

FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO



**FEUP**

# **Detecção automática de pedículos em imagens radiológicas da coluna vertebral**

**Pedro Jorge Morgado e Cunha**

Mestrado Integrado em Engenharia Informática e Computação

Orientador: Daniel Cardoso de Moura (Doutor)

Co-orientador: Jorge Manuel Gomes Barbosa (Prof. Doutor)

17 de Junho de 2011



# **Detecção automática de pedículos em imagens radiológicas da coluna vertebral**

**Pedro Jorge Morgado e Cunha**

Mestrado Integrado em Engenharia Informática e Computação

Aprovado em provas públicas pelo júri:

Presidente: Jaime dos Santos Cardoso (Prof. Doutor)

Vogal Externo: Jorge Manuel Fernandes dos Santos (Prof. Doutor)

Orientador: Daniel Cardoso de Moura (Doutor)

---



# Resumo

A avaliação de deformações da coluna, tal como a escoliose, requer uma estimativa tridimensional. No entanto, devido a limitações das técnicas imagiológicas 3D, a coluna é reconstruída a partir de radiografias planares. Os métodos de reconstrução de referência requerem um tempo de interacção muito elevado (>1h) para identificação das estruturas anatómicas de interesse. Recentemente foram propostos métodos semi-supervisionados baseados em modelos estatísticos que reduzem o tempo de interacção para apenas alguns minutos, no entanto, estes métodos têm mostrado dificuldade em determinar com precisão os pedículos presentes nas vértebras que são fundamentais para a obtenção de vários índices clínicos.

Nesta dissertação é proposto um novo método para a detecção de estruturas anatómicas baseado na utilização de descritores de imagem e de imagens radiológicas da coluna anotadas por um perito para o treino de um classificador binário. Este método testa as hipóteses de que os descritores de imagem são capazes de detectar padrões complexos, tais como os pedículos, e que os descritores baseados na distribuição do gradiente irão obter os melhores resultados.

A avaliação do método foi realizada recorrendo a um conjunto de imagens retiradas de radiografias da coluna vertebral de utentes com escoliose, tendo este conjunto sido dividido num sub-conjunto de treino e num sub-conjunto de teste. A avaliação foi realizada por comparação com uma marcação manual de um perito. Nesta dissertação foi seleccionado da bibliografia um conjunto de descritores e de classificadores utilizados em três experiências. A primeira para determinar qual a melhor combinação pré-processamento, descritor e classificador, uma segunda onde foi testado o impacto na detecção dos parâmetros do melhor descritor de entre os testados, e uma terceira onde foram utilizados os parâmetros óptimos e foi testado o impacto da técnica de pós-processamento proposta. De entre os classificadores e descritores testados, os melhores resultados foram obtidos pelo descritor histograma de gradientes orientados (*Histogram of Oriented Gradients* - HOG) com o classificador Máquinas de Vectores de Suporte, beneficiando tanto das técnicas de pré-processamento como do pós-processamento. No final da dissertação verificamos que os resultados obtidos foram superiores às expectativas, em que os melhores alcançaram uma distância média, em relação às coordenadas de referência dadas pelo técnico, com um valor dentro dos limites esperados para o erro que o operador comete. A hipótese de que os descritores de imagem são capazes de detectar padrões complexos verificou-se e concluímos que os baseados na distribuição do gradiente são os que proporcionam melhores resultados na detecção de pedículos.



# Abstract

Assessing spinal deformations, such as scoliosis, requires a 3D evaluation. However, due to restrictions of conventional 3D imaging techniques the 3D reconstructions are typically performed from 2D planar radiographs. The reconstruction methods require a large interaction time (>1h) for the identification of anatomical structures of interest. Recently, semi-supervised methods based on statistical models were proposed that enable to reduce interaction time to a few minutes. However, these methods have shown difficulties to determine precisely the pedicles of vertebrae, which are fundamental for calculating several clinical indices.

In this thesis it is proposed a new method for the detection of anatomical structures based in the use of feature descriptors and radiological images of the spine annotated by an expert to train a binary classifier. This method evaluates the hypothesis that feature descriptors are capable of detect complex patterns, such as the pedicles, and that the descriptors based in distribution of the gradient will get the best results.

The method evaluation was performed using images taken from spinal radiographs of patients with scoliosis which were divided into two sets, training and testing. The evaluation was performed by comparison with a manual identification from an expert. Several descriptors and classifiers were selected from the literature for conducting a series of 3 experiments. A first one to determine the best combination of pre-processing, descriptor and classifier, a second one where the impact on the detection of parameters of the best descriptor among those selected was tested, and the third one where we used the optimal parameters and tested the impact of the post-processing technique proposed. Among the classifiers and descriptors tested the best results were obtained using Histograms of Oriented Gradients (HOG) and Support Vector Machines (SVM), benefiting both from the pre-processing and post-processing stages. At the end of this dissertation it is possible to conclude that results overcome the initial expectations, with the best results achieving an average distance compared to the reference coordinates given by the expert with a value within the expected error of the expert. The hypothesis that feature descriptors are capable of detecting complex patterns was verified as well as the one stating that distribution-based descriptors are the most suitable for this application.



# Agradecimentos

O meu mais profundo agradecimento é dirigido ao Professor Daniel Moura, não só pela orientação científica deste trabalho, como pela confiança em mim depositada para a sua concretização, pela paciência, pelo entusiasmo e disponibilidade que sempre me dispensou. Queria também agradecer ao Professor Jorge Barbosa pela orientação científica e pela experiência que transmitiu nos seus conselhos. Gostaria também de agradecer à escola *Polytechnique Montréal* pelas imagens dispensadas para o meu estudo e aos responsáveis do LIACC pela utilização das instalações.

Aos meus pais e irmãos, pela suporte e encorajamento sempre dispensado. À minha família, pela sua tolerância, compreensão e carinho quando estava a escrever em vez de atender às suas necessidades.

Estou ainda em dívida para com muitas pessoas, principalmente colegas e professores, pela sua ajuda, apoio e paciência. E é por isso que quero dedicar esta Dissertação a todos aqueles que, sem reservas, partilharam comigo os seus conhecimentos.

Pedro Cunha



# Conteúdo

<b>1</b>	<b>Introdução</b>	<b>1</b>
1.1	Motivação e Objectivos . . . . .	2
1.2	Estrutura da dissertação . . . . .	3
<b>2</b>	<b>Revisão Bibliográfica</b>	<b>5</b>
2.1	Descritores de Imagem para detecção de padrões . . . . .	5
2.1.1	Descritores Baseados na Distribuição . . . . .	5
2.1.2	Técnicas de Frequência Espacial . . . . .	10
2.1.3	Descritores Diferenciais . . . . .	10
2.1.4	Outros descritores . . . . .	11
2.2	Classificadores . . . . .	15
2.2.1	Redes Neurais Artificiais . . . . .	17
2.2.2	Máquinas de Vectors de Suporte . . . . .	20
2.2.3	Naive Bayes . . . . .	26
2.3	Trabalho relacionado . . . . .	27
2.4	Sumário . . . . .	30
<b>3</b>	<b>Detecção Automática de Pedículos</b>	<b>31</b>
3.1	Pré-Processamento . . . . .	32
3.2	Treino . . . . .	34
3.3	Detecção . . . . .	39
3.4	Sumário . . . . .	40
<b>4</b>	<b>Implementação e Avaliação da Detecção</b>	<b>41</b>
4.1	Materiais . . . . .	41
4.2	Descritores . . . . .	42
4.2.1	Momentos Estatísticos . . . . .	42
4.2.2	Momentos Invariantes . . . . .	43
4.2.3	Matriz de co-ocorrências . . . . .	43
4.2.4	Histogramas de Gradientes Orientados - HOG . . . . .	43
4.2.5	<i>Speeded-Up Robust Features</i> . . . . .	45
4.2.6	Daisy . . . . .	46
4.3	Classificadores . . . . .	47
4.3.1	Naive Bayes . . . . .	47
4.3.2	Máquinas de Suporte Vectorial . . . . .	48
4.3.3	Redes Neurais . . . . .	49
4.4	Experiências . . . . .	50

## CONTEÚDO

4.4.1	Experiência 1: Avaliação dos descritores, classificadores e pré-processamento . . . . .	52
4.4.2	Experiência 2: Avaliação do impacto dos parâmetros do HOG . . .	53
4.4.3	Experiência 3: Avaliação do HOG com parâmetros otimizados e do pós-processamento . . . . .	54
4.5	Sumário . . . . .	54
<b>5</b>	<b>Resultados</b>	<b>55</b>
5.1	Experiência 1: Avaliação dos descritores, classificadores e pré-processamento	55
5.1.1	Naive Bayes . . . . .	55
5.1.2	Redes Neurais Artificiais . . . . .	60
5.1.3	Máquinas de Suporte Vectorial . . . . .	65
5.1.4	Discussão . . . . .	71
5.2	Experiência 2: Avaliação do impacto dos parâmetros do HOG . . . . .	71
5.2.1	Tamanho do bloco da imagem . . . . .	71
5.2.2	Nível de Sobreposição . . . . .	72
5.2.3	Tamanho do bloco do HOG . . . . .	74
5.2.4	Tipo de Normalização do HOG . . . . .	75
5.2.5	Número de Orientações do HOG . . . . .	77
5.2.6	Discussão . . . . .	79
5.3	Experiência 3: Avaliação do HOG com parâmetros otimizados . . . . .	79
5.3.1	Discussão . . . . .	83
5.4	Análise Temporal . . . . .	83
5.5	Sumário . . . . .	84
<b>6</b>	<b>Conclusão</b>	<b>85</b>
6.1	Trabalho Futuro . . . . .	86
	<b>Referências</b>	<b>89</b>

# Lista de Figuras

1.1	Exemplos de radiografias com os pedículos anotados. . . . .	2
2.1	Exemplo do cálculo do descritor HOG . . . . .	6
2.2	Cálculo da orientação no SURF . . . . .	8
2.3	Cálculo do descritor SURF . . . . .	8
2.4	Cálculo do descritor Daisy . . . . .	9
2.5	Cálculo da matriz de co-ocorrências . . . . .	14
2.6	Processo para obter descritor SSIM . . . . .	15
2.7	Exemplos de aplicação do descritor SSIM . . . . .	16
2.8	Neurónio - elemento básica de uma rede neuronal artificial . . . . .	17
2.9	Tipos de rede neuronal artificial . . . . .	18
2.10	Tipos de ligação de uma rede neuronal artificial . . . . .	18
2.11	Cálculo da distância $d$ entre os hiperplanos $H1$ e $H2$ , no contexto de uma SVM [LdC08]. . . . .	22
2.12	Exemplo de transformação de conjunto de dados não-linear para um novo espaço de características [LdC08]. . . . .	24
2.13	Exemplo de detecção de uma pessoa, utilizando o Histograma de Gradientes Orientados . . . . .	27
2.14	Resultado do trabalho de Duong <i>et al.</i> , para a detecção da coluna utilizando um descritor baseado em momentos estatísticos. . . . .	28
2.15	Resultado do trabalho de Chen <i>et al.</i> , para emparelhamento de imagens da retina utilizando o descritor PIIFD. . . . .	29
3.1	Visão geral do método proposto para a detecção dos pedículos. . . . .	32
3.2	Comparação entre imagem original e imagem filtrada. . . . .	33
3.3	Comparação entre imagem original e imagem com rotação eliminada. . . . .	34
3.4	Exemplos de artefactos presentes nas imagens radiológicas. . . . .	35
3.5	Diferentes configurações na divisão em blocos, para o treino do classificador. . . . .	35
3.6	Exemplo de curva ROC [PL04]. . . . .	37
3.7	Exemplo de varrimento numa imagem de treino. . . . .	39
3.8	Gráfico 3D com o resultado do classificador. . . . .	40
4.1	Exemplo de configuração do HOG. . . . .	44
4.2	Descritor Daisy e representação gráfica dos seus parâmetros. . . . .	47
4.3	Exemplo de rede neuronal utilizada com três camadas. . . . .	49
4.4	Exemplo do gráfico <i>accuracy</i> em função do raio do erro do operador. . . . .	51

## LISTA DE FIGURAS

5.1	Gráficos da <i>accuracy</i> em função do raio do erro do operador para o Naive Bayes no primeiro cenário. . . . .	56
5.2	Diagramas de extremos e quartis da distância para o Naive Bayes no primeiro cenário. . . . .	57
5.3	Gráficos da <i>accuracy</i> em função do raio do erro do operador para o Naive Bayes no segundo cenário. . . . .	58
5.4	Diagramas de extremos e quartis da distância para o Naive Bayes no segundo cenário. . . . .	58
5.5	Gráficos da <i>accuracy</i> em função do raio do erro do operador para o Naive Bayes no terceiro cenário. . . . .	59
5.6	Diagramas de extremos e quartis da distância para o Naive Bayes no terceiro cenário. . . . .	60
5.7	Gráficos da <i>accuracy</i> em função do raio do erro do operador para o RNA no primeiro cenário. . . . .	61
5.8	Diagramas de extremos e quartis da distância para o RNA no primeiro cenário. . . . .	61
5.9	Gráficos da <i>accuracy</i> em função do raio do erro do operador para o RNA no segundo cenário. . . . .	62
5.10	Diagramas de extremos e quartis da distância para o RNA no segundo cenário. . . . .	63
5.11	Gráficos da <i>accuracy</i> em função do raio do erro do operador para o RNA no terceiro cenário. . . . .	64
5.12	Diagramas de extremos e quartis da distância para o RNA no terceiro cenário. . . . .	64
5.13	Gráficos da <i>accuracy</i> em função do raio do erro do operador para o SVM no primeiro cenário. . . . .	65
5.14	Diagramas de extremos e quartis da distância para o SVM no primeiro cenário. . . . .	66
5.15	Gráficos da <i>accuracy</i> em função do raio do erro do operador para a SVM no segundo cenário. . . . .	67
5.16	Diagramas de extremos e quartis da distância para a SVM no segundo cenário. . . . .	68
5.17	Gráficos da <i>accuracy</i> em função do raio do erro do operador para a SVM no terceiro cenário. . . . .	68
5.18	Diagramas de extremos e quartis da distância para a SVM no terceiro cenário. . . . .	69
5.19	Gráficos da <i>accuracy</i> em função do raio do erro do operador para a comparação entre o tamanho do bloco da imagem. . . . .	72
5.20	Diagramas de extremos e quartis da distância para a comparação entre o tamanho do bloco da imagem. . . . .	73
5.21	Gráficos da <i>accuracy</i> em função do raio do erro do operador para a comparação entre o nível de sobreposição do bloco da imagem. . . . .	73
5.22	Diagramas de extremos e quartis da distância para a comparação entre o nível de sobreposição do bloco da imagem. . . . .	74
5.23	Gráficos da <i>accuracy</i> em função do raio do erro do operador para a comparação entre o tamanho do bloco do HOG. . . . .	75

## LISTA DE FIGURAS

5.24	Diagramas de extremos e quartis da distância para a comparação entre o tamanho do bloco do HOG. . . . .	76
5.25	Gráficos da <i>accuracy</i> em função do raio do erro do operador para a comparação entre a normalização utilizada no HOG. . . . .	76
5.26	Diagramas de extremos e quartis da distância para a comparação entre a normalização utilizada no HOG. . . . .	77
5.27	Gráficos da <i>accuracy</i> em função do raio do erro do operador para a comparação entre o número de orientações ( <i>bins</i> ) do HOG. . . . .	78
5.28	Diagramas de extremos e quartis da distância para a comparação entre o número de orientações ( <i>bins</i> ) do HOG. . . . .	79
5.29	Gráficos da <i>accuracy</i> em função do raio do erro do operador para a comparação entre os melhores resultados obtidos. . . . .	80
5.30	Diagramas de extremos e quartis da distância para a comparação entre os melhores resultados obtidos. . . . .	81
5.31	Histograma do Ranking para a melhor configuração alcançada. . . . .	81
5.32	<i>Accuracy</i> em função do sigma no filtro de pós-processamento. . . . .	82
5.33	Média e Mediana da distância em função do sigma no filtro de pós-processamento. . . . .	82

## LISTA DE FIGURAS

# Lista de Tabelas

3.1	Matriz de Confusão . . . . .	36
4.1	Parâmetros do Histograma de Gradientes Orientados . . . . .	44
4.2	Tipos de normalização no Histograma de Gradientes Orientados . . . . .	45
4.3	Parâmetros do Daisy . . . . .	46
4.4	Tipos de normalização no Daisy . . . . .	46
4.5	Tipos de SVM disponíveis na LIBSVM . . . . .	48
4.6	Tipos de <i>kernel</i> disponíveis na LIBSVM . . . . .	49
5.1	Sumário dos resultados obtidos na primeira experiência para o classifica- dor Naive Bayes. . . . .	70
5.2	Sumário dos resultados obtidos na primeira experiência para o classifica- dor Redes Neurais Artificiais. . . . .	70
5.3	Sumário dos resultados obtidos na primeira experiência para o classifica- dor Máquinas de Suporte Vectorial. . . . .	70

## LISTA DE TABELAS

# Abreviaturas e Símbolos

2D	Duas Dimensões
3D	Três Dimensões
AUC	Área sob a curva
GLOH	Gradient Location-Orientation Histogram
HOG	Histogram of Oriented Gradients
MCO	Matriz de co-ocorrências
PIIFD	Partial Intensity Invariant Feature Descriptor
RBF	Radial-Basis Function
RNA	Redes Neurais Artificiais
SIFT	Scale-Invariant Feature Transform
SSIM	Self-Similarity
SURF	Speeded-Up Robust Features
SVM	Máquinas de Vetores de Suporte

## ABREVIATURAS E SÍMBOLOS

# Capítulo 1

## Introdução

A coluna vertebral é uma das principais estruturas de sustentação do corpo humano e é formada por um conjunto de vértebras sobrepostas que unem a cabeça à região pélvica. O principal método de avaliação clínica de deformações na coluna é a imagem radiológica planar (duas dimensões), pois é a única que permite visualizar a coluna com o utente em posição de pé (requisito necessário para avaliação de deformações da coluna) [CPE02] [GD04]. O trabalho aqui descrito surge no âmbito de um outro, cujo objectivo é a reconstrução a três dimensões da coluna vertebral utilizando radiografias planares da coluna [MBBT09] [MBB<sup>+</sup>11]. O trabalho de Moura *et al.* utiliza imagens anotadas por peritos e modelos estatísticos para a reconstrução da coluna e é actualmente o método semi-supervisionado de reconstrução 3D com melhor exactidão a localizar as vértebras e com menor tempo de interacção com o utilizador (<2 minutos). O processo de reconstrução permite o cálculo de diversos índices clínicos [Sto94], como por exemplo o ângulo de Cobb, que são de extrema importância para a avaliação de deformações na coluna, como a escoliose. No entanto, métodos semi-supervisionados [MBBT09] [MBB<sup>+</sup>11] [DML<sup>+</sup>03] mostram dificuldades na localização dos pedículos das vértebras, que são estruturas anatómicas importantes para a avaliação de deformações da coluna [Sto94]. Actualmente, os métodos que permitem a localização dos pedículos com maior exactidão como o proposto em [ADD<sup>+</sup>95] são extremamente supervisionados, requerendo um longo tempo de interacção e um técnico experiente.

