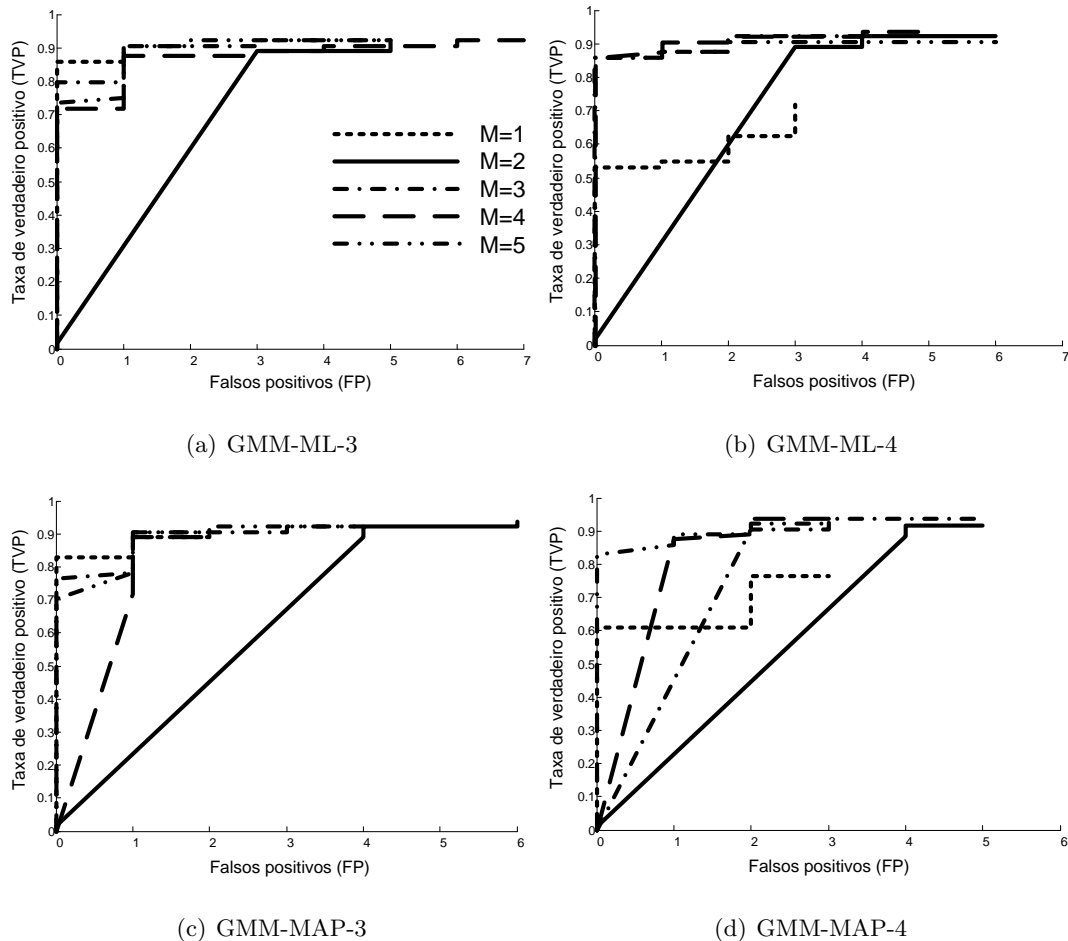


Figura 7.6: Curvas ROC do classificador GMM para a classe2: carros/veículos.

De um modo directo e conciso, o desempenho de um classificador é função do comportamento da curva ROC quando esta atinge um ponto “óptimo”. Ou seja, a resposta do classificador atinge um valor elevado na TVP (acima de 0.9) mantendo, no entanto, uma TFP relativamente baixa. Após este ponto óptimo, a curva ROC atinge um comportamento de “estabilidade”: o incremento na TVP é insignificante e a TFP segue em crescimento. Conclui-se que classificadores com curvas cujo ápice está na região superior-

esquerda do espaço ROC possuem bom desempenho, e quanto mais extremo for este ápice (ponto óptimo), melhor a resposta do classificador.