Neste trabalho é proposto um método de detecção de estruturas anatómicas que envolve a utilização de descritores de imagem e classificadores supervisionados. No método é explorado o efeito do pré-processamento das imagens e o modo como o tipo de descritor e classificador influencia os resultados. Este método foi utilizado para a detecção dos pedículos, estruturas localizados nas extremidades das vértebras, testando diferentes combinações de descritor-classificador com o objectivo de verificar qual que melhor se

adequa a este problema.

## 1.1 Motivação e Objectivos

O método para a detecção de estruturas anatómicas tem como objectivo utilizar informação da imagem para detectar a localização de estruturas anatómicas da coluna vertebral, nomeadamente os pedículos que tipicamente não são bem detectados nos métodos semi-automáticos. Um método como o que é proposto para a detecção das estruturas anatómicas pode não só fazer reduzir a interacção com o utilizador como melhorar os resultados de métodos semi-automáticos. A automatização tornou-se possível pela capacidade de detecção das regiões de interesse na imagem, utilizando uma abordagem semi-supervisionada baseada em modelos estatísticos, previamente proposta em [MBBT09]. Assim, para cada pedículo é conhecida uma região de interesse onde ele estará contido e sobre a qual poderá ser feita uma pesquisa tendo em vista encontrar a sua localização.

O objectivo principal é a detecção automática de padrões nas imagens que indiquem a localização de estruturas anatómicas de interesse. Em particular, pretende-se detectar os pedículos localizados nas extremidades das vértebras em radiografias frontais da coluna, dada uma área de interesse na radiografia. Na figura 1.1 é ilustrado um exemplo do tipo de radiografia que será utilizado com a localização dos pedículos assinalada.

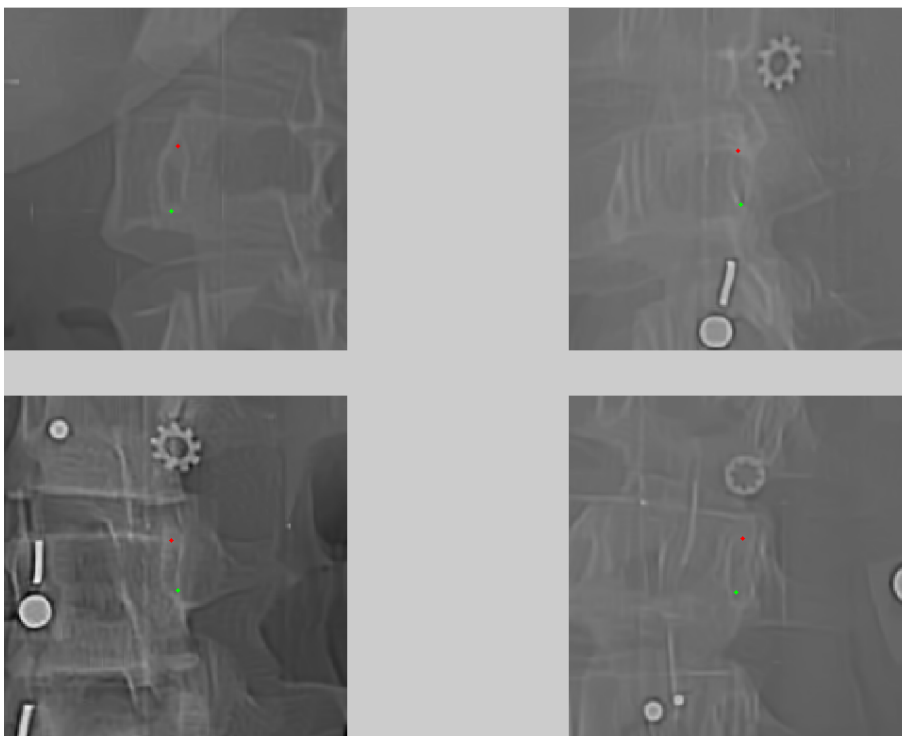


Figura 1.1: Exemplos de radiografias com os pedículos anotados. Ponto vermelho extremidade superior e ponto verde extremidade inferior.

Neste trabalho testa-se a hipótese que os descritores de imagem são capazes de detectar padrões complexos, como as estruturas anatómicas da coluna vertebral, nomeadamente os pedículos. Outra hipótese levantada é que descritores que se baseiam na distribuição do gradiente obtêm melhores resultados, devido à sua capacidade para descrever a forma dos objectos na imagem, enquanto outros descritores se focam principalmente na textura. É ainda levantada a hipótese de que as técnicas de pré-processamento e pós-processamento têm elevada importância nos resultados obtidos.

### **1.2 Estrutura da dissertação**

Este documento encontra-se dividido em diversos capítulos. Neste primeiro capítulo é feita uma breve introdução à dissertação, indicando o seu enquadramento, a sua motivação e os seus objectivos. No capítulo 2, será apresentada a revisão bibliográfica, dividido em três sub-seções. Na primeira são apresentados diferentes descritores de imagem, divididos em categorias. Na segunda são dados alguns exemplos de classificadores e na terceira são descritos trabalhos relacionados e exemplos de aplicação de descritores de imagem e classificadores. No capítulo 3 é feita uma descrição pormenorizada do método proposto com as suas várias fases que incluem o pré-processamento, o treino e o teste. No capítulo 4 são apresentados os detalhes de implementação do método, incluindo que descritores e classificadores foram seleccionados, uma descrição das experiências que foram efectuadas e as métricas propostas para a avaliação dos resultados. No capítulo 5 são apresentados os resultados obtidos com o recurso a gráficos e tabelas para as várias experiências realizadas. Por fim, no capítulo 6, são apresentadas algumas conclusões obtidas nesta dissertação e são indicadas futuras direcções de investigação.

## Introdução

## Capítulo 2

# Revisão Bibliográfica

Neste capítulo será feita uma revisão bibliográfica sobre o tema deste trabalho. Será dividido em três sub-seções, sendo que na primeira será feita uma revisão sobre descritores de imagem. Na segunda, serão abordados classificadores que serão utilizados na identificação dos pedículos. No final desta revisão bibliográfica, será feita uma apresentação dos trabalhos relacionados e de exemplos de aplicação de descritores de imagem na detecção de padrões e em imagens médicas.

### 2.1 Descritores de Imagem para detecção de padrões

Um descritor de imagem é um vector de valores numéricos que descreve uma região da imagem e pode ser comparado com um outro recorrendo a uma medida de semelhança. Existem diversas técnicas para descrever uma região de uma imagem e, a mais simples, é um vector constituído pelos pixels da imagem e pode ser utilizada a correlação cruzada para calcular um valor de similaridade entre os dois descritores. Devido ao elevado número de dimensões envolvidas, esta técnica resulta numa complexidade computacional muito elevada para o reconhecimento, sendo maioritariamente utilizada para encontrar correspondências entre duas imagens. Existem várias categorias de descritores, entre as quais os descritores baseados na distribuição (*Distribution-based Descriptors*), técnicas de frequência espacial (*Spatial-Frequency Techniques*), descritores diferenciais (*Differential Descriptors*) e outros, de acordo com Mikolajczyk e Schmid [MS05].

#### 2.1.1 Descritores Baseados na Distribuição

Esta classe de descritores é baseada na utilização de histogramas com o objectivo de representar diferentes características da imagem, como a sua forma. Um exemplo simples

desta classe é recorrer à distribuição das intensidades dos pixels da imagem, representados por um histograma. Um descritor desta classe foi apresentado por Lazebnik e Schmid [LSP05] e consiste num descritor baseado na intensidade e que é invariante à rotação, denominado *intensity-domain spin images*. Este é inspirado no método de correspondência de dados descrito por Johnson e Hebert [JH99]. Tem por base um histograma de duas dimensões que codifica a distribuição do brilho da imagem na vizinhança de um ponto de referência. As dimensões do histograma são  $d$ , a distância ao ponto de referência e  $i$  o valor da intensidade. A “parte” da imagem correspondente a um  $d$  fixo é simplesmente o histograma dos valores de intensidade dos pixels localizados a uma distância  $d$  do ponto de referência central.

Um descritor robusto a alterações de iluminação foi proposto por Zabih e Woodfill [ZW94]. Utiliza histogramas de ordenação e, também se baseia, nas relações recíprocas entre os valores de intensidade dos pixels. Este descritor é aplicável à representação de textura, embora seja necessário um elevado número de dimensões para ser robusto [OPM02].

O *scale-invariant feature transform*, SIFT, foi proposto por Lowe [Low04] e consiste num método para extrair características invariantes distintas de imagens que podem ser utilizadas para efectuar uma correspondência entre diferentes vistas de um objecto. Este método combina um detector de regiões invariante à escala e um descritor baseado na distribuição do gradiente dessas regiões. O histograma utilizado no descritor do SIFT, é denominado histograma de gradientes orientados (*histogram of oriented gradients*), HOG, e já foi demonstrado, por Dalal e Triggs [DT05], ser eficaz na detecção de padrões complexos. Na figura 2.1 é apresentado um exemplo do cálculo do HOG. A primeira etapa consiste em

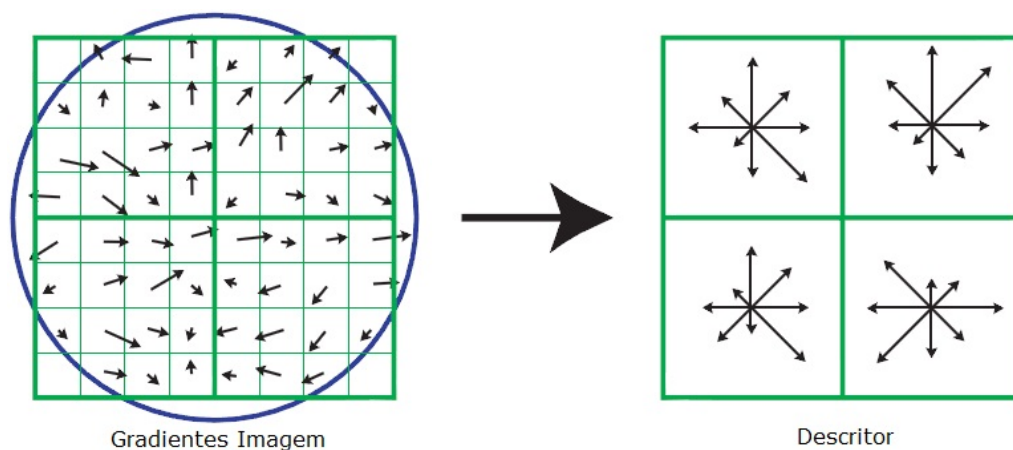


Figura 2.1: Exemplo do cálculo do descritor HOG

calcular a magnitude e a orientação do gradiente para cada ponto na região centrada num ponto chave, utilizando a sua escala para seleccionar o nível de suavização Gaussiana para a imagem. Para obter invariância relativamente à orientação, as coordenadas do descritor

e as orientações do gradiente são rodadas relativamente à orientação do ponto chave. Na fase seguinte é utilizada uma função de peso gaussiana com  $\sigma$  igual a um meio da largura da janela para atribuir o peso à magnitude de cada ponto. O objectivo da utilização desta função é evitar alterações bruscas ao descritor com pequenas variações na posição da janela e dar uma menor importância aos pontos longe do centro. Do lado direito da imagem 2.1 podemos ver o descritor. Consiste numa sub-região de  $4 \times 4$  que acumula os gradientes em histogramas de orientação com 8 direcções, em que o valor de cada uma das setas representa a magnitude do histograma nessa direcção. O descritor é então formado por um vector contendo todos os valores dos histogramas, correspondentes ao tamanho de cada uma das setas. No exemplo dado, os histogramas orientados formam um *array* de  $2 \times 2$ , embora este tamanho seja variável. O tamanho  $n$  da região de  $n \times n$  dos histogramas orientados e o número de direcções  $d$  a calcular são os parâmetros utilizados para variar a complexidade do descritor sendo o seu tamanho igual a  $dn^2$ . Segundo Lowe, [Low04] quanto maior for o tamanho do descritor maior será a sua capacidade de discriminar em conjuntos grandes sendo, no entanto, mais susceptível a distorções na forma e a oclusões. Baseados no descritor do SIFT surgiram muitos outros descritores como o histograma de gradientes de localização-orientação (*gradient location-orientation histograms* - GLOH) que calculam o descritor do SIFT utilizando uma grelha de localização polar logarítmica com 3 direcções radiais e 8 direcções angulares, resultando em 17 componentes. Obtém-se assim um descritor com um histograma de 272 componentes, cujo tamanho é reduzido recorrendo a uma técnica de PCA (*principal component analysis*). Este descritor foi proposto por Mikolajczyk e Schmid [MS05].

O SURF (*Speeded-Up Robust Features*) [BTVG06] é outro descritor com base no proposto no método SIFT. Este descritor é baseado nas respostas wavelet de primeira ordem de Haar em  $x$  e em  $y$ , em vez do gradiente, explorando imagens integrais para o aumento de velocidade e utilizando apenas 64 dimensões. Segundo os autores do SURF, esta utilização diminui o tempo de cálculo e aumenta a robustez. Para além disso, propõem um novo passo na indexação baseado no sinal da Laplaciana que aumenta a sua robustez. O cálculo do descritor consiste em vários passos, começando pela atribuição da orientação, para permitir a sua invariância relativa a rotações. Este passo recorre ao cálculo das respostas wavelet de Haar em  $x$  e  $y$  numa vizinhança, de raio  $6s$ , do ponto de interesse. Após o cálculo das respostas e do peso, utilizando uma Gaussiana ( $\sigma = 2s$ ) centrada no ponto de interesse, as respostas são representadas como pontos no espaço, com a força da resposta horizontal na abcissa e a força da resposta vertical na ordenada. A orientação dominante é estimada calculando a soma de todas as respostas com uma janela de orientação deslizante de tamanho  $\frac{\pi}{3}$ , parametrizável, como podemos ver na figura 2.2. As respostas horizontal e vertical são somados e, das duas somas, obtemos um vector de orientação local. O maior define a orientação do ponto de interesse. Para a extracção do descritor é obtida uma região quadrada centrada no ponto de interesse e orientada na

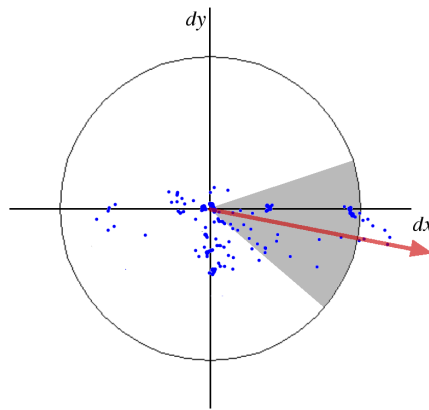


Figura 2.2: Cálculo da orientação no SURF

direcção escolhida previamente. O tamanho desta janela é de 20s. De seguida, é feita uma divisão em sub-regiões de  $4 \times 4$ . Esta operação preserva informação espacial importante. Para cada sub-região são computadas as respostas wavelet de Haar em pontos de amostra de  $5 \times 5$ , espaçados regularmente. Por simplicidade denomina-se  $d_x$ , a resposta na direcção horizontal e  $d_y$  na direcção vertical. Na figura 2.3 vemos ilustrado o processo de cálculo do descritor. Posteriormente, as respostas em cada sub-região são somadas

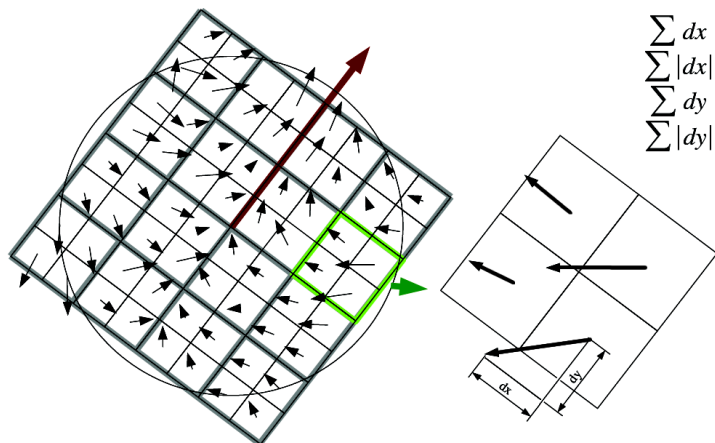


Figura 2.3: Cálculo do descritor SURF

e formam as primeiras entradas no vector do descritor. Para aumentar a informação sobre a polaridade das mudanças de intensidade, os autores do SURF também extraíram a soma dos valores absolutos das respostas, obtendo um vector de quatro dimensões,  $v = (\sum d_x, \sum d_y, \sum |d_x|, \sum |d_y|)$ . Reunindo todas as sub-regiões, obtém-se o vector do descritor de comprimento 64. A parametrização do SURF apresentada foi a proposta pelos seus autores que testaram diversas configurações ao nível do tamanho das sub-regiões e também utilizaram uma versão com mais valores calculados (SURF-128).

O Daisy, foi proposto por Tola *et al.* [TLF10] [TLF08] e é um descritor muito eficiente de calcular de forma densa. Este descritor é inspirado no SIFT [Low04] e no GLOH [MS05], mas pode ser calculado de forma mais eficiente no domínio utilizado pelos seus autores. Em comparação com o SURF [BTVG06], o Daisy não introduz artefactos que afectam a sua performance, quando calculados densamente, segundo Tola *et al.* [TLF10] [TLF08]. O processo de computação do Daisy começa por calcular  $H$ , número de *maps* de orientação,  $G_i$ ,  $1 \leq i \leq H$ , um para cada uma das direcções, em que  $G_o(u, v)$  é igual à norma do gradiente da imagem em  $(u, v)$ , para a direcção  $o$  se este for maior que zero, caso contrário é zero. Este cálculo preserva a polaridade das mudanças de intensidade. Depois cada um dos *maps* de orientação é convuluído diversas vezes com *kernels* Gaussianos de diferentes valores de sigma ( $\Sigma$ ) para obter *maps* de orientação convuluídos. Os diferentes  $\Sigma$  são utilizados para controlar o tamanho da região. Na figura 2.4 podemos ver o descritor Daisy, e em cada localização de um pixel, o descritor consiste num vector constituído pelos valores dos *maps* de orientação convuluídos localizados em círculos concêntricos centrados na localização, e em que a quantidade de suavização Gaussiana é proporcional ao raio dos círculos. Como se pode ver na figura isto dá um aspecto de flor ao descritor que origina o seu nome. Este descritor utiliza uma grelha circular em

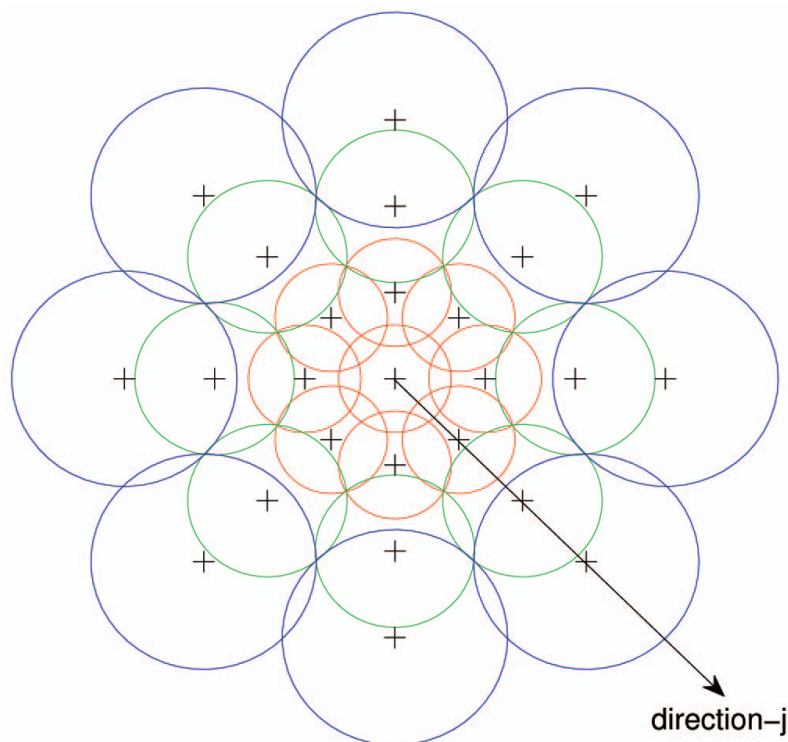


Figura 2.4: Cálculo do descritor Daisy

vez da grelha regular do SIFT dado que foi demonstrado ter melhores propriedades de localização [MS05]. Neste aspecto o Daisy aproxima-se mais ao GLOH do que ao SIFT.

A combinação de um *kernel* Gaussiano isotrópico com a grelha circular torna o descritor mais resistente a perturbações de rotação. A sobreposição das regiões garante um descritor com alterações suaves ao longo do eixo de rotação e o aumento desta sobreposição pode tornar o Daisy mais robusto até ao ponto em que começa a perder a sua capacidade de discriminação. Ao nível de parâmetros, o descritor possui seis: o raio ( $R$ ), o número de convoluções ( $Q$ ), o número de histogramas numa única camada ( $T$ ), número de direcções de cada histograma ( $H$ ), a possibilidade de interpolação espacial ( $SI$ ) e interpolação em camadas ( $LI$ ) e, por fim, o tipo de normalização ( $NT$ ).

Para terminar esta categoria existem outros dois descritores: histograma geométrico [ATRB95] e contexto da forma [BMP02]. As suas ideias são muito semelhantes às do SIFT e ambos calculam um histograma que descreve a distribuição de uma aresta numa região.

### 2.1.2 Técnicas de Frequência Espacial

Nesta classe de descritores são utilizadas técnicas que descrevem o conteúdo das frequências de uma imagem. Recorre-se às transformadas de Fourier para permitir analisar uma imagem no domínio das frequências, possibilitando a sua análise com técnicas de filtragem digitais. Contudo, com esta representação é difícil obter uma aproximação local de uma região. A utilização da transformada de Gabor [Gab46] ultrapassa esta limitação e baseia-se na utilização de filtros de Gabor para capturar alterações na frequência e na orientação. A utilização de filtros de Gabor pode ter duas aproximações: a expansão de Gabor e a filtragem de Gabor [KKK04]. Na primeira aproximação, as funções de Gabor representam áreas de informação da imagem distintas e necessitam da computação de funções de análise que é uma tarefa custosa. A segunda aproximação, utiliza as funções de Gabor como filtros de análise no processamento da imagem. A combinação de diversas respostas de filtros de Gabor permite representar objectos complexos e esta deve ser parametrizada. Para obter o descritor é construída uma matriz num ponto da imagem, recorrendo aos filtros de Gabor e pode ser utilizada por um classificador.

### 2.1.3 Descritores Diferenciais

Neste tipo de descritores são utilizadas as derivadas da imagem até uma certa ordem para aproximar a vizinhança de um ponto de interesse.

Freeman e Adelson [FA91] desenvolveram os filtros dirigidos (*steerable filters*). Os criadores utilizam o termo “filtro dirigido” para descrever uma classe de filtros, em que um filtro de uma orientação arbitrária é sintetizado como uma combinação linear de um conjunto de “filtros de base”. Neste descritor são calculadas todas as derivadas gaussianas até às de quarta ordem que são as utilizadas no descritor.

Ainda nesta categoria, enquadram-se os filtros complexos propostos por Baumberg [Bau00] e por Schaffalitzky e Zisserman [SZ02]. Utilizam os filtros derivados da família  $K(x, y, \Theta) =$

$f(x,y)\exp(i\Theta)$ , em que  $\Theta$  representa a orientação. Baumberg distingue-se por utilizar derivadas Gaussianas na função  $f(x,y)$ , enquanto que Schaffalitzky e Zisserman utilizam uma função polinomial. Os filtros complexos são derivados da equação  $K_{m,n}(x,y) = (x+iy)^m + (x-iy)^n G(x,y)$ . Este tipo de filtros distingue-se das derivadas Gaussianas pela alteração das coordenadas lineares no domínio da resposta do filtro.

#### 2.1.4 Outros descritores

O conceito de momentos invariantes 2D foi introduzido por Hu [Hu62]. O momento 2D ( $m$ ) de ordem  $p+q$  de uma imagem digital  $f(x,y)$  é definido na equação 2.1.

$$m_{pq} = \sum_x \sum_y x^p y^q f(x,y), p, q = 0, 1, 2, \dots \quad (2.1)$$

O momento central ( $\mu$ ) correspondente é definido em 2.2.

$$\mu_{pq} = \sum_x \sum_y (x-\bar{x})^p (y-\bar{y})^q f(x,y) \quad (2.2)$$

Na equação 2.2,  $\bar{x} = \frac{m_{10}}{m_{00}}$  e  $\bar{y} = \frac{m_{01}}{m_{00}}$ . O momento central normalizado ( $\eta$ ) é representado em 2.3.

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^\gamma}, p, q = 0, 1, 2, \dots \quad (2.3)$$

Na equação anterior, 2.3,  $\gamma = \frac{p+q}{2} + 1, p+q = 2, 3, \dots$ . Partindo das equações anteriores é possível definir um conjunto de 7 momentos invariantes 2D, que dão origem a um descritor invariante à rotação, translação, mudança de escala e simetria. Estes momentos são definidos nas equações 2.4 a 2.10.

$$\phi_1 = \eta_{20} + \eta_{02} \quad (2.4)$$

$$\phi_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \quad (2.5)$$

$$\phi_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \quad (2.6)$$

$$\phi_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \quad (2.7)$$

$$\phi_5 = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] \\ + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \quad (2.8)$$

$$\phi_6 = (\eta_{20} - \eta_{02})[(\eta_{30} \\ + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03}) \quad (2.9)$$

$$\phi_7 = (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] \\ + (3\eta_{12} - \eta_{30})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \quad (2.10)$$

Com o objectivo de reduzir o intervalo de variação dos momentos invariantes e, com isso, simplificar a sua interpretação, Gonzalez *et al.* [GWE04] sugerem que os valores sejam escalados segundo a equação 2.11.

$$\text{sgn}(\phi_i) \log_{10}(|\phi_i|) \quad (2.11)$$

Duong *et al.* no seu trabalho [DCL09], que será descrito com maior detalhe na secção 2.3, utilizou um descritor que consiste num conjunto de medidas baseadas em momentos estatísticos aplicados em histogramas locais. São descritos por Gonzalez *et al.* [GWE04]. A computação destes momentos é derivada da equação geral de um momento estatístico, permitindo o cálculo do  $n$ -ésimo momento de um variável aleatória  $z$  sobre a intensidade média  $m$ . O descritor consiste em 6 medidas, em que a primeira é a intensidade média ( $m$ ) e é apresentada na equação 2.12.

$$m = \sum_{i=0}^{L-1} z_i p(z_i) \quad (2.12)$$

Em 2.12,  $z$  é uma variável aleatória que representa os valores de cinzento,  $p(z_i)$  é o histograma normalizado dos pixels do bloco considerado e  $L$  é o número de níveis de cinzento distintos. O segundo momento é uma medida de contraste ( $\sigma$ ). É descrito na equação 2.14, em que  $\mu_2$  é o segundo momento derivado de 2.13.

$$\mu_n = \sum_{i=0}^{L-1} (z_i - m)^n p(z_i) \quad (2.13)$$

$$\sigma = \sqrt{\mu_2(z)} \quad (2.14)$$

O terceiro momento é uma medida de suavidade (*smoothness*),  $R$ , e é descrita na equação 2.15.

$$R = 1 - \frac{1}{(1 + \sigma^2)} \quad (2.15)$$

A quarta dimensão do descritor é a assimetria (*skewness*) do histograma definido pelo terceiro momento estatístico, presente na equação 2.16.

$$\mu_3 = \sum_{i=0}^{L-1} (z_i - m)^3 p(z_i) \quad (2.16)$$

O quinto momento é a uniformidade ( $U$ ) e é obtido através de 2.17.

$$U = \sum_{i=0}^{L-1} p^2(z_i) \quad (2.17)$$

A última dimensão do descritor é a entropia ( $e$ ), medida de aleatoriedade, que caracteriza a região. A sua fórmula de cálculo é apresentada em 2.18.

$$e = - \sum_{i=0}^{L-1} p(z_i) \log_2 p(z_i) \quad (2.18)$$

As diferentes dimensões do descritor foram avaliadas utilizando análise multi-dimensional e inspecção visual confirmando a sua capacidade de discriminação para a classificação de imagens [DCL09].

Van Goolen *et al.* [VGMU96] utilizaram o conceito de momentos invariantes generalizados para descrever a natureza multi-espectral dos dados de uma imagem. Os momentos invariantes combinam os momentos centrais definidos por  $M_{pq}^a = \int \int_{\Omega} x^p y^q [I(x,y)]^a dx dy$  de ordem  $p+q$  e de grau  $a$ . O momento é uma medida estatística que caracteriza a forma e a distribuição da intensidade na região  $\Omega$ . Eles são independentes e podem ser calculados para diferentes graus e ordens. A utilização do invariante reduz o número de dimensões. Mikolajczyk e Schmid [MS05] utilizaram no seu estudo, um descritor em que os momentos foram calculados com  $M_{pq}^a = \frac{1}{xy} \sum_{x,y} x^p y^q [I_d(x,y)]^a$  em que  $I_d$  representa o gradiente da imagem na direcção  $d$ . As derivadas são calculadas nas direcções  $x$  e  $y$ . O descritor obtido possui 20 dimensões ( $2 \times 10$  sem  $M_{00}^a$ ). Este tipo de descritor foi originalmente calculado para imagens a cores.