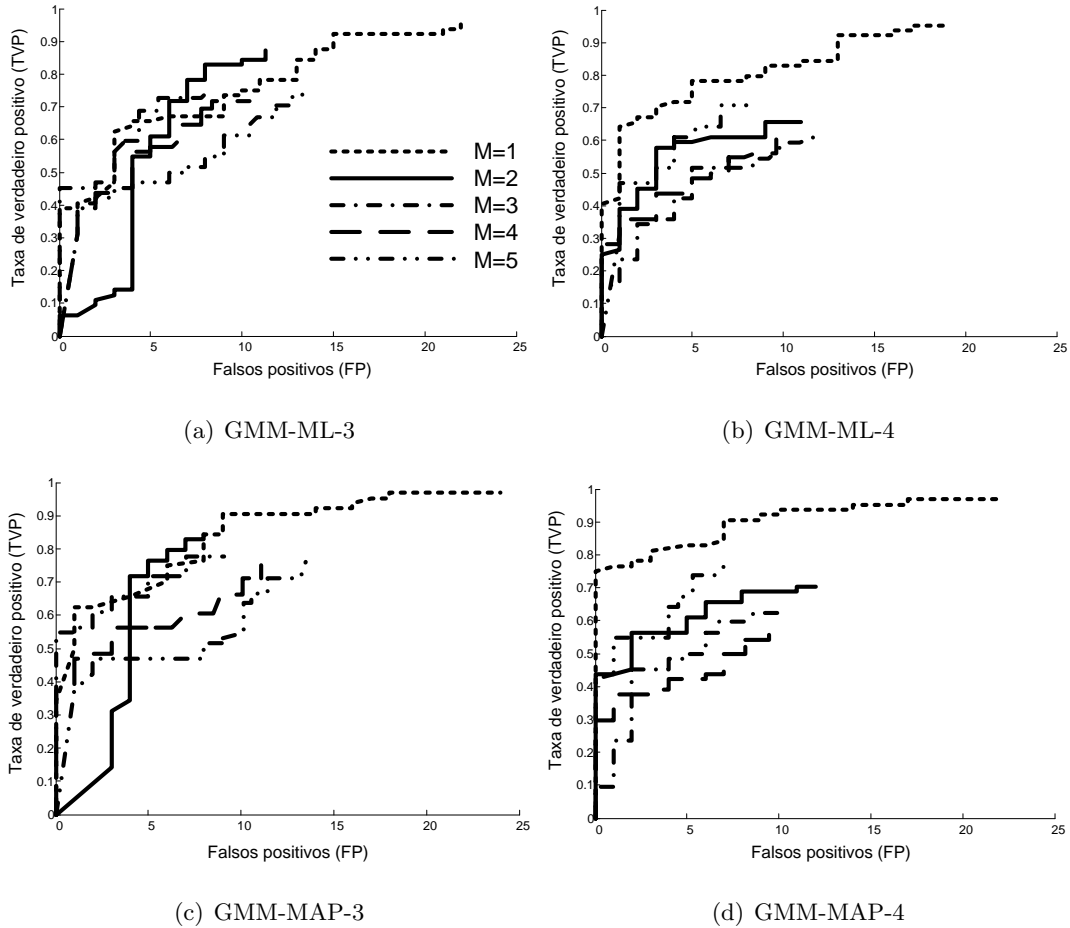


Figura 7.7: Curvas ROC do classificador GMM para a classe3: árvores/postes.

As Figuras 7.5, 7.6, e 7.7 apresentam a curva ROC para vinte “variações” implementadas do classificador GMM, representadas da seguinte forma: GMM-*rd-dvc*, onde *rd* (regra de decisão) e *dvc* (dimensão do vector de características). Por exemplo, o classificador GMM-MAP-3 refere-se ao classificador com regra de decisão tipo MAP, e utilizando as três primeiras componentes do vector de características<sup>5</sup>. Uma rápida análise nas curvas

<sup>5</sup>Além de comparar o classificador GMM com regras de decisão distintas, e com  $M$  modelos GMM, optou-se por variar a dimensão do vector de características

ROC pode indicar que o classificador GMM-MAP e o GMM-ML apresentam, na maioria dos casos, desempenho semelhantes.

Entretanto há uma diferença fundamental entre os dois tipos de regras de decisão: a regra ML não depende do “histórico” passado do objecto sob seguimento, enquanto a regra MAP depende da probabilidade *a priori* calculada no instante anterior e que reflecte o passado probabilístico do objecto. Na prática, o classificador GMM-ML tem resultados instantâneos a cada período de tempo  $k$  independente da probabilidade condicional anterior, sendo por isso um classificador completamente dependente do valor do vector de características observado naquele instante. O classificador GMM-MAP possui um comportamento mais estável pois depende tanto da probabilidade condicional (função do vector característico observado no instante) quanto da probabilidade *a priori* (função do histórico probabilístico dos instantes anteriores).

A Tabela 7.4 ilustra o comportamento do classificador GMM com regras ML e MAP numa situação prática: um peão (classe1) sob seguimento apresenta resultados diferentes na resposta do classificador (valor probabilístico) ao longo do tempo, ocorrendo o mesmo para as outras classes. Nota-se que o classificador GMM-MAP tende a manter o resultado mais estável, ao contrário do GMM-ML que apresenta variações mais abruptas. Esse comportamento está comprovado pelos valores médios e os desvios-padrão apresentados na Tabela 7.4.

Independente de qual regra utilizada, ambos os classificadores dependem da correcta associação entre os segmentos detectados e os filtros existentes associados aos *trackers* sob seguimento. Caso haja um erro na associação o resultado e o histórico do módulo classificador ficam automaticamente prejudicados.

## 7.5 Combinação entre classificadores

O pressuposto para implementar a regra de combinação de classificadores (6.19) é que cada classificador tenha como saída, a cada instante de tempo, um valor probabilístico para

Classificador <b>GMM-MAP</b>								
Classes	Instantes de tempo / Probabilidades						Média	Desvio-padrão
	k	k+1	k+2	k+3	k+4	k+5	$\mu$	$\sigma$
Pedestre	0.9196	0.9544	0.9871	0.9999	0.9667	1.0	0.9713	0.0312
Veículo	0.0759	0.0078	0.0129	0.0001	0.0333	0.0	0.0216	0.0292
Árvore/poste	0.0045	0.0378	0.0	0.0	0.0	0.0	0.0070	0.0152
Classificador <b>GMM-ML</b>								
Pedestre	0.9196	0.5461	0.7999	0.9979	0.6027	0.9994	0.8109	0.1979
Veículo	0.0759	0.0540	0.2001	0.0021	0.3973	0.0006	0.1216	0.1535
Árvore/poste	0.0045	0.3999	0.0	0.0	0.0	0.0	0.0674	0.1629