A utilização de medidas como os momentos não fornecem informação sobre o posicionamento relativo dos pixels uns em relação aos outros. Este tipo de informação pode ser importante quando se tem como objectivo descrever uma textura. Uma maneira de o fazer é utilizar uma matriz de co-ocorrências (MCO), cujo método de cálculo se encontra na figura 2.5. Considerando um operador  $Q$  que define a posição relativa entre dois pixels e uma imagem  $f$  com  $L$  valores distintos de intensidade. A MCO  $G$  tem no seu elemento  $g_{ij}$  o número de vezes que um par de pixels com intensidades  $z_i$  e  $z_j$ , ocorre em  $f$ , na posição especificada por  $Q$ . Na figura 2.5,  $L = 8$  e o operador  $Q$  é definido como “um pixel imediatamente à direita”. O método de cálculo desta matriz é descrito com maior detalhe em [GWE04]. Utilizando a matriz de co-ocorrências, surgem dois diferentes descritores: o primeiro, utiliza a matriz total com todos os seus elementos e o segundo utiliza medidas características obtidas dessa matriz que serão descritas de seguida. As medidas mais comuns incluem a correlação, a energia (também conhecida como uniformidade), o contraste e a homogeneidade. Para uma matriz de dimensões  $K \times K$  e considerando  $p_{ij}$  o  $i$ -ésimo termo de  $G$  dividido pela soma de todos os elementos de  $G$  apresentam-se as

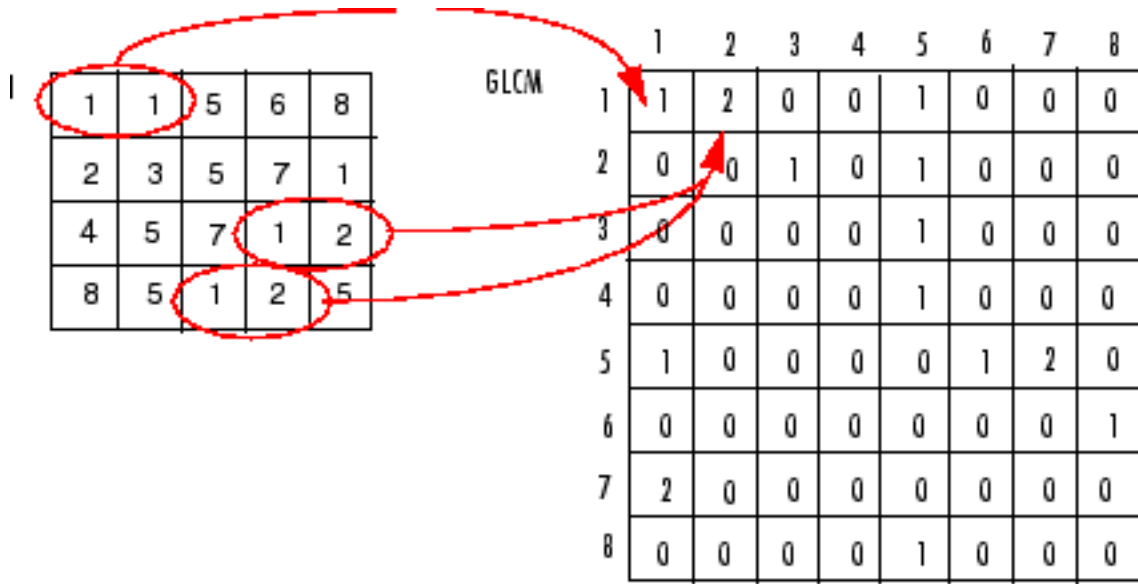


Figura 2.5: Cálculo da matriz de co-ocorrências

fórmulas de cálculo da correlação, na equação 2.19, da energia em 2.20, do contraste em 2.21 e, na equação 2.22, da homogeneidade.

$$\text{Correlação} = \sum_{i=1}^K \sum_{j=1}^K \frac{(i - m_r)(j - m_c)p_{ij}}{\sigma_r \sigma_c} \quad (2.19)$$

$$m_r = \sum_{i=1}^K i \sum_{j=1}^K p_{ij}$$

$$m_c = \sum_{j=1}^K j \sum_{i=1}^K p_{ij}$$

$$\sigma_r^2 = \sum_{i=1}^K (i - m_r)^2 \sum_{j=1}^K p_{ij}$$

$$\sigma_c^2 = \sum_{j=1}^K (j - m_c)^2 \sum_{i=1}^K p_{ij}$$

$$\text{Energia} = - \sum_{i=1}^K \sum_{j=1}^K p_{ij} \log_2 p_{ij} \quad (2.20)$$

$$\text{Contraste} = \sum_{i=1}^K \sum_{j=1}^K (i - j)^2 p_{ij} \quad (2.21)$$

$$\text{Homogeneidade} = \sum_{i=1}^K \sum_{j=1}^K \frac{p_{ij}}{1 + |i - j|} \quad (2.22)$$

No conjunta das equações 2.19 a 2.22,  $m_r$  representa a média calculada ao longo das

linhas,  $m_c$  a média calculada ao longo das colunas,  $\sigma_r^2$  e  $\sigma_c^2$  representam o desvio-padrão ao longo das linhas e das colunas, respectivamente. Para uma explicação mais detalhada do significado de cada uma das medidas refere-se o livro de Gonzalez *et al.* [GWE04]. Outro descritor é apresentado por Shechtman e Irani e denomina-se auto-similaridade (*self-similarity* - SSIM) [SI07]. Surge com o objectivo de medir a similaridade entre entidades visuais, imagens ou vídeos, baseado em emparelhar similaridades internas entre elas. Na figura 2.6 está ilustrado o processo para gerar o descritor SSIM  $d_q$  associado a um pixel  $q$ . O bloco da imagem em volta, tipicamente de  $5 \times 5$  é comparado com

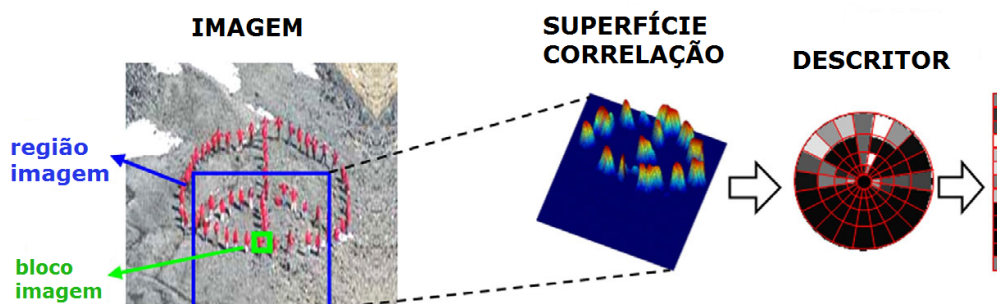


Figura 2.6: Processo para obter descritor SSIM

uma imagem centrada em  $q$  (tipicamente de raio 40), usando uma simples diferença da soma dos quadrados (*SSD*) entre si. A distância de superfície resultante  $SSD_q(x, y)$  é normalizada e transformada numa “superfície de correlação”  $S_q(x, y)$  segundo a equação 2.23, em que  $var_{noise}$  corresponde à variação fotométrica e  $var_{auto}(q)$  tem em conta o contraste do bloco e a sua estrutura em padrão.

$$S_q(x, y) = \exp\left(-\frac{SSD_q(x, y)}{\max(var_{noise}, var_{auto}(q))}\right) \quad (2.23)$$

No passo seguinte  $S_q(x, y)$  é transformado em coordenadas logarítmicas-polares centradas em  $q$  e divididas em 80 direcções (20 ângulos, 4 intervalos radiais). O valor máximo de correlação é escolhido em cada uma das direcções. Esses 80 valores máximos formam as 80 dimensões do descritor SSIM associado ao pixel  $q$ . No final o vector do descritor é normalizado para um intervalo  $[0..1]$  para ser invariante a diferenças de padrão e distribuição de cor dos diferentes blocos e as regiões circundantes. Na figura 2.7 podemos ver alguns exemplos da aplicação deste descritor.

## 2.2 Classificadores

A aprendizagem automática (*machine learning*) é um ramo da inteligência artificial dedicado à concepção e estudo de métodos que permitem a sistemas computacionais me-

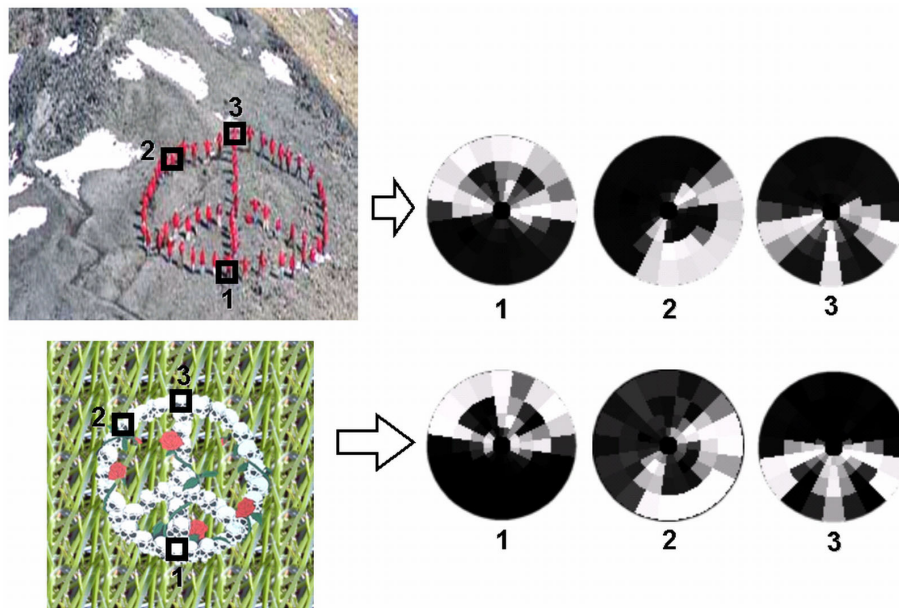


Figura 2.7: Exemplos de aplicação do descritor SSIM

lhorar o seu desempenho com base na experiência [Mit97]. A aprendizagem automática é normalmente dividida em três diferentes categorias, supervisionada, não supervisionada e de reforço, de acordo com o tipo de *feedback* obtido no processo de aprendizagem. A aprendizagem supervisionada (*supervised learning*) caracteriza-se pelo facto de o conjunto de dados fornecido conter uma classificação associada a cada um dos exemplos. O algoritmo aprende uma função de classificação baseada nas entradas e saídas dos dados. O principal objectivo é que, após a fase de aprendizagem, o sistema consiga prever a classe dos novos exemplos fornecidos.

Os métodos não supervisionados (*unsupervised learning*) envolvem a aprendizagem em que os exemplos fornecidos não possuem uma classificação, sendo o seu objectivo o de encontrar padrões nos dados que permitam a sua separação em categorias. Uma das principais técnicas não supervisionadas é o *clustering*.

Os métodos de aprendizagem por reforço (*reinforcement learning*) baseiam-se no problema de um agente que necessita de aprender um comportamento utilizando técnicas de tentativa e erro num mundo dinâmico. Existem duas principais estratégias para resolver problemas de reforço de aprendizagem. A primeira consiste em procurar no espaço dos comportamentos, com o objectivo de encontrar um que tenha um bom desempenho no ambiente. A segunda consiste na utilização de modelos estatísticos e de programação dinâmica para estimar a utilidade das acções nos diferentes estados do mundo.

Neste problema serão explorados métodos de aprendizagem supervisionada, em que os dados de treino são imagens radiológicas planares anotadas por peritos, que irão permitir o treino de um classificador que indica se estamos ou não na presença de um pedículo. Os

classificadores explorados serão binários, dado que o problema consiste em identificar a presença ou não de um pedículo. A área dos classificadores é muito vasta e novos exemplos surgem com muita frequência. Alguns deles não são adequados ao problema em estudo, como por exemplo, aqueles baseados em regras e as árvores de decisão, pelo facto de os dados utilizados no trabalho dos pedículos serem numéricos o que não se adequa a este tipo de classificadores. Para este trabalho vão ser explorados três classificadores: as redes neuronais artificiais, as máquinas de vectores de suporte e o Naive Bayes. Estes foram seleccionados com base no tipo de dados inteiros que serão utilizados, no tipo de output desejado e por serem classificadores muito utilizados e que demonstraram ser capazes de obter bons resultados em muitos problemas. Um exemplo é o trabalho de Duong *et al.* [DCL09] que utilizaram máquinas de vectores de suporte na detecção da coluna vertebral.

### 2.2.1 Redes Neuronais Artificiais

As redes neuronais artificiais são provavelmente uma das aproximações mais comuns para a classificação. Estas são inspirados na neurofisiologia e pretendem simular a capacidade de aprendizagem do cérebro humano. O elemento básico da rede neuronal artificial é o neurónio artificial, cujo esquema podemos ver na figura 2.8. O neurónio é constituído

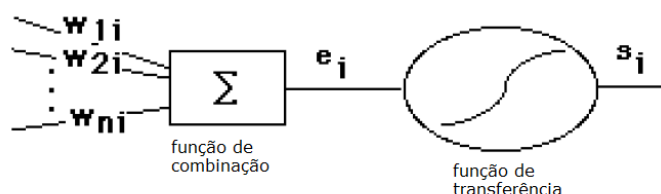


Figura 2.8: Neurónio - elemento básica de uma rede neuronal artificial

por cinco partes fundamentais: a entrada, pela qual se obtém a informação, os pesos das conexões com os neurónios, que determinam a influência dos valores de entrada no estado deste neurónio, a função de combinação, que regra geral é uma soma ponderada das entradas, a função de transferência, cujo caso mais comum é a função sigmóide, que permite calcular o estado do neurónio e a saída, que é o resultado da função de transferência.

As redes podem variar muito na sua configuração, tendo diversas estruturas de ligação representadas por parâmetros como o tipo de rede, o tipo de fluxo de informação, o tipo de conexão e a definição do número de nós e camadas. Relativamente ao tipo, existem redes totalmente conectadas, redes de camada única (camada de entrada e de saída) e redes de múltiplas camadas em que existem camadas intermédias. Na figura 2.9 podemos ver a distinção entre estes três tipos de redes. Quanto ao fluxo de informação existem redes directas e redes recorrentes, dependendo se existe apenas uma direcção no fluxo de

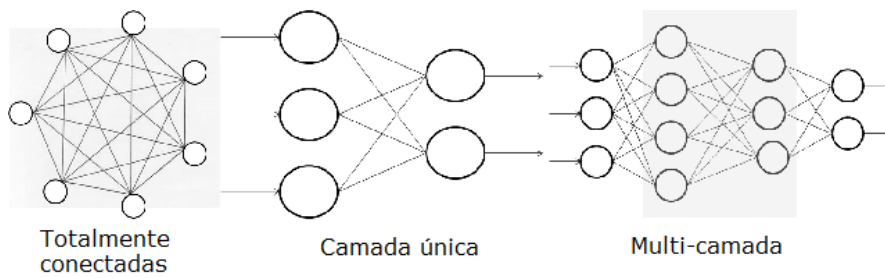


Figura 2.9: Tipos de rede neuronal artificial

informação, redes directas, ou se existe uma realimentação da rede, redes recorrentes. O tipo de conexão nas redes pode ser directa, inter-directa e intra-directa, cuja distinção é feita na figura 2.10. A rede neuronal artificial tem um método de aprendizagem baseado

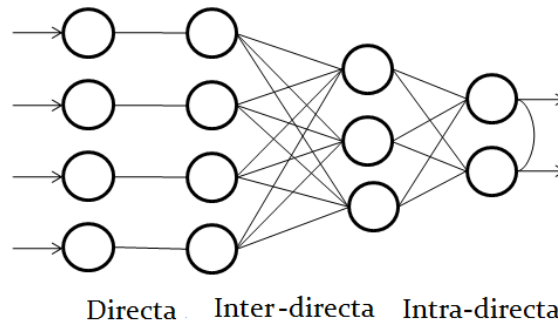


Figura 2.10: Tipos de ligação de uma rede neuronal artificial

na adaptação do peso das ligações entre os seus nós, denominados neurónios e pode ser supervisionado, não-supervisionado ou de reforço consoante o modo como é feita esta adaptação. A abordagem não-supervisionada deve ser utilizada quando não se conhece, *a priori*, as categorias em que se divide a informação. Existem dois algoritmos principais, o competitivo e o de Kohonen [Koh06].

Na abordagem de reforço, a única informação fornecida à rede, é se uma determinada saída é, ou não, correcta. Nesta abordagem existem dois algoritmos importantes, o de Hebb e o de Hopfield [JMM96].

Na aprendizagem supervisionada são dadas à rede entradas e as respectivas saídas classificadas. A rede adapta-se modificando os pesos das conexões, com base na diferença entre a saída esperada e a saída obtida. Na abordagem da aprendizagem supervisionada existem dois algoritmos importantes, o *perceptron* e o de retro-propagação (*backpropagation*). O *perceptron* [Ros58], criado por Rosenblatt, é uma rede neuronal artificial normalmente com uma só camada e capaz de classificar dados linearmente separáveis. É considerado o exemplo mais simples de algoritmo de classificação supervisionado das redes neuronais.

Neste método, os valores de saída apenas admitem dois valores, verdadeiro ou falso. O algoritmo consiste em apresentar exemplos à rede e a ajustar os seus pesos apenas quando a rede não toma a decisão correcta. Inicialmente, todos os pesos são zero e são ajustados consoante a lei 2.24, em que  $c$  é constante,  $s_j$  é a saída da unidade  $j$  e  $d_j$  é a saída correcta da unidade  $j$  fornecida pelo classificador.

$$w_{ij} = c * (s_j - d_j) * s_i \quad (2.24)$$

O método da retro-propagação (*backpropagation*) é um pouco mais complexo, utilizando redes de múltiplas camadas e está dividido em duas fases: a primeira que consiste na propagação positiva do sinal (*feed forward*) e a segunda que consiste na retro-propagação do erro. Vamos explicitar as fases do método, supondo que a rede possui uma camada de entrada com A unidades, uma camada intermédia com B unidades e uma camada de saída com C unidades, denominando os estados das unidades de entrada de  $e_i$  ( $i=1...A$ ), os estados das unidades intermédias de  $c_j$  ( $j=1...B$ ) e os estados das unidades de saída de  $s_k$  ( $k = 1...C$ ) e denominando os pesos das conexões que ligam a camada de entrada e a camada intermédia de  $w_1$  e os que ligam a camada intermédia e a camada de saída de  $w_2$ . No exemplo apresentado  $f(e)$  representa a função de transferência e  $\eta$  a taxa de aprendizagem. Na fase de *feed forward* o sinal de entrada é propagado desde as entradas até à saída ser gerada. O valor de saída é então comparado com o valor correcto fornecido pelo supervisor e é calculado um valor de erro ( $\delta_k$ ) segundo a fórmula 2.25, em que  $d_k$  representa o valor correcto.

$$\delta_k = d_k - s_k, k = 1...C \quad (2.25)$$

Na fase de retro-propagação, o erro é propagado para trás, como se fossem dados de entrada no sentido inverso e, cada valor do erro ponderado com o peso da conexão será usado para o cálculo do novo peso. O erro é retro-propagado para a camada intermédia segundo a fórmula 2.26.

$$\delta c_j = c_j * (1 - c_j) * \left( \sum_k \delta s_k * w_{2jk} \right), j = 1...B, k = 1...C \quad (2.26)$$

Os pesos das conexões entre a camada intermédia e a camada de saída são modificados segundo a lei 2.27.

$$w_{2jk}(n+1) = w_{2jk}(n) - \eta * \delta s_k * \frac{\partial f(e)}{\partial e} * c_j(n), j = 1...B, k = 1...C \quad (2.27)$$

Os pesos das conexões entre a camada de entrada e a intermédia são modificados

segundo a lei 2.28.

$$w_{1ij}(n+1) = w_{1ij}(n) - \eta * \delta_{c_j} * \frac{\partial f(e)}{\partial e} * e_i(n), i = 1 \dots A, j = 1 \dots B \quad (2.28)$$

O algoritmo termina quando forem utilizados todos os exemplos de teste. Este algoritmo tem diversas dificuldades, das quais se destacam a escolha da constante de aprendizagem cujo valor quanto mais elevado for, maiores são as modificações efectuadas no valor dos pesos e mais rápida é a aprendizagem. Contudo, se esse valor for demasiado elevado, podem ocorrer oscilações nos pontos próximos de curvatura pronunciada e o valor de aprendizagem deve manter-se baixo levando a uma aprendizagem mais lenta. Para evitar que o valor dos pesos caia em mínimos locais, pode-se adicionar o factor de *momentum* que faz com que o valor da última variação do peso seja um factor a ter em conta, evitando oscilações em torno de um valor. Outros factores a ter em conta são o tamanho da rede, o número de ligações e a arquitectura da rede. Se o tamanho da rede for demasiado elevado, ela perde a sua capacidade de generalização efectuando associações directas entre a saída e a entrada. A capacidade de generalização da rede é também afectada pelo número de ligações entre as camadas. Quanto mais baixo for este número, mais se força a rede a generalizar. O problema de seleccionar a configuração para a rede pode ser visto como um sistema de equações, em que o número de ligações independentes representa o número de variáveis e, o número de saídas multiplicado pelo número de exemplos de treino representa o número de equações. A configuração da rede deve respeitar a equação 2.29.

$$n^{\circ} \text{ ligações independentes} < n^{\circ} \text{ saídas} * n^{\circ} \text{ exemplos treino} \quad (2.29)$$

No problema estudado vai ser abordada a utilização de modelos supervisionados, utilizando imagens anotadas por peritos como dados de treino.

## 2.2.2 Máquinas de Vectores de Suporte

As máquinas de vectores de suporte (SVM - *support vector machines*) constituem uma técnica de aprendizagem supervisionada que tem vindo a receber uma crescente atenção da comunidade da área da aprendizagem de máquina (*machine learning*) [Mit97]. Os seus resultados de aplicação são comparáveis e, muitas vezes superiores, aos obtidos por outros métodos de aprendizagem, como as redes neuronais artificiais já anteriormente referidas. São aplicadas em diferentes domínios como a recuperação de informação [DSG02], a bioinformática [Nob04] e o processamento de imagem [KJPK02]. As SVMs têm como base o princípio da minimização do risco estrutural (*structural risk minimization* - SRM) desenvolvido por Vladimir Vapnik [Vap00]. A sua ideia básica é encontrar o hiperplano que maximiza a margem entre as classes a classificar. Por vezes, pode ser necessário transformar o conjunto de dados de treino num espaço de maior dimensão

para permitir a separação entre as classes. Existem duas categorias de SVM, as lineares e as não-lineares. A primeira lida com problemas linearmente separáveis e a segunda categoria é uma extensão à primeira que permite a divisão de dados não-linearmente separáveis.

Nas SVM lineares, considerando  $T$  um conjunto treino com  $l$  exemplos  $x_i, y_i, i = 1, \dots, l$ , em que cada exemplo tem  $d$  entradas ( $x_i \in R^d$ ) e os seus respectivos *labels*  $y_i$  que tomam um de dois valores  $y_i \in \{-1, 1\}$ . Todos os hiperplanos em  $R^d$  são parametrizados por um vector  $w$  e uma constante  $b$ , como demonstra a equação 2.30.

$$w \cdot x + b = 0 \quad (2.30)$$

Na equação 2.30,  $w \cdot x$  representa o produto escalar entre os vectores  $w$  e  $x$ .  $w$  é o vector normal ao hiperplano descrito e  $\frac{b}{\|w\|}$  representa a distância do hiperplano em relação à origem. O hiperplano  $(w, b)$  separa os dados em duas regiões e uma função sinal, equação 2.31, pode ser utilizada para a classificação dos dados.

$$f(x) = \text{sign}(w \cdot x + b) \quad (2.31)$$

A partir do hiperplano dado, é possível obter um número infinito de hiperplanos equivalentes pela multiplicação de  $w$  e  $b$  por uma constante  $\lambda \in R^+$ . É então definido um hiperplano canónico, em que  $w$  e  $b$  são escalados de forma a que os exemplos mais próximos ao hiperplano verifiquem a equação 2.32.

$$|w \cdot x + b| = 1 \quad (2.32)$$

São então considerados os hiperplanos que verificam a equação 2.33.

$$y_i(w \cdot x_i + b) \geq 1, \quad \forall i \quad (2.33)$$

Seja  $x_1$  um ponto do hiperplano  $H_1 : w \cdot x + b = +1$  e  $x_2$  um ponto do hiperplano  $H_2 : w \cdot x + b = -1$ , conforme ilustrado na figura 2.11. Projectando  $x_1 - x_2$  na direcção de  $w$ , perpendicular ao plano separador  $w \cdot x + b = 0$ , é possível obter a distância entre os hiperplanos  $H_1$  e  $H_2$ . Essa projecção é apresentada na equação 2.34.

$$(x_1 - x_2) \left( \frac{w}{\|w\|} \cdot \frac{(x_1 - x_2)}{\|x_1 - x_2\|} \right) \quad (2.34)$$

Sabendo que  $w \cdot x_1 + b = +1$  e  $w \cdot x_2 + b = -1$ , a diferença entre estas equações é  $w \cdot (x_1 - x_2) = 2$  [HDO<sup>+</sup>98]. Substituindo este resultado na equação 2.34 obtemos a equação

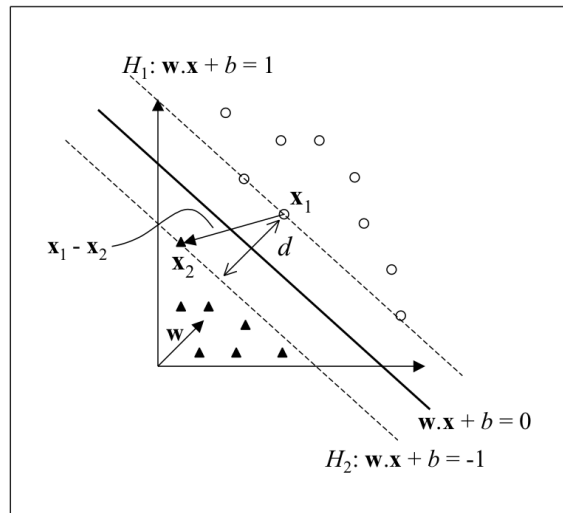


Figura 2.11: Cálculo da distância  $d$  entre os hiperplanos  $H_1$  e  $H_2$ , no contexto de uma SVM [LdC08].

2.35.

$$\frac{2(x_1 - x_2)}{\|w\| \|x_1 - x_2\|} \quad (2.35)$$

Para obter o comprimento do vector projectado, utiliza-se a norma da equação 2.35, obtendo a equação 2.36.

$$\frac{2}{\|w\|} \quad (2.36)$$

Essa é a distância  $d$ , apresentada na figura 2.11. Como  $w$  e  $b$  foram escalados de forma a não haver exemplos entre  $H_1$  e  $H_2$ ,  $\frac{1}{\|w\|}$  é a distância mínima entre o hiperplano separador e os dados de treino. Essa distância é definida como a margem geométrica do classificador linear. Partindo das conclusões anteriores, verifica-se que a maximização da margem de separação dos dados em relação ao hiperplano pode ser obtida pela minimização de  $\|w\|$ . Para isso recorre-se ao problema de optimização [SS02] descrito na equação 2.37.

$$\left\{ \begin{array}{l} \text{Minimizar}_{w,b} \quad \frac{1}{2} \|w\|^2 \\ \text{Com as restrições} \quad y_i(w \cdot x_i + b) - 1 \geq 0, \quad \forall i = 1, \dots, n \end{array} \right. \quad (2.37)$$

O problema de optimização obtido é quadrático e a sua solução tem uma ampla e estabelecida teoria matemática. Este tipo de problemas podem ser resolvidos com a introdução de uma função Lagrangiana que engloba as restrições à função objectivo, associadas a parâmetros denominados multiplicadores de Lagrange. O problema é transformado no

da equação 2.38.

$$\text{Maximizar}_\alpha \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j y_i y_j (x_i \cdot x_j) \quad (2.38)$$

$$\text{Com as restrições } 0 \leq \alpha_i \leq C, \forall i, \dots, n \quad \sum_{i=1}^n \alpha_i y_i = 0 \quad (2.39)$$

Na equação 2.39,  $\alpha$  é o vector do  $n$  não-negativo multiplicador de Lagrange a ser determinado e  $C$  é uma constante. Partindo destas equações verifica-se que o hiperplano óptimo pode ser escrito pela equação 2.40.

$$w = \sum_i \alpha_i y_i x_i \quad (2.40)$$

Da equação 2.40 verifica-se que o vector  $w$  é uma combinação linear dos exemplos de treino. Também pode ser demonstrada a equação 2.41.

$$\alpha_i (y_i (w \cdot x_i + b) - 1) = 0, \quad \forall_i \quad (2.41)$$

A equação 2.41 significa que se a distância de um exemplo for estritamente maior que 1, então  $\alpha_i = 0$ . Logo apenas os pontos mais próximos contribuem para  $w$ . Os exemplos de treino para os quais  $\alpha_i > 0$  são denominados vectores de suporte. Estes vectores são os únicos necessários para a definição do hiperplano óptimo. Ainda mais interessante é que  $\alpha_i$  pode ser visto como a “classificação de dificuldade” de um exemplo  $x_i$ , significando a importância deste na determinação do hiperplano. Assumindo que se obteve o  $\alpha$  óptimo, a partir do qual se constrói  $w$ , ainda é necessária a determinação de  $b$  para a especificação completa do hiperplano. Para o fazer, utiliza-se um exemplo de vector de suporte “positivo” e um “negativo”,  $x^+$  e  $x^-$ , a partir dos quais obtemos as equações 2.42 e 2.43.

$$(w \cdot x^+ + b) = +1 \quad (2.42)$$

$$(w \cdot x^- + b) = -1 \quad (2.43)$$

Resolvendo as equações 2.42 e 2.43 obtêm-se a 2.44.

$$b = -\frac{1}{2}(w \cdot x^+ + w \cdot x^-) \quad (2.44)$$

A necessidade da condição que surgiu na equação 2.39 é descrita na equação 2.45.

$$\alpha_i \leq C \quad (\forall_i) \quad (2.45)$$

Nesta restrição quando  $C = \infty$ , o hiperplano óptimo é aquele que separa completamente

os dados, caso exista. Se  $C$  for finito, altera o problema para descobrir o hiperplano de margens amaciadas [Bor09], que permite que alguns dados sejam mal classificados. As SVM lineares são eficazes na classificação de dados linearmente separáveis ou que possuam uma distribuição aproximadamente linear, sendo que a versão de margens amaciadas tolera a presença de algum ruído. No entanto, existem casos em que não é possível dividir satisfatoriamente os dados de treino com um hiperplano. Na figura 2.12 é ilustrado um exemplo de um destes casos. As SVM lidam com o problemas de dados não

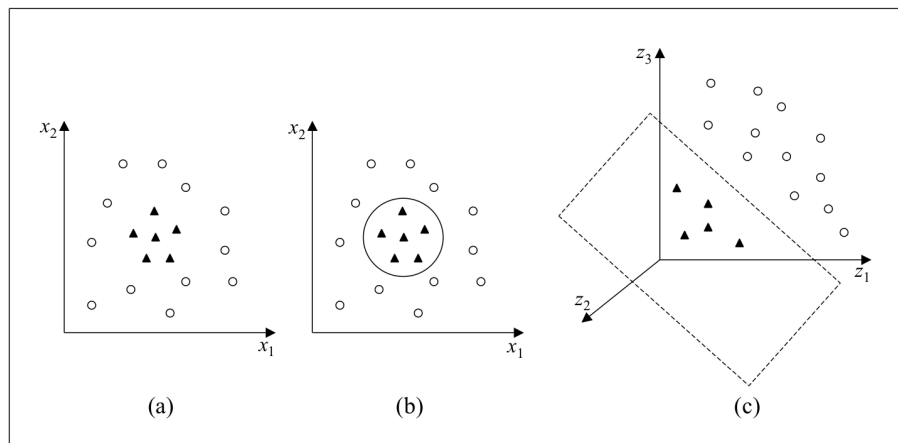


Figura 2.12: Exemplo de transformação de conjunto de dados não-linear para um novo espaço de características. (a) Conjunto de dados não linear; (b) Fronteira não linear no espaço de entradas; (c) Fronteira linear no espaço de características [LdC08].

lineares mapeando o conjunto de dados do seu espaço original para um novo espaço de maior dimensão, denominado espaço de características (*feature space*) [HDO<sup>+</sup>98]. Seja  $\Phi : X \rightarrow \Gamma$  um mapeamento, em que  $X$  é o espaço das entradas e  $\Gamma$  denota o espaço das características. A escolha apropriada de  $\Phi$  faz com que o conjunto de treino mapeado em  $\Gamma$  possa ser separado por uma SVM linear. O uso desse procedimento é motivado pelo teorema de Cover [Hay94]. Dado um conjunto de dados não linear no espaço de entradas  $X$ , o teorema afirma que  $X$  pode ser transformado num espaço de características  $\Gamma$  no qual, com elevada probabilidade, os dados são linearmente separáveis. Para isso, devem ser satisfeitas duas condições: a transformação ser não linear e a dimensão do espaço de características ser suficientemente alta. O mapeamento dos dados para um espaço de maior dimensão é então efectuado recorrendo a  $\Phi$  e é aplicada uma SVM linear sobre este espaço. Esta encontra o hiperplano com a maior margem de separação, garantindo assim uma boa generalização. É utilizada a SVM de margens amaciadas para permitir lidar com algum ruído. Para realizar o mapeamento, aplica-se  $\Phi$  aos exemplos presentes na equação

2.39, obtendo a equação 2.46.

$$\text{Maximizar}_\alpha \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j y_i y_j (\Phi(x_i) \cdot \Phi(x_j)) \quad (2.46)$$

Na equação 2.46 verificam-se as mesmas restrições. Daqui resulta que o hiperplano seria o descrito na equação 2.47.

$$w = \sum_i \alpha_i y_i \Phi(x_i) \quad (2.47)$$

E que a função sinal utilizada para a classificação dos dados seria a da equação 2.48.

$$\begin{aligned} f(x) &= \text{sign}(w \cdot \Phi(x) + b) \\ &= \text{sign}\left(\sum_i \alpha_i y_i (\Phi(x_i) \cdot \Phi(x)) + b\right) \end{aligned} \quad (2.48)$$

Destas equações observa-se que a única informação necessária sobre o mapeamento é de como realizar o cálculo dos produtos escalares entre os dados no espaço de características, dado que sempre que  $\Phi(x_i)$  aparece está relacionado com  $\Phi(x_j)$  por meio de um produto escalar. Este é obtido utilizando funções denominadas de *kernels*. Um *kernel*  $K$  é uma função que recebe dois pontos  $x_i$  e  $x_j$  do espaço de entradas e calcula o seu produto escalar no espaço de características [Her02]. Na equação 2.49 temos uma função de *kernel*.

$$K(x_i, x_j) = \Phi(x_i) \cdot \Phi(x_j) \quad (2.49)$$

É comum utilizar a função de *kernel* sem conhecer o mapeamento  $\Phi$  que é obtido implicitamente. A utilidade dos *kernels* está na simplicidade do seu cálculo e na sua capacidade de representar dados abstractos. Alguns dos *kernels* mais utilizados são os Polinomiais, os Gaussianos ou RBF (*Radial-Basis Function*) e os Sigmoidais.

A formulação original das SVMs apenas é capaz de lidar com problemas de classificação binários. Existe uma série de técnicas que podem ser utilizadas na generalização das SVMs para a solução de problemas de várias classes. No caso em estudo, apenas estamos interessados na classificação binária, contudo um exemplo de generalização seria recorrer à decomposição do problema de várias classes em diferentes sub-problemas binários ou reformular o algoritmo de treino das SVMs para versões multi-classe. Esta última opção, em geral origina algoritmos de elevado custo computacional o que leva a que a estratégia de decomposição seja mais frequente [HL02].