Tabela 7.4: Comportamento dos classificadores GMM-ML e GMM-MAP num caso de falha na associação entre objectos de classes distintas.

cada classe do objecto sob seguimento. O classificador GMM com regra de decisão MAP já disponibiliza a probabilidade *a posteriori* para cada objecto. Para o caso do classificador MV, adoptou-se a seguinte expressão para estimar a probabilidade de classificação:

$$P(q_i|X(k)) \triangleq \frac{NV_i}{\sum_n V(n)_i} = \frac{NV_i}{NV_T} \quad (7.1)$$

onde  $NV_i$  é o somatório de todos os votos dos actores de votação da classe  $q_i$  até ao momento corrente e  $NV_T$  é o número total de votos possíveis para esta classe. Ou seja, a probabilidade *a posteriori* é uma combinação da resposta das funções de votos normalizada pelo número total de votos.

No caso do classificador AdaBoost, o valor estimado da probabilidade pode ser calculado com base na formulação chamada *Logistic Correction* (Friedman, Hastie, & Tibshirani, 2000). Considere-se um conjunto de classificadores fracos da forma  $sign(x^j - c)$ , onde  $x^j$  é um valor da característica  $j$  para  $j = 1, \dots, p$ , e  $p$  é a dimensão do vector característico. Uma constante  $c$  é calculada de forma que o classificador fraco tenha uma taxa de erro mínima. Sendo  $g$  e  $h(> g)$  os índices das categorias, vamos considerar o caso do problema



de classificação de duas categorias  $C_g$  e  $C_h$ , onde os índices  $g$  e  $h$  têm valores de +1 e -1 respectivamente. A saída de cada classificador fraco, a cada ciclo  $k$  do AdaBoost, é dada por  $f_{gh}^{(k)}(x)$ , onde  $k = 1, \dots, T$ . A saída combinada do classificador é dada por:

$$F_{gh}(x_i) \equiv \sum_{k=1}^T \alpha_{gh}^k f_{gh}^k(x_i) \quad (7.2)$$

onde  $x_i$  é o conjunto de dados e  $\alpha_{gh}^k$  é a função de peso de cada classificador fraco  $f_{gh}^k(x_i)$ . O valor da classe para  $y_i$  é estimado com base no “sinal” da função  $F_{gh}(x_i)$ . Definindo as probabilidades (Nishii, 2003)

$$\begin{aligned} PY_i = g | x_i, Y_i \in g, h = p_i(g, h) \\ PY_i = h | x_i, Y_i \in g, h = p_i(h, g) \end{aligned} \quad (7.3)$$

a probabilidade *a posteriori* de saída do classificador AdaBoost para cada classe ( $g$  ou  $h$ ), é dada por:

$$\begin{aligned} p_i(g, h) &= \frac{1}{1 + \exp -2F_{gh}(x_i)} \\ p_i(h, g) &= 1 - p_i(g, h) \end{aligned} \quad (7.4)$$

Com base na transformação de coordenadas entre os sensores, apresentada na secção 6.7, obtém-se a correspondência directa entre alguns dos objectos detectados pelo LRF e aqueles detectados pela câmara. A Figura 7.8 ilustra, num determinado instante de tempo, alguns objectos detectados no plano imagem com o auxílio da informação enviada pelo módulo de cooperação sensorial entre o laser e a câmara. As regiões de interesse (ROI) na imagem estão representadas pelas linhas verticais, cujas coordenadas foram calculadas com base na informação do laser e na relação de transformação entre os referencias sensoriais.

Para facilitar a interpretação dos resultados, os testes experimentais realizados, na combinação dos classificadores, restringiram-se aos objectos que são detectados, num mesmo *frame*, pelo laser e pela câmara; os resultados ficaram condicionados a duas classes: pessoas (peões) e carros (veículos). A Tabela 7.5 demonstra as taxas de classificação positivas

(TVP) e o número de falsos positivos numa situação experimental de classificação de objectos (peões e veículos) em ambiente exterior.

Classificador	Peões		Veículos	
	TVP	FP	TVP	FP
Classificador <i>AdaBoost</i>	89.8	47	93.4	31
Classificador GMM	90.3	17	89.9	28
Múltiplos Votos (MV)	82.5	23	84.0	30
Combinação dos classificadores	92.4	9	93.8	11

Tabela 7.5: Taxa de classificação (TVP) e número de falsos positivos (FP)



Figura 7.8: Detecção e classificação de objectos usando o classificador *AdaBoost*. As linhas verticais definem as ROI no plano imagem em função da informação do módulo de cooperação sensorial com o laser.