### 2.2.3 Naive Bayes

O classificador de Bayes atribui a um exemplo a classe para a qual ele tem a maior probabilidade de pertencer. O Naive Bayes assume que as características do exemplo são independentes para uma dada classe como apresentado na equação 2.50 em que  $X$  é o vector de características, neste trabalho será considerado o descritor, e  $C$  é uma classe.

$$P(X|C) = \prod_{i=1}^n P(X_i|C), X = (X_1, \dots, X_n) \quad (2.50)$$

O classificador consiste numa função que atribui a um exemplo a classe a que este pertence. Considerando que a probabilidade de um exemplo  $X$  ser da classe  $c$  é dado pela equação 2.51.

$$P(C|E) = \frac{P(E|C)P(C)}{P(E)} \quad (2.51)$$

E que a probabilidade de  $E$  ser classificado por uma classe  $c_1$ , e considerando que existem apenas duas classes ( $c_1$  e  $c_2$ ), é dada pela equação 2.52.

$$f_b(E) = \frac{P(C = c_1)}{P(C = c_2)} \geq 1 \quad (2.52)$$

Na equação 2.52  $f_b(E)$  representa um classificador de Bayes. Assumindo que todos os atributos do exemplo são independentes resulta na equação 2.53.

$$P(E|C) = P(X_1, X_2, \dots, X_n|C) = \prod_{i=1}^n P(X_i|C) \quad (2.53)$$

Utilizando a equação 2.53 o classificador é então descrito na equação 2.54.

$$f_{nb}(E) = \frac{P(C = c_1)}{P(C = c_2)} \prod_{i=1}^n \frac{P(X_i|C = c_1)}{P(X_i|C = c_2)} \quad (2.54)$$

Neste caso  $f_{nb}$  representa um classificador Naive Bayes. Este classificador é a forma mais simples de uma rede de Bayes, em que todos os atributos são independentes dado o valor do variável da classe. Isto é chamado de independência condicional. A suposição de independência condicional raramente se verifica no mundo real, como por exemplo, no HOG em que o valor do gradiente numa casa tem uma relação com o valor das casas vizinhas. Uma abordagem simples para superar esta limitação é ampliar este classificador, permitindo a representação das dependências entre os atributos. Apesar da sua limitação o Naive Bayes é capaz de obter bons resultados [DP97] [Ris01] e poderá ser explorado neste trabalho.

## 2.3 Trabalho relacionado

A detecção de pedículos em radiografias usando classificadores de descritores não se encontra descrita na literatura. De facto, o único trabalho relacionado com a detecção de pedículos em radiografia é o de Doré *et al.* [DDCC07] que consiste na segmentação semi-automática destas estruturas. Nesse trabalho Doré *et al.* [DDCC07] realizaram um estudo de viabilidade para desenvolver uma segmentação automática dos pedículos em imagens radiológicas de pacientes com escoliose. O seu método é baseado na utilização de algoritmos de segmentação, o modelo de contorno activo (*Active Contour Model*) de Chan-Vese proposto em [CV01]. O sistema implementado por Doré *et al.* requer que um utilizador identifique um ponto no interior do pedículo para iniciar o processo de segmentação que é formado por duas fases, a primeira que encontra uma aproximação do contorno e uma segunda onde é realizada uma segmentação mais fina. Este método permite obter a totalidade do contorno do pedículo. Este trabalho está relacionado com esta dissertação pelo facto de a estrutura anatómica a detectar ser comum, tendo no entanto um objectivo diferente. Nesta dissertação o objectivo é encontrar os extremos do pedículo, enquanto que Doré *et al.* pretendeu apenas encontrar o contorno, não providenciando informação sobre a localização dos extremos. Outro ponto em comum é a utilização de imagens anotadas por especialistas, sendo no caso de Doré *et al.* a forma de validação dos resultados.

Em termos de utilização de descritores de imagem e de classificadores existem alguns exemplos de aplicação na detecção de padrões complexos e em imagens médicas. Dalal e Triggs [DT05] demonstraram que a utilização de descritores de imagem com base no gradiente, nomeadamente o HOG, permite a detecção de padrões complexos. No seu trabalho foi utilizado a base de dados de peões do MIT [PP00] e foi efectuada a detecção de pessoas nesse conjunto de imagens. Na figura 2.13 podemos ver um exemplo do processo de detecção utilizado. Existem igualmente aplicações em que os descritores de



Figura 2.13: Exemplo de detecção de uma pessoa, utilizando o Histograma de Gradientes Orientados

imagem são utilizados para a detecção de estruturas anatómicas, tal como é o objectivo deste trabalho. A análise de imagem e a detecção automática em aplicações médicas são geralmente baseadas no processamento de imagem (melhoramento da imagem, detecção

de orlas) ou na classificação de padrões (textura, classificação de histogramas ou gradientes). Duncan e Ayache [DA00] sugerem que a combinação destas duas técnicas pode levar a um aumento de precisão e robustez. De seguida, serão apresentados alguns exemplos de trabalhos que utilizam descritores de imagem com aplicações em imagem médica. Duong *et al.* [DCL09] desenvolveram uma nova metodologia para a detecção automática da coluna em radiografias pósterio-anteriores, motivados pela possibilidade de facilitar a avaliação de deformações na coluna. O seu método consiste, numa primeira fase, em extrair uma região de interesse de um conjunto de radiografias da coluna, utilizando a sua variação de forma a duas dimensões. Esta região inclui 17 áreas que delimitam cada nível de vértebra desde o T1 ao L5. Numa segunda fase, é utilizado um filtro para melhorar a visibilidade das estruturas anatómicas. Na seguinte são calculados descritores baseados na textura para cada região e estes são utilizados para o treino em máquinas de suporte vectorial (*Support Vector Machine* - SVM). O corpo de cada uma das vértebras é então inferido para cada um dos diferentes níveis. A curva da coluna é ajustada aos centros das regiões previstas das vértebras e é feita uma comparação com uma identificação manual recorrendo a um teste  $t$  de Student. Na figura 2.14 podemos ver um exemplo do resultado obtido nesse trabalho. Uma validação clínica de 100 radiografias de pacientes com



Figura 2.14: Resultado do trabalho de Duong *et al.*, para a detecção da coluna utilizando um descritor baseado em momentos estatísticos.

escoliose revelou que a curva detectada é estatisticamente similar ( $p < 0.05$ ), em 93% dos casos, à curva identificada manualmente. Os descritores utilizados por Duong *et al.* [DCL09] são descritos por Gonzalez *et al.* [GWE04] e consistem em medidas baseadas em momentos estatísticos aplicados em histogramas locais. Este descritor foi referido na secção 2.1.4.

Um outro exemplo de aplicação de descritores foi apresentado por Chen *et al.* [CTL<sup>+</sup>10].

O projecto consistiu em utilizar um descritor para efectuar emparelhamento entre imagens da retina. A motivação para a sua realização foi a necessidade de fazer corresponder diferentes imagens para conseguir avaliar a evolução de patologias da retina. Na figura 2.15 podemos ver um exemplo do resultado obtido por esse trabalho. Chen *et al.* de-

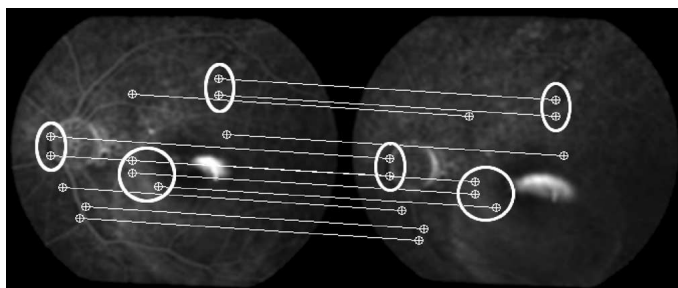


Figura 2.15: Resultado do trabalho de Chen *et al.*, para emparelhamento de imagens da retina utilizando o descritor PIIFD.

envolveram uma *framework* denominada Harris-PIIFD constituída por diversos passos. O primeiro consiste em utilizar um detector de Harris [HS88] para obter os cantos. A fase seguinte consiste em escolher uma orientação para cada um deles. O terceiro passo consiste em extrair o descritor PIIFD e o último em efectuar o emparelhamento entre diferentes descritores. Os passos seguintes são remover as correspondências incorrectas, refinar as suas localizações e seleccionar o modo de transformação. O descritor apresentado neste trabalho foi proposto pelos seus autores e chama-se descritor invariante à intensidade parcial (*partial intensity invariant feature descriptor* - PIIFD) e é semelhante ao descrito apresentado no SIFT [Low04]. Ambos os descritores são parcialmente invariantes a transformações afim e o seu método de extracção é muito semelhante. O PIIFD, para ser parcialmente invariante à intensidade, é extraído recorrendo aos contornos (linhas que marcam os limites de um objecto ou de uma figura) da imagem. No estudo de Chen, a extracção do contorno é simplificada pela aplicação de duas operações aos gradientes da imagem. A primeira consistir em normalizar a magnitude do gradiente. A segunda é baseada na ideia de que as orientações do gradiente, em localizações correspondentes, apontam na mesma direcção ou em direcções opostas. Para concluir, o PIIFD consiste na combinação linear de dois sub-descritores, um constituído por uma matriz de histogramas de orientação e o outro por uma versão rodada da matriz anterior.

Outro exemplo de aplicação de descritores foi apresentado por Marín *et al.* [MAGB10]. Apresentam um novo método supervisionado para a segmentação de vasos sanguíneos em imagens da retina. O método utiliza uma rede neuronal para a classificação dos pixels e um vector de 7 dimensões composto pelos níveis de cinzento e por características baseadas em momentos invariantes. Numa primeira fase do método é feito um pré-processamento para uma homogeneização dos níveis de cinzento e um realce dos vasos

sanguíneos. Na segunda é realizada a extracção do descritor. Na fase seguinte é utilizado um classificador para etiquetar um pixel como sendo, ou não, um vaso sanguíneo. No final do método são removidos pontos isolados que representam falsos positivos e as falhas nos vasos são preenchidas. A fase do cálculo do descritor é sub-dividida em duas. Na primeira são obtidas características baseadas no nível de cinzento, utilizando a diferença entre o nível de cinzento do pixel a analisar e um valor estatístico que representa o valor da sua vizinhança que gera cinco valores para o descritor. A segunda utiliza um momento invariante para descrever pequenas regiões formadas por uma janela centrada no pixel actual e fornece os dois restantes valores para o descritor.

Ao longo desta secção foram apresentados alguns exemplos de aplicações que recorreram a descritores de imagem e utilizaram imagens médicas. Verificou-se que a utilização de descritores é um método utilizado recorrentemente na literatura para a análise e pesquisa de padrões numa imagem médica.

## 2.4 Sumário

Ao longo deste capítulo foi apresentado o estado da arte para as diferentes áreas cobertas por este trabalho. Incidiu em duas diferentes áreas: descritores de imagem para a detecção de padrões e classificadores. Na primeira, foram apresentadas as diferentes categorias de descritores e dados exemplos de cada uma das categorias. Os descritores de imagem são utilizados neste trabalho para a caracterização de regiões ou pontos da imagem. Na segunda parte, foram apresentados vários tipos de classificadores que podem ser usados neste tipo de trabalhos. No início foram referidas diferentes classes de classificador e, no final, foram descritos com detalhe os três explorados nesta dissertação. Ainda neste capítulo foi apresentado trabalho relacionado, nomeadamente tendo em conta a estrutura anatómica ou tendo em conta a aplicação de descritores, especialmente em imagem médica. Nesta parte, destaca-se o trabalho de Duong *et al.* onde é utilizado um classificador binário e descritores de textura detectar a curva da coluna vertebral.

## Capítulo 3

# Detecção Automática de Pedículos

Neste capítulo será descrito com detalhe o método proposto para o problema da detecção automática dos pedículos. O processo aqui descrito não é exclusivo para os pedículos e poderia ser aplicado a outras estruturas anatómicas. Na figura 3.1 é apresentada uma visão geral das diversas etapas que o constituem.

O método é formado por duas partes principais, o treino de um classificador e a classificação de novas imagens dada uma área de interesse. A fase de treino recorre a um conjunto de instâncias anotadas, apresentadas como imagens de treino, sobre as quais é efectuado um pré-processamento. Neste passo são abordados factores como a rotação, a eliminação do ruído e a presença de artefactos que possam ter influência na performance do classificador. O processo de treino prossegue com a divisão da área de pesquisa em blocos, sendo calculado um descritor para cada um dos bloco. O método permite a utilização de diversos descritores, revistos no capítulo 2. Combinando o resultado do descritor com a classificação de cada um dos blocos (i.e. a ponta do pedículo encontra-se ou não dentro do bloco) forma-se a entrada dada ao classificador para o seu treino. O resultado desta primeira fase é um classificador apto a classificar novas instâncias, ou seja, um classificador que permite diferenciar blocos que contêm a estrutura anatómica de interesse de blocos dos que não a incluem, dado um descritor desse bloco.

A classificação de novas imagens parte de uma área de interesse, sobre a qual se utiliza novamente um pré-processamento em tudo semelhante ao da fase de treino. Seguidamente é feito um varrimento, em que para cada pixel, é criado um bloco centrado nele. A fase de teste prossegue com a extracção do descritor para cada um dos blocos. Após esta etapa utiliza-se o classificador para detectar a presença do pedículo. É utilizada a saída do classificador que indica a probabilidade de um dado bloco descrever ou não um pedículo. Idealmente, o bloco com valor de saída mais elevado corresponderá ao pedículo, no entanto esta técnica permite também identificar potenciais candidatos alternativos através

## Detecção Automática de Pedículos

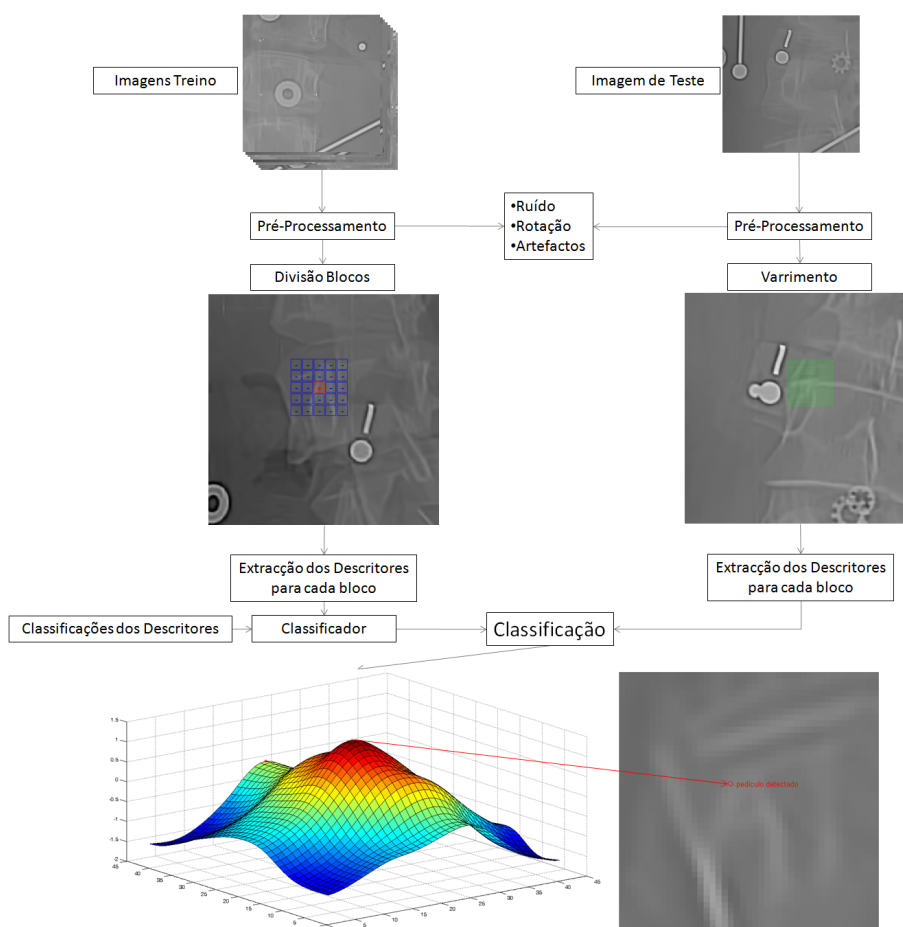


Figura 3.1: Visão geral do método proposto para a detecção dos pedículos. No diagrama vemos como é feita a divisão em blocos e sua classificação. Na outra imagem vemos um exemplo de varrimento. No gráfico mostra-se o resultado do classificador para a área de interesse da imagem de teste e à sua direita vemos o ponto mais provável para a localização do pedículo de acordo com o resultado do classificador.

da análise de máximos locais.

Nas secções seguintes serão detalhadas os principais processos do método.