**Algoritmo 4** Associação de dados para o sistema MTDTS**Input:** Vector de observações  $\mathbf{Z}(k) = CM_k$  para o conjunto de segmentos  $S(k)$  detectados.**Output:** Actualização do histórico dos filtros estocásticos, trackers e objectos.

```

1:  $N_{seg}(k-1)$ : número de segmentos detectados no instante anterior
2:  $N_{trk}(k-1)$ : número de trackers existentes
3:  $N_{obj}(k-1)$ : número de objectos existentes
4:  $TV$ : tempo de vida;  $FSA$ : filtro sem associação
5:  $RV$ : região de validação: rectangular, objecto ou elipsoidal
6: while  $k$  do
7:   Constrói  $N_{seg}(k)$   $RV_r$  : para associar  $S(k-1) \rightarrow Z(k)$ 
8:   Constrói  $N_{obj}(k)$   $RV_o$  : para cada objecto existente
9:   Constrói  $N_{trk}(k)$   $RV_e$  : para cada tracker existente
10:  for  $j = 1$  to  $N_{seg}(k-1)$  do
11:    if  $\mathbf{z}_j(k) \in RV_r(k|k-1)$  and  $TV \geq 2$  then
12:      Criação: inicia um novo tracker
13:    else
14:      Finalização: destruição de um tracker latente
15:    end if
16:  end for
17:  for  $j = 1$  to  $N_{obj}(k-1)$  do
18:    if  $\mathbf{z}_j(k) \in RV_o(k|k-1)$  then
19:      Manutenção: actualização do filtro associado ao objecto (TV++)
20:    else
21:      FSA ++
22:    end if
23:  end for
24:  for  $j = 1$  to  $N_{trk}(k-1)$  do
25:    if  $\mathbf{z}_j(k) \in RV_e(k|k-1)$  then
26:      Manutenção: actualização do filtro associado ao tracker (TV++)
27:      Classificação: Tracker  $\implies$  Objecto
28:    else
29:      FSA ++
30:    end if
31:  end for
32: end while

```

---

**Algoritmo 5** Pseudo-código para a construção da ROC

---

**Input:**  $\mathcal{X}$ : conjunto de amostras (instâncias);  $f(i)$ : a probabilidade estimada pelo classificador que a instância  $i$  é positiva;  $thr$ : limiar ( $0 \leq thr \leq 1$ );  $inc$ : incremento, a menor diferença entre dois valores de  $f$  quaisquer.

**Output:** Pontos bidimensionais para a ROC.

```
1: FP: falso positivo; VP: verdadeiro positivo, P: n° de positivos.
2: thr = 1.0;
3: inc = 0.002;
4: while thr do
5:    $FP \leftarrow 0$ 
6:    $VP \leftarrow 0$ 
7:   for  $i = 1 \in X$  do
8:     if  $f(i) \geq thr$  then
9:       if  $i$  é uma instância positiva then
10:         $VP \leftarrow VP + 1$ 
11:       else
12:         $FP \leftarrow FP + 1$ 
13:       end if
14:     end if
15:   end for
16:   Guarda o ponto  $(FP, VP/P)$  para a curva ROC;
17:   thr = thr - inc;
18: end while
```

---



## Capítulo 8

# Conclusões e trabalho futuro

### Conteúdo

---

8.1	Conclusões . . . . .	115
8.2	Trabalho Futuro . . . . .	117

---

**E**STE capítulo apresenta as principais conclusões referentes ao sistema implementado e discutido no capítulo 7, e aponta novos desenvolvimentos para trabalhos futuros.

### 8.1 Conclusões

O trabalho aqui apresentado centrou-se na área da percepção e classificação de objectos, presentes em ambientes exteriores semi-estruturados, que são detectados por sensores embarcados em robots ou veículos autónomos. O contexto a que se refere o termo semi-estruturado diz respeito a cenários com pavimento regular e aproximadamente plano, nos quais os veículos e objectos se movem a baixas velocidades (inferiores a 20 Km/h), e cujos objectos de interesse são peões, carros e eventuais árvores ou postes de sinalização ou iluminação. Os componentes principais do projecto, designado por sistema MTDTS, são os subsistemas de segmentação, seguimento e associação de dados, e classificação dos objectos. Apesar de haver um módulo de visão embarcado nos veículos, os resultados ex-

perimentais alcançados e as conclusões obtidas estão centradas no tratamento dos dados laser adquiridos por um LRF LMS200.

O sistema MTDTS é composto por subsistemas (módulos) interdependentes, sendo que a falha ou mau funcionamento de um dos módulos compromete directa ou indirectamente a fiabilidade de todo o sistema. Com isso em mente, várias técnicas, algoritmos e métodos relacionados a cada um dos subsistemas foram avaliados, implementados e comparados de modo a escolher aqueles que seriam utilizados na arquitectura do sistema MTDTS. Em resumo, as principais conclusões obtidas são:

- O módulo de aquisição e segmentação não representa grandes preocupações ou desafios de projecto;
- O módulo de extracção de características em si é relativamente simples, não fosse a grande importância que este tem associado ao classificador;
- Apesar de alguns trabalhos não mencionarem ou não implementarem um subsistema de classificação de objectos, fica evidente que um classificador associado aos módulos de seguimento e associação de dados dá muito mais flexibilidade (adaptabilidade) ao sistema;
- A escolha de filtros estocásticos no contexto deste trabalho ficou facilitada face aos pressupostos associados aos modelos do processo e das medidas, nomeadamente a utilização de modelos lineares com ruído branco de média nula e, principalmente, à consideração de ambientes pouco densos (*cluttered*). Esta última consideração é talvez a mais importante, apesar de mais restritiva, pois permite utilizar múltiplos filtros independentes para cada objecto detectado;
- O subsistema de associação de dados, neste tipo de arquitectura MTDTS, é o mais crítico e responsável, em grande parte, pela maior robustez e fiabilidade do sistema. Por este motivo foi dado ênfase às técnicas de associação de dados comumente utilizadas, nomeadamente a região de validação elipsoidal. No entanto, um dos grandes desafios em sistemas MTDTS baseados em sensor LRF é que um mesmo

objecto pode, durante o processo de detecção, ser formado por um ou vários grupos de segmentos e, se estivermos a considerar um ambiente com grande densidade de objectos, o problema da associação de dados torna-se mesmo um grande desafio.