### 3.1 Pré-Processamento

A fase de pré-processamento foca-se em três diferentes factores: a eliminação do ruído, a invariância à rotação e a eliminação de artefactos. A presença de ruído nas imagens é um factor que pode interferir na capacidade de detecções de padrões, como os pedículos, e nesse sentido foi utilizado um filtro de eliminação de ruído com o objectivo de criar novas imagens para as experiências. Uma opção possível seria utilizar um filtro gaussiano que permite eliminar alguns tipos de ruído, mas pode “borrar” as imagens perdendo-se detalhes. Com isto em perspectiva, foi necessário recorrer a algoritmos mais

avançados de eliminação de ruído (*denoising*), como é o caso do BM3D [DFKE07]. Este filtro foi proposto por Dabov *et al* e utiliza uma estratégia baseada numa representação esparsa melhorada e em filtragem colaborativa. Para detalhes sobre este filtro consultar [DFKE07]. Para este trabalho as imagens estavam disponíveis tanto na sua forma original como pré-processadas com o filtro BM3D. Na figura 3.2 são comparadas duas imagens, sendo uma a original, com a presença de ruído, e uma outra em que foi utilizado o filtro.

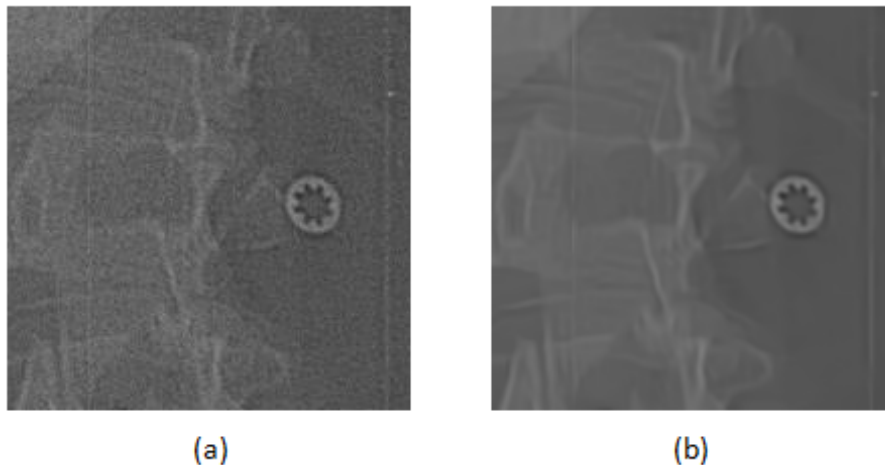


Figura 3.2: Comparação entre imagem original(a) e imagem filtrada(b).

A rotação presente nas vértebras no plano frontal faz com que a orientação dos pedículos varie entre as imagens, tornando-os padrões diferentes consoante a vértebra em análise, o que pode dificultar a capacidade de generalização do método. Assim, este pré-processamento tem em vista tornar o método invariante à rotação aplicando uma rotação 2D à imagem que permita uniformizar a orientação dos pedículos entre imagem, eliminando a rotação presente na vértebra. O ângulo de rotação a aplicar foi calculado a partir de dois pontos identificados manualmente, um na ponta superior e outro na ponta inferior do pedículo. Esta opção foi escolhida apenas para efeitos de avaliação deste tipo de pré-processamento, sendo que existem métodos semi-supervisionados de reconstrução da coluna [MBB<sup>+</sup>11] que permitem calcular de forma expedita esta rotação para todas as vértebras e que poderiam ser utilizados para providenciar esta entrada. Com as coordenadas de referência é possível o cálculo da rotação existente que é depois aplicada para formar novas imagens em que a rotação da vértebra é zero. O ângulo é calculado através da função arco-tangente e será denominado  $\theta$ . A rotação da imagem é realizada recorrendo a um algoritmo de interpolação bilinear [Pra01]. Após aplicada a rotação na imagem é necessário determinar as novas coordenadas dos extremos do pedículo (coordenadas de referência) que é feita por uma transformação semelhante à utilizada na aplicação da rotação. Foram criadas novas imagens com a rotação aplicada ao longo do processo de detecção, quer na versão original das imagens quer na versão na qual foi

aplicado o filtro de eliminação de ruído. Na figura 3.3 podemos distinguir uma imagem original de um outra em que foi aplicada a rotação.

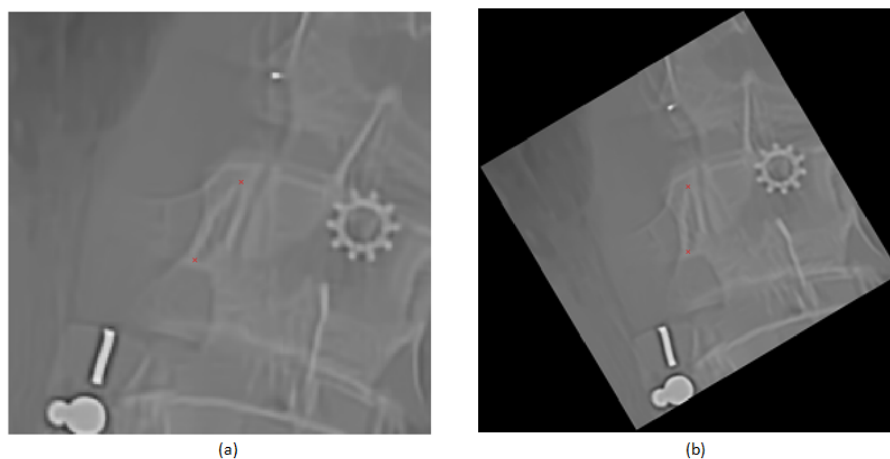


Figura 3.3: Comparação entre imagem original(a) e imagem com rotação aplicada(b).

A presença de artefactos é outro factor que pode limitar a detecção dos padrões. Um artefacto é um objecto radio-opaco (metálico) que pode ter diversas formas e é utilizado para calibrar geometricamente as radiografias. Esta calibração consiste em encontrar as matrizes necessárias para o mapeamento 2D para 3D (projectar pontos 3D para as radiografias; e/ou triangular pontos 2D visíveis nas radiografias frontal e lateral para assim obter as coordenadas 3D). Este tipo de estruturas podem influenciar os resultados quer por aparecerem sobrepostos a pedículos quer por se encontrarem presentes dentro da área de interesse. Esta presença influencia o cálculo de alguns dos descritores devido à sua natureza em que a intensidade dos pixels é um elemento importante no seu processo de extracção. Com o objectivo de limitar a influência dos artefactos algumas imagens do conjunto de dados não foram seleccionadas. O processo de selecção foi feito através de uma controlo manual em que foram detectadas as imagens com artefactos sobrepostos aos pedículos. Após esta remoção, algumas imagens possuem ainda artefactos na área de interesse aos que se espera que o classificador seja imune. Na figura 3.4 é possível ver os diferentes tipos de artefactos que podem surgir nas imagens radiológicas.

O pré-processamento é aplicado em duas diferentes alturas, antes do treino da fase de treino e antes da classificação.

## 3.2 Treino

A fase de treino é subdividida em várias etapas em que são efectuadas múltiplas experiências com diferentes combinações para a obtenção dos melhores parâmetros possíveis.

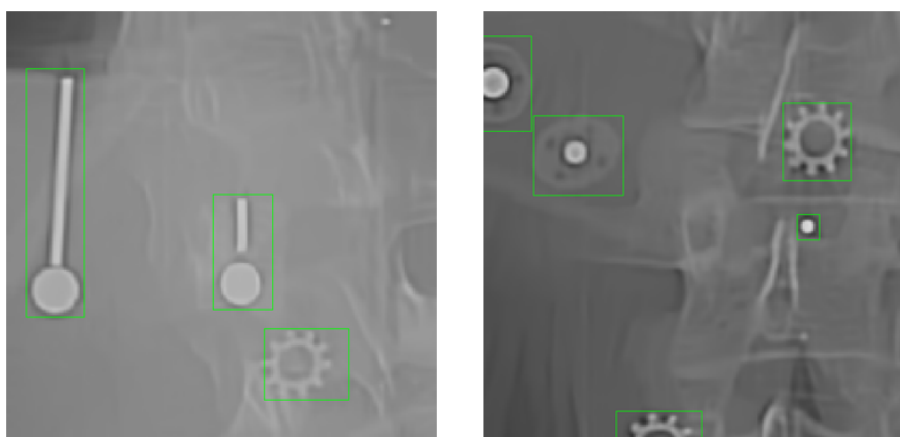


Figura 3.4: Exemplos de artefactos presentes nas imagens radiológicas.

Esta fase inicia-se com uma divisão da área de pesquisa em blocos. Este método é parametrizável com o tamanho do bloco e o grau de sobreposição entre os blocos. Na figura 3.5 apresentam-se exemplos de configurações possíveis.

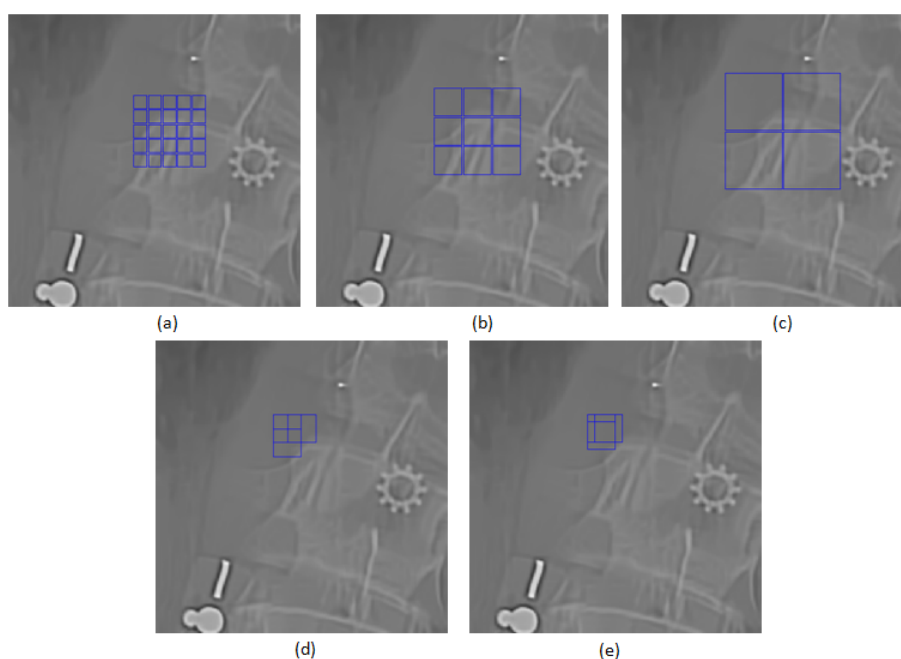


Figura 3.5: Diferentes configurações na divisão em blocos, para o treino do classificador. (a),(b) e (c) diferentes tamanhos no bloco. (d) e (e) diferentes sobreposições

A divisão de blocos é justificada pelo facto de o pedículo ter um padrão que necessita de um área superior a um pixel para ser capturado. No capítulo 5 são comparadas as várias combinações possíveis. O próximo passo da fase de treino é a extracção de um descritor, para cada um dos blocos obtidos anteriormente. Utilizando o *output* do descritor e

a classificação de cada um dos blocos, dado que o processo de treino é supervisionado em que um perito identificou os pedículos, é realizado o treino do classificador. Um factor a considerar no processo de treino é o desequilíbrio existente entre os exemplos positivos e negativos. Este facto decorre de para cada imagem de treino apenas é considerado positivo o bloco centrado no pedículo e os restantes blocos são considerados negativos. Ainda neste processo de treino, dependendo do tipo de classificador, pode ser necessário uma fase de optimização dos parâmetros do classificador. Nesta optimização recorre-se a uma validação cruzada  $k$ -desdobrável (*k-fold cross-validation*) que consiste em dividir de forma aleatória o conjunto de treino em  $k$  sub-conjuntos. Cada um dos sub-conjuntos é utilizado como validação, enquanto os restantes  $k - 1$  sub-conjuntos são utilizados como dados de treino. O processo de validação é realizado  $k$  vezes utilizando cada uma das amostras uma única vez. A maioria das medidas utilizadas neste processo de optimização tem como base uma matriz de confusão (*confusion matrix*) como a apresentada na tabela 3.1. Nesta tabela são definidos conceitos de verdadeiro positivo (*true positive* - TP), verdadeiro negativo (*true negative* - TN), falso positivo (*false positive* - FP) e falso negativo (*false negative* - FN).

Tabela 3.1: Matriz de Confusão

	classificados como positivos	classificados como negativo
reais positivos	verdadeiros positivos (TP)	falsos negativos (FN)
reais negativos	falsos positivos (FP)	verdadeiros negativos (TN)

A medida mais comum é a *accuracy*. Na fórmula 3.1 está apresentado o seu método de cálculo.

$$accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (3.1)$$

Esta métrica apresenta alguns problemas quando utilizada em conjuntos de dados desequilibrados, como é o caso neste problema onde para cada imagem existe apenas 1 exemplo positivo para vários negativos. Dado um classificador básico que classifique todos os exemplos como negativos (se esta for a classe maioritária) terá um valor elevado de *accuracy*, contudo será inútil na detecção de positivos que é o aspecto chave deste trabalho. Existem outras medidas muito utilizadas neste contexto de desequilíbrio. Um exemplo é a medida F (*F-measure*) [TZCK09] que é utilizada em situações em que se está interessado em detectar apenas uma classe com elevada probabilidade. O seu cálculo utiliza outras duas medidas o nível de precisão (*precision*) e de recuperação (*recall*). O modo de obter os seus valores é mostrado em 3.2 e 3.3.

$$precisão = \frac{TP}{TP + FP} \quad (3.2)$$

$$recuperação = \frac{TP}{TP + FN} \quad (3.3)$$

A *F-measure* integra a precisão e a recuperação numa medida única para facilitar a sua utilização [VR79]. Na equação 3.4 apresenta-se a fórmula de *F-measure*.

$$F - measure = \frac{2 * precisão * recuperação}{precisão + recuperação} \quad (3.4)$$

A área sobre a curva ROC (*receiver operating characteristic*) é outra métrica que pode ser utilizada no processo de validação cruzada (*cross-validation*) [Bra97]. A curva de ROC é um gráfico em que no eixo das abcissas temos a taxa de falsos positivos e no eixo das ordenadas a taxa de verdadeiros positivos. É utilizada a área sobre a curva com o objectivo de ter uma única medida para análise. Na figura 3.6 é apresentado um exemplo de uma curva ROC.

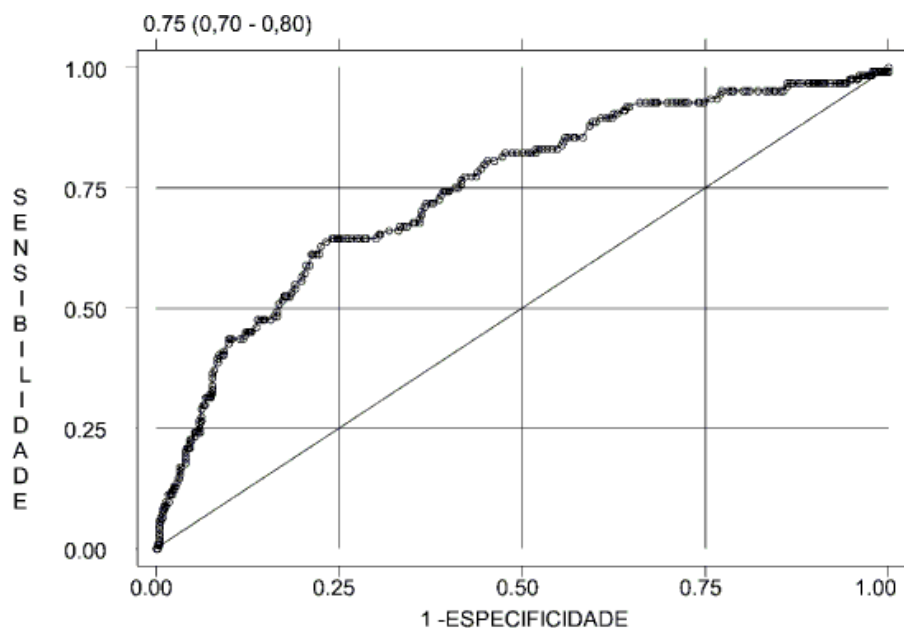


Figura 3.6: Exemplo de curva ROC [PL04].

Uma outra medida muito popular que é utilizada é o *G-mean* proposto em [Vap00]. Esta medida é cada vez mais utilizada na área da aprendizagem máquina [VCC99] [WC03]. Esta métrica tem como objectivo obter um balanceamento óptimo da capacidade

de classificação entre as duas classes separadamente, utilizando duas outras a sensibilidade (*sensitivity*) e a especificidade (*specificity*). A sensibilidade, também denominada taxa de verdadeiros positivos ou *accuracy* da classe positiva e a especificidade, também denominada taxa de verdadeiros negativos ou *accuracy* da classe negativa, foram combinadas numa média geométrica formando o *G-mean* [Vap00]. As equações 3.5 e 3.6 descrevem, respectivamente, a sensibilidade e a especificidade, sendo a *G-mean* descrita na equação 3.7.

$$\text{sensibilidade} = TP / (TP + FN) \quad (3.5)$$

$$\text{especificidade} = TN / (TN + FP) \quad (3.6)$$

$$G - \text{mean} = \sqrt{\text{sensibilidade} * \text{especificidade}} \quad (3.7)$$

Pelo facto de esta medida ser adoptada em diversos trabalhos [WC03] [KM97] [KHM97] e pela sua facilidade de cálculo esta foi a utilizada no método de validação cruzada efectuado no treino do classificador.