- O módulo de combinação de classificadores depende directamente de dois outros subsistemas: transformação de coordenadas do módulo de cooperação sensorial, e do módulo de seguimento e associação de dados. Qualquer falha num destes módulos compromete o desempenho do resultado da combinação de classificadores, sendo por isso necessário repensar e encontrar uma estratégia de cooperação sensorial mais fiável.

## 8.2 Trabalho Futuro

Como sugestões de trabalho futuro enumeram-se as seguintes:

1. Estudo e implementação de filtros estocásticos do tipo *joint*, i.e. técnicas de seguimento de múltiplos objectos nas quais um único algoritmo trata do problema de predição, estimação e associação num sentido de interdependência entre os *trackers*;
2. Adoptar modelos de estado mais completos para descrever o comportamento dos objectos, como exemplo a adição de variáveis de estado proporcionais a outras características que definem um objecto e não somente às cinemáticas: posição, velocidade e aceleração;
3. Tirar partido da informação sensorial, se disponível, do ambiente envolvente, de maneira a maximizar a percepção do cenário e dos objectos ou obstáculos presentes;
4. Explorar, com mais aprofundamento, as técnicas de cooperação e fusão sensorial, tanto com sensores actualmente instalados quanto com outros tipos de sensores que possam ser adicionados à arquitectura;



5. Similarmente ao ponto anterior, as técnicas de cooperação e combinação de classificadores, a trabalhar em espaços de medição diferentes, precisam ser investigadas com mais pormenor.

# Apêndice A

## Matriz de confusão

NESTE apêndice apresenta-se as matrizes (tabelas) de confusão para as situações experimentais em que o classificador GMM foi utilizado, tanto com a regra ML como MAP.

### A.1 Matriz de confusão para o classificador GMM-ML

		Pos	Neg	Pos	Neg	Pos	Neg	Pos	Neg	Pos	Neg
$f(i)$	Classe	170	35	182	58	217	120	192	103	181	108
	Não-classe	85	404	42	312	28	272	30	226	40	224
		1-GMM		2-GMM		3-GMM		4-GMM		5-GMM	

Tabela A.1: Matriz de confusão usando regra ML para a Classe-1: peões

As Tabelas A.1, A.2 e A.3 referem-se aos resultados experimentais do classificador GMM, de regra de decisão ML, com vector característico composto por três componentes características: tamanho, desvio padrão dos pontos e velocidade estimada.

As Tabelas A.4, A.5 e A.6 referem-se aos resultados experimentais do classificador GMM, de regra de decisão ML, com vector característico composto por quatro componen-

		Pos	Neg	Pos	Neg	Pos	Neg	Pos	Neg	Pos	Neg
$f(i)$	Classe	158	58	175	106	198	144	159	125	164	107
	Não-classe	85	319	56	228	33	227	41	215	33	212
		1-GMM		2-GMM		3-GMM		4-GMM		5-GMM	

Tabela A.2: Matriz de confusão usando regra ML para a Classe-2: carros/veículos

		Pos	Neg	Pos	Neg	Pos	Neg	Pos	Neg	Pos	Neg
$f(i)$	Classe	131	4	94	14	117	16	120	21	118	15
	Não-classe	26	443	13	400	13	372	13	298	13	287
		1-GMM		2-GMM		3-GMM		4-GMM		5-GMM	

Tabela A.3: Matriz de confusão usando regra ML para a Classe-3: árvores/postes

		Pos	Neg	Pos	Neg	Pos	Neg	Pos	Neg	Pos	Neg
$f(i)$	Classe	97	14	94	23	99	15	98	13	108	22
	Não-classe	48	380	12	309	13	326	10	276	14	268
		1-GMM		2-GMM		3-GMM		4-GMM		5-GMM	

Tabela A.4: Matriz de confusão usando regra ML para a Classe-1: peões

		Pos	Neg	Pos	Neg	Pos	Neg	Pos	Neg	Pos	Neg
$f(i)$	Classe	273	84	218	34	155	12	106	12	106	25
	Não-classe	12	301	51	276	107	334	93	312	95	299
		1-GMM		2-GMM		3-GMM		4-GMM		5-GMM	

Tabela A.5: Matriz de confusão usando regra ML para a Classe-2: carros/veículos

tes características (descritas na Secção 6.6).