Existem outras alternativas na forma de lidar com o problema do desequilíbrio e uma das mais populares é influenciar o classificador de modo a dar uma maior importância aos exemplos positivos [AKJ04]. Este método é denominado ponderação (*weighting*) e pode ser efectuado, por exemplo, com o aumento do peso da penalidade dada a um erro na má classificação de um exemplo positivo. Outras abordagens possíveis incluem técnicas de pré-processamento dos dados, como a amostragem (*sampling*), em que existe a possibilidade de reduzir o número de exemplos da classe maioritária (*undersampling*) ou aumentar o número de exemplos da classe minoritária (*oversampling*) para criar um conjunto de dados balanceado. Nas técnicas de pré-processamento temos diversos trabalhos relacionados, como o de Kubat e Matwin [KHM97] ou o de Japkowicz [Jap00]. Chawla *et al.* criaram um método denominado SMOTE (*Synthetic Minority Oversampling Technique*) que cria novas instâncias através de uma técnica de “transdução de fantasma” (“*phantom transduction*”) [CBHK02]. Outra possibilidade é utilizar diferentes métricas no processo de validação cruzada (*cross-validation*) do classificador em vez da *accuracy* como, por exemplo, o *G-mean* descrito anteriormente. Neste trabalho optou-se pela utilização do primeiro método, de alterar o peso dado a cada classe, e a última utilizando diferentes métricas no processo de treino. O *undersampling* não foi utilizado pelo facto de poder descartar exemplos que poderiam ser importantes para o classificador [AKJ04] e por normalmente não obter bons resultados [BPM04]. O *oversampling* é uma técnica que apresenta bons resultados [CBHK02] [BPM04] e é um elemento que no futuro poderá ser explorado no problema dos pedículos.

### 3.3 Detecção

A fase de detecção, ou de teste, recorre também a um pré-processamento cuja etapas já foram apresentadas previamente neste capítulo. O processo prossegue com um varrimento feito na área de interesse em que, ao contrário da divisão em blocos da fase de treino, são classificados todos os pixels presentes. Neste processo de varrimento para cada um dos pixels analisado é formado um bloco centrado nele. Na figura 3.7 é mostrado um exemplo do varrimento numa imagem.

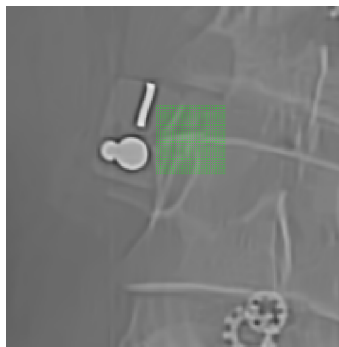


Figura 3.7: Exemplo de varrimento numa imagem de treino.

A detecção prossegue com a extracção do descritor em cada um dos blocos, de forma similar ao já descrito no processo de treino do classificador. A classificação de cada um dos blocos recorre ao classificador previamente treinado e utiliza como entrada para cada pixel o vector obtido através da extracção do descritor. Após o processo de classificação existem diversas formas de avaliar o resultado dado pelo classificador. No caso de os valores retornados serem binários (verdadeiro ou falso), a forma de avaliação é limitada, sendo apenas possível verificar quais os pixels classificados como pedículos, sem informação acerca da probabilidade de cada um e de qual o classificador considera ser o melhor candidato. Assim, propôs-se a utilização de classificadores que permitam retornar a probabilidade de um descritor corresponder ou não a um pedículo (exemplo positivo). Considera-se que a localização do pedículo corresponde ao candidato com o maior valor de saída do classificador (máximo global), sendo no entanto possível explorar outros potenciais candidatos realizando uma análise de máximos locais. É proposto utilizar um filtro gaussiano tendo em vista regularizar a saída do classificador fazendo com que os vizinhos de um pixel contribuam para o resultado desse pixel e dessa forma aumentar a probabilidade de detectar o ponto correcto e eliminar valores discrepantes (*outliers*), assim como falsos máximos locais resultantes de eventual ruído na saída. Na figura 3.8 apresenta-se um gráfico 3D com as coordenadas de cada pixel e o respectivo resultado dado pelo classificador. Comparam-se a versão original e a suavizada.

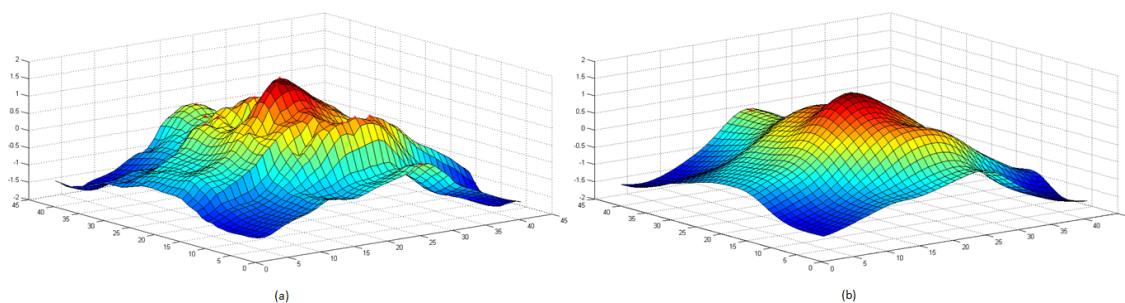


Figura 3.8: Gráfico 3D com o resultado do classificador.(a) Original, (b) Suavizada.

### 3.4 Sumário

Ao longo deste capítulo foi apresentada de forma geral a metodologia proposta para a detecção dos pedículos. Foram apresentados alguns factores que podem ter influência nos resultados, como o ruído presente na imagem, os artefactos e a existência de rotação diferente em cada uma das vértebras. Foi também explicado o método de treino e teste do classificador, detalhando quais as entradas e saídas que serão utilizadas nas diferentes fases.

## Capítulo 4

# Implementação e Avaliação da Detecção

Neste capítulo será detalhado o método de avaliação da detecção dos pedículos e serão descritos os pormenores de implementação de todas as fases apresentadas no capítulo anterior. No início são descritos os materiais utilizados, indicando a origem das imagens radiológicas. Será especificado o filtro de ruído utilizado para as imagens e o método para obter a invariância à rotação nas vértebras. Os descritores e classificadores utilizados nas experiências, com os seus respectivos parâmetros serão também apresentados. O capítulo termina com a descrição das experiências efectuadas, onde são também caracterizadas as métricas de avaliação propostas explicando o seu significado.

### 4.1 Materiais

Neste trabalho utilizou-se um conjunto de 270 imagens de radiografias frontais da coluna vertebral cedidas pelo *Laboratoire d'Imagerie et de Vision 4D (LIV4D)* da *Polytechnique Montréal*. Estas imagens têm um tamanho de 200 por 200 pixels com uma resolução de 0,4mm/pixel. Estão em escala de cinzento, com 8 bits e foram extraídas de radiografias frontais da coluna vertebral adquiridas pelo Saint-Justine Hospital Centre de Montreal, Canada, com um sistema FCR7501 (Fuji Medical, Tóquio, Japão). O conjunto foi dividido em dois diferentes grupos, o primeiro para ser utilizado no treino do classificador e o segundo para ser utilizado como teste. A divisão foi efectuada de forma aleatória partindo do conjunto de imagens inicial. Para cada uma das imagens está disponível um conjunto de dois pontos a duas dimensões, um para cada uma das extremidades do pedículo. Estes pontos são considerados como as coordenadas de referência do pedículo e foram identificados por um perito constituindo a referência para este estudo.

A implementação da totalidade do trabalho foi feita recorrendo a MATLAB, na sua versão R2010b [MAT10].

## 4.2 Descritores

Nesta secção serão apresentados os descritores seleccionados para fazerem parte das experiências. Para cada um dos descritores serão apresentados detalhes da sua implementação, nomeadamente a origem do código fonte nos casos em que o descritor não foi implementado. Os fundamentos teóricos de todos os descritores seleccionados foram apresentados no capítulo 2. Nos descritores em que existam parâmetros estes serão descritos, em conjunto com o modo como podem ser variados. A escolha dos descritores foi realizada tendo em conta a sua popularidade, adequação ao problema e tendo em vista abranger as diferentes categorias de descritores, nomeadamente descritores baseados na distribuição (HOG e Daisy), descritores baseados em medidas estatísticas (momentos invariantes e momentos estatísticos), descritores baseados em respostas *wavelet* (SURF) e descritores que utilizam a informação do posicionamento relativo entre pixels (matriz de co-ocorrências).

### 4.2.1 Momentos Estatísticos

Este descritor é constituído por um conjunto de medidas baseadas em momentos estatísticos aplicados em histogramas locais e foram descritos por Gonzalez *et al.* [GWE04]. Foram utilizados por Duong *et al.* no seu trabalho [DCL09], descrito com maior detalhe na secção 2.3. Este descritor demonstrou bons resultados na segmentação da coluna vertebral e por isso irá ser testado neste trabalho. O descritor é constituído por 6 medidas:

- Intensidade Média ( $m$ )
- Medida de contraste ( $\sigma$ )
- Medida de suavidade ou *smoothness*( $R$ )
- Assimetria ou *skewness* ( $\mu_3$ )
- Uniformidade ( $U$ )
- Entropia ( $e$ )

O método de cálculo destas medidas foi apresentado anteriormente nas equações 2.12 a 2.18 no capítulo 2. Foi efectuada uma implementação própria destas medidas em MATLAB.

#### 4.2.2 Momentos Invariantes

Os momentos consistem num conjunto de medidas baseados no conceito de momentos invariantes 2D que foi introduzido por Hu [Hu62]. Nas experiências é utilizado um descritor com 7 dimensões, em que cada uma é um momento invariante. No capítulo 2 são descritos os conceitos teóricos destes momentos e nas equações 2.4 a 2.10 é apresentado o método do cálculo de cada um dos 7 momentos invariantes utilizados. Este descritor não apresenta parâmetros e a única alteração feita partindo da sua formulação original foi proposta por Gonzalez *et al.* [GWE04]. Esta alteração tem como objectivo reduzir o intervalo de variação dos momentos e, com isso, simplificar a sua interpretação e consiste em escalar os valores de cada momento segundo a equação 2.11. A implementação deste descritor tem por base a apresentada no livro [GWE03].

#### 4.2.3 Matriz de co-ocorrências

Outro tipo de descritores que se podem utilizar são aqueles que utilizam informação sobre o posicionamento relativo dos pixels uns em relação aos outros. Assim, com o objectivo de recorrer a diferentes tipos de descritores vão ser também testados dois descritores que se baseiam na posição relativa dos pixels. O primeiro, matriz de co-ocorrências, consiste na utilização de todos os elementos da matriz de co-ocorrências (MCO), cujo método de cálculo foi brevemente apresentado no capítulo 2 e cujos detalhes estão descritos em [GWE04]. O segundo descritor consiste no cálculo de uma série de métricas características sobre a matriz de co-ocorrências. As medidas utilizadas são a correlação, a energia ou uniformidade, o contraste e a homogeneidade. A sua forma de cálculo foi apresentada previamente.

O cálculo da matriz de co-ocorrências requer a definição de 3 parâmetros. O primeiro, limites de cinzento, foi definido de acordo com os valores presentes nas imagens, sendo utilizados os valores 0 para o mínimo e 255 para o máximo. O número de níveis foi definido como 8 e representa o número de valores utilizado no escalamento dos níveis de cinzento. Este parâmetro origina o uma matriz de  $8 \times 8$ . Outro parâmetro está relacionado com o operador utilizado no preenchimento da matriz e é definido com o pixel imediatamente à direita.

Para a implementação deste 2 descritores foram utilizadas duas funções disponibilizadas pelo MATLAB, a *graycomatrix* e a *graycoprops* que estão presentes na *toolbox* de processamento de imagem (*Image Processing Toolbox*).

#### 4.2.4 Histogramas de Gradientes Orientados - HOG

O histogramas de gradientes orientados (*histograms of oriented gradients* - HOG) é um descritor que é baseado na análise da orientação do gradiente da imagem nas suas

diferentes partes. Este descritor utiliza o cálculo do gradiente para formar o conjunto de valores que são utilizados pelo classificador. Neste trabalho foi feita uma implementação do HOG realizada em colaboração com os elementos do LIACC (Laboratório de Inteligência Artificial e Ciência de Computadores) laboratório onde esta dissertação foi realizada.

Este descritor é parametrizável e possui 4 factores que podem ser optimizados. Na tabela 4.1 são apresentados estes parâmetros. Os parâmetros  $nX$  e  $nY$  podem variar num grande

Tabela 4.1: Parâmetros do Histograma de Gradientes Orientados

Nome	Função
$nX$	Número de blocos na direcção x
$nY$	Número de blocos na direcção y
$nBins$	Número de orientações por histograma
$tNorm$	Tipo de normalização

intervalo a partir de 1 bloco até, na teoria, infinitos blocos. Os valores mais típicos são 3 ou 4, em ambas as direcções com a particularidade de ser comum o  $nX$  e o  $nY$  terem o mesmo valor (divisão em blocos regular). O  $nBins$  é um argumento que também pode variar num grande intervalo e o seu valor típico é 8 que corresponde a um histograma em que os gradientes são colocados em contentores de  $45^\circ$ , nomeadamente:  $0^\circ$ - $45^\circ$ - $90^\circ$ - $135^\circ$ - $180^\circ$ - $225^\circ$ - $270^\circ$ - $315^\circ$ . Estes três argumentos influenciam o número de dimensões ( $N$ ) deste descritor em que  $N = nX * nY * nBins$ . Na figura 4.1 vemos um exemplo de configuração do HOG. O último parâmetro, tipo de normalização, influencia o modo como

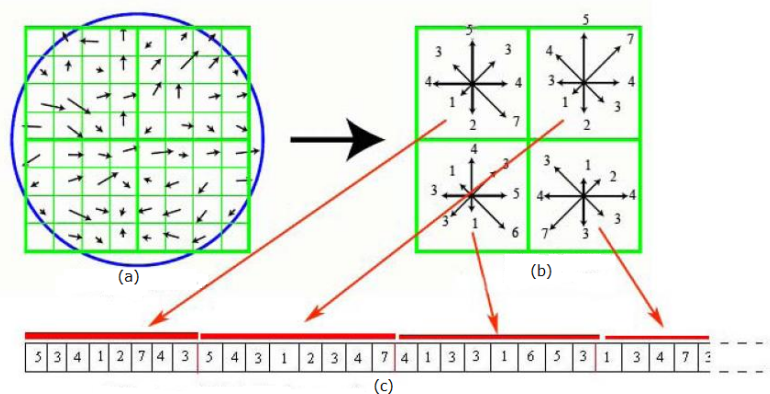


Figura 4.1: Exemplo de configuração do HOG. (a) Janela de  $2 \times 2$ ,  $nX=2$  e  $nY=2$ , (b) com 8 orientações por histograma e (c) as 32 dimensões desta configuração.

os valores do histograma são utilizados. Existem 5 tipos de normalização que podem ser utilizados e estão representados na tabela 4.2. Na tabela 4.2 é referida a utilização de alguns esquemas de normalização dos blocos como o L2-norm e o L1-sqrt que foram explorados por Dalal & Triggs [DT05]. Estes podem ser calculados considerando  $v$  um

Tabela 4.2: Tipos de normalização no Histograma de Gradientes Orientados

Tipo	Descrição
0	Não utilização de normalização
1	Local (por bloco), utilizando L2-norm
2	Global (todos os blocos), utilizando o máximo
3	Global (todos os blocos), utilizando L1-sqrt
4	Global (todos os blocos), utilizando L2-norm

vector do descritor não-normalizado,  $\|v\|_k$  a normalização- $k$  ( $k=1,2$ ) e  $\varepsilon$  uma constante com um valor pequeno. Na equação 4.1 é apresentado o método de obtenção do L2-norm. Para obter L1-sqrt é necessário primeiro recorrer ao cálculo da L1-norm (equação 4.2) e só depois utilizar a equação 4.3 para obter a normalização L1-sqrt. Na equação 4.4 é apresentado o método de cálculo da normalização global utilizando o máximo (tipo 2). A normalização local (tipo 1) tem um método de cálculo semelhante à global e a única diferença é ser aplicada em cada um histogramas de forma independente, ao contrário da global que é aplicada por igual ao conjunto de todos os histogramas.

$$v = \frac{v}{\sqrt{\|v\|_2^2 + \varepsilon^2}} \quad (4.1)$$

$$v = \frac{v}{\|v\|_1 + \varepsilon} \quad (4.2)$$

$$v = \sqrt{\frac{v}{\|v\|_1 + \varepsilon}} \quad (4.3)$$

$$v = \frac{v}{\max(v)} \quad (4.4)$$

#### 4.2.5 *Speeded-Up Robust Features*

Outro descritor seleccionado foi o SURF