		Pos	Neg	Pos	Neg	Pos	Neg	Pos	Neg	Pos	Neg
$f(i)$	Classe	222	74	134	42	128	21	117	28	104	11
	Não-classe	13	255	103	269	134	297	115	257	93	272
		1-GMM		2-GMM		3-GMM		4-GMM		5-GMM	

Tabela A.6: Matriz de confusão usando regra ML para a Classe-3: árvores/postes

		Pos	Neg	Pos	Neg	Pos	Neg	Pos	Neg	Pos	Neg
$f(i)$	Classe	170	35	182	58	217	120	192	103	181	108
	Não-classe	85	404	42	312	28	272	30	226	40	224
		1-GMM		2-GMM		3-GMM		4-GMM		5-GMM	

Tabela A.7: Matriz de confusão usando regra MAP para a Classe-1: peões

		Pos	Neg	Pos	Neg	Pos	Neg	Pos	Neg	Pos	Neg
$f(i)$	Classe	158	58	175	106	198	144	159	125	164	107
	Não-classe	85	319	56	228	33	227	41	215	33	212
		1-GMM		2-GMM		3-GMM		4-GMM		5-GMM	

Tabela A.8: Matriz de confusão usando regra MAP para a Classe-2: carros/veículos

		Pos	Neg	Pos	Neg	Pos	Neg	Pos	Neg	Pos	Neg
$f(i)$	Classe	131	4	94	14	117	16	120	21	118	15
	Não-classe	26	443	13	400	13	372	13	298	13	287
		1-GMM		2-GMM		3-GMM		4-GMM		5-GMM	

Tabela A.9: Matriz de confusão usando regra MAP para a Classe-3: árvores/postes

## A.2 Matriz de confusão para o classificador GMM-MAP

As Tabelas A.7, A.8 e A.9 referem-se aos resultados experimentais do classificador GMM, de regra de decisão MAP, com vector característico composto por três componentes características: tamanho, desvio padrão dos pontos e velocidade estimada.

		Pos	Neg	Pos	Neg	Pos	Neg	Pos	Neg	Pos	Neg
$f(i)$	Classe	97	14	94	23	99	15	98	13	108	22
	Não-classe	48	380	12	309	13	326	10	276	14	268
		1-GMM		2-GMM		3-GMM		4-GMM		5-GMM	

Tabela A.10: Matriz de confusão usando regra MAP para a Classe-1: peões

		Pos	Neg	Pos	Neg	Pos	Neg	Pos	Neg	Pos	Neg
$f(i)$	Classe	273	84	218	34	155	12	106	12	106	25
	Não-classe	12	301	51	276	107	334	93	312	95	299
		1-GMM		2-GMM		3-GMM		4-GMM		5-GMM	

Tabela A.11: Matriz de confusão usando regra MAP para a Classe-2: carros/veículos

		Pos	Neg	Pos	Neg	Pos	Neg	Pos	Neg	Pos	Neg
$f(i)$	Classe	222	74	134	42	128	21	117	28	104	11
	Não-classe	13	255	103	269	134	297	115	257	93	272
		1-GMM		2-GMM		3-GMM		4-GMM		5-GMM	

Tabela A.12: Matriz de confusão usando regra MAP para a Classe-3: árvores/postes

As Tabelas A.10, A.11 e A.12 referem-se aos resultados experimentais do classificador GMM, de regra de decisão MAP, com vector característico composto por quatro componentes características (descritas na Secção 6.6).

# Bibliografia

- Adams, M. D. (2001). On-line gradient based surface discontinuity detection for outdoor scanning range sensors. In *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems*. USA.
- Amara, M., Brucq, D. de, Courtellemont, P., Wallon, P., Mesmin, C., Lecourtier, Y. (1996). A Recursive Estimation Of Parameters Of Straight Lines And Circles: Application To The Segmentation Of (The) Rey'S Complex Figure. *ICPR, B*, 467-471.
- ARGO, A. V. (1999). *The ARGO autonomous vehicle project*. [online]. [www.argo.ce.unipr.it/ARGO/](http://www.argo.ce.unipr.it/ARGO/), (accessed: 2006).
- Arulampalam, S., Maskell, S., Gordon, N., Clapp, T. (2001). A tutorial on particle filters for on-line non-linear/non-Gaussian Bayesian tracking. In *IEEE Transactions on Signal Processing* (Vol. 50, p. 174-188).
- Bar-Shalom, Y., Fortmann, T. (1988). *Tracking and data association*. Academic Press.
- Bar-Shalom, Y., Li, X. (1995). *Multitarget-multisensor tracking: principles & techniques*. YBS Publishing.
- Bento, L., Nunes, U. (2005). Autonomous navigation control with magnetic markers guidance of a cybernetic car using fuzzy logic. In *Journal of Nonlinear Dynamics and Chaos in Engineering Systems*. Japan (in press): Springer Netherlands.
- Bertozzi, M., Broggi, A., Fascioli, A., Tibaldi, A., Chapuis, R., Chausse, F. (2004). Pedestrian localization and tracking system with Kalman filtering. In *IEEE Intelligent Vehicles Symposium*.
- Bilmes, J. (1997). *A gentle tutorial on the EM algorithm and its application to parameter estimation for gaussian mixture and hidden markov models* (Technical Report ICSI-TR-97-021). International Computer Science Institute (ICSI), Berkeley, CA.

- Blackman, S. S. (1986). *Multiple-Target Tracking with Radar Applications*. Dedham, MA: Artech House.
- Borges, G. A., Aldon, M. J. (2004). Line extraction in 2D range images for mobile robotics. In *Journal of Intelligent & Robotic Systems* (Vol. 40, p. 267-297).
- CityMobil. (2006). *The European CityMobil project*. [online]. www.citymobil-project.eu, (accessed: 2006).
- Cybercars. (2001). *Cybernetic technologies for the cars in the city*. [online]. www.cybercars.org, (accessed: 2006).
- Dietmayer, K., Sparbert, J., Streller, D. (2001). Model based object classification and object tracking in traffic scenes from range images. In *IV IEEE Intelligent Vehicles Symposium*.
- Duda, R., Hart, P. (1973). *Pattern classification and scene analysis*. John Wiley and Sons, Inc, NY.
- Duda., R. O., Hart, P. E., Stork, D. G. (2001). *Pattern classification*. John Wiley & Sons, Inc, NY.
- Fitzgibbon, A., Pilu, M., Fisher, R. (1999). Direct least square fitting of ellipses. In *IEEE Trans. on Pattern Analysis and Machine Intelligence* (Vol. 21, p. 476-480).
- Fortmann, T., Bar-Shalom, Y., Scheffe, M. (1983). Sonar tracking of multiple targets using joint probabilistic data association. In *IEEE Journal of Oceanic Engineering* (Vol. 8, p. 173-184).
- Friedman, J., Hastie, T., Tibshirani, R. (2000). Additive logistic regression: a statistical view of boosting. In *The Annals of Statistics* (Vol. 38).
- Fuerstenberg, K., Willhoeft, V. (2001). Object tracking and classification using laserscanners - pedestrian recognition in urban environment. In *4th IEEE Intelligent Transportation Systems Conference (ITSC)* (p. 453-455). Oakland (CA), USA.
- Gamini, D., Newman, P., Clark, S., Durrant-Whyte, H., Csorba, M. (2001). A solution to the simultaneous localization and map building (SLAM) problem. In *IEEE Transactions on Robotics and Automation* (Vol. 17, p. 229-241).
- Gander, W., Golub, G., Strebel, R. (1994). Least-squares fitting of circles and ellipses.

- In *BIT* (Vol. 34, p. 558-578).
- Guivant, J., Nebot, E., Durrant-Whyte, H. (2000). Simultaneous localization and map building using natural features in outdoor environments. In *VI Intelligent Autonomous Systems*.
- Gustafsson, F., Gunnarsson, F., Bergman, N., Forsell, U., Jansson, J., Karlsson, R., et al. (2002). Particle filters for positioning, navigation, and tracking. In *IEEE Transaction on Signal Processing* (Vol. 50, p. 425-437).
- Julier, S., Uhlmann, J., DurrantWhyte, H. F. (2000). A new method for the nonlinear transformation of means and covariances in filters and estimators. In *IEEE Transaction on Automatic Control*.
- Kirubarajan, T., Bar-Shalom, Y. (2004). Probabilistic data association techniques for target tracking in clutter. In *Proceedings of the IEEE* (Vol. 92, p. 536-557).
- Kittler, J., Hatef, M., Duin, R., Matas, J. (1998). On combining classifiers. In *IEEE Trans. on Pattern Anal. and Machine Intelligent* (Vol. 20, p. 226-239).
- Lee, K. (2001). *Reactive navigation for an outdoor autonomous vehicle*. Tese de Mestrado, University of Sydney, Department of Mechanical and Mechatronic Engineering.
- Leonard, J. J., Durrant-Whyte, H. F. (1991). Mobile robot localization by tracking geometric beacons. In *IEEE Transaction on Robotics and Automation* (Vol. 7, p. 376-382).
- Marques, C., Cristovão, J., Lima, P., Frazão, J., Ribeiro, M. I., Ventura, R. (2006). RAPOSA: Semi-Autonomous Robot for Rescue Operations. In *Proc. of IROS2006 - IEEE/RSJ International Conference on Intelligent Robots and Systems*. Beijing, China.
- McLachlan, G., Krishnan, T. (1997). *The EM Algorithm and Extensions*. John Wiley and Sons, New York, NY.
- Mendes, A. (2004). *Detecção e seguimento de alvos com laser range finder*. Dissertação de mestrado, Universidade de Coimbra, Departamento de Engenharia Electrotécnica e de Computadores, Coimbra, Portugal.
- Mendes, A., Bento, L., Nunes, U. (2004). Multi-target detection and tracking with laser



- range finder. In *IEEE Intelligent Vehicles Symposium (IV 2004)*. Parma, Italy.
- Monteiro, G., Peixoto, P., Nunes, U. (2006). Vision-based pedestrian detection using Haar-like features. In *Proc. 6th National Festival of Robotics, Scientific Meeting (ROBOTICA)*.
- Monteiro, G., Premebida, C., Peixoto, P., Nunes, U. (2006). Tracking and Classification of Dynamic Obstacles Using Laser Range Scanner and Vision. In *Proc. of the IEEE/IOS Workshop on: Safe Navigation in Open and Dynamic Environments, IOS-2006*.
- Neira, J., Tardós, J., Horn, J., Schmidt, G. (1999). Fusing range and intensity images for mobile robot localization. In *IEEE Trans. Robotics and Automation* (Vol. 15, p. 76-84).
- Nguyen, V. T., Martinelli, A., Tomatis, N., Siegwart, R. (2005). A Comparison of line extraction algorithms using 2D laser rangefinder for indoor mobile robotics. In *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*.
- Nishii, R. (2003). Contextual image segmentation based on AdaBoost and Markov random fields. In *IEEE International Geoscience and Remote Sensing Symposium, IGARSS '03* (Vol. 6, p. 3507 - 3509).
- Paalanen, P., Kamarainen, J. K., Ilonen, J., Kälviäinen, H. (2005). *Feature representation and discrimination based on Gaussian mixture model probability densities - practices and algorithms* (Research Report-95). ([www.it.lut.fi/project/gmmbayes/](http://www.it.lut.fi/project/gmmbayes/)): Lappeenranta University of Technology, Dep. of Information.
- Parent, M. (2005). New Technologies for Sustainable Urban Transportation in Europe. In *Proceedings of 4th Asia Pacific Conference on Transportation and the Environment*.
- Premebida, C. (2005a). *Segmentation and geometric primitives extraction from 2D laser range data for mobile robots applications* (Technical Report N° ISRLM2005/02). University of Coimbra, Portugal: Instituto de Sistemas e Robótica (ISR).
- Premebida, C. (2005b). *Seguimento e Associação de Dados em Tarefas de Robótica Móvel Usando Informação de um LRF* (Technical Report N° ISRLM2005/05). University of Coimbra, Portugal: Instituto de Sistemas e Robótica (ISR).

- Premebida, C., Nunes, U. (2005). Segmentation and geometric primitives extraction from 2D laser range data for mobile robot applications. In *Proc. 5th National Festival of Robotics, Scientific Meeting (ROBOTICA)*.
- Premebida, C., Nunes, U. (2006). A multi-target tracking and GMM-classifier for intelligent vehicles. In *IEEE 9th Int. Conf. on Intelligent Transportation Systems (ITSC)*.
- Rekleitis, I. (2004). *A Particle filter tutorial for mobile robot localization* (IReport TR-CIM-04-02). McGill University, Montreal, Quebec, Canada: Centre for Intelligent Machines.
- Roumeliotis, S., Bekey, G. (2000). Segments: a Layered, dual Kalman filter algorithm for indoor feature extraction. In IEEE (Ed.), *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems* (p. 454-461).
- Santos, S., Faria, J., Soares, F., Araujo, R., Nunes, U. (2003). Tracking of multi-obstacles with laser range data for autonomous vehicles. In *Proc. 3rd National Festival of Robotics Scientific Meeting (ROBOTICA)* (p. 59-65). Lisbon, Portugal.
- Sarfraz, M., Rasheed, A., Muzaffar, Z. (2005). A Novel Linear Time Corner Detection Algorithm. In *CGIV '05: Proceedings of the International Conference on Computer Graphics, Imaging and Visualization (CGIV'05)* (p. 191-196). Washington, DC, USA: IEEE Computer Society.
- Siadat, A., Kaske, A., Klausmann, S., Dufaut, M., Husson, R. (1997). An optimized segmentation method for a 2D laser-scanner applied to mobile robot navigation. In *3rd IFAC Symposium on Intelligent Components and Instruments for Control Applications* (p. 153-158). France.
- Sparbert, J., Dietmayer, K., Steller, D. (2001). Lane detection and street type classification using laser range images. In *4th IEEE Intelligent Transportation Systems Conference (ITSC)*. Oakland (CA), USA.
- Streller, D., Dietmayer, K. (2004). Object tracking and classification using a multiple hypothesis approach. In *IEEE Intelligent Vehicles Symposium*.
- Streller, D., Dietmayer, K., Sparbert, J. (2001). Object tracking in traffic scenes with

- multi-hypothesis approach using laser range images. In *8th World Congress on Intelligent Transport Systems (ITSC)*. Sydney, Australia.
- Strobl, K., Sepp, W., Fuchs, S., Paredes, C., Arbter, K. (2005). *Camera Calibration Toolbox for Matlab*. [online]. [www.vision.caltech.edu/bouguetj/calibdoc](http://www.vision.caltech.edu/bouguetj/calibdoc), (accessed: 2006).
- ULTra. (2003). *Advanced transport systems*. [online]. [www.atsltd.co.uk](http://www.atsltd.co.uk), (accessed: 2006).
- Vandorpe, J., Brussel, H., Xu, H. (1996). Exact dynamic map building for a mobile robot using geometrical primitives produced by a 2D range finder. In *IEEE International Conference on Robotics and Automation*.
- Viola, P., Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. In *In IEEE Conference on Computer Vision and Pattern Recognition, CVPR* (Vol. 1).
- Wijesoma, W., Kodagoda, K., Balasuriya, A. (2002). A laser-vision sensing for road detection and reconstruction. In *IEEE 5th Int. Conf. on Intelligent Transportation Systems (ITSC)*.
- Wolf, D., Sukhatme, G., Fox, D., Burgard, W. (2005). Autonomous Terrain Mapping and Classification Using Hidden Markov Models. In *IEEE International Conference on Robotics and Automation (ICRA)*.
- Xavier, J., Pacheco, M., Castro, D., Ruano, A., Nunes, U. (2005). Fast line arc/circle and leg detection from laser scan data in a player driver. In *IEEE International Conference on Robotics and Automation (ICRA)*.
- Zhang, S., Xie, L., Adams, M. (2003). Geometrical feature extraction using 2D range scanner. In *Proc. of the IV Int. Conf. Control and Automation* (p. 901-905). Canada.
- Zhang, S., Xie, L., Adams, M. (2005). An Efficient Data Association Approach to Simultaneous Localization and Map Building. In *International Journal of Robotics Research* (Vol. 24, p. 49-60).
- Zhao, H., Shibasaki, R. (2005). A novel system for tracking pedestrians using multiple single-row laser-range scanners. In *IEEE Transactions on Systems, Man, and Cybernetics-Part A* (Vol. 35, p. 283-291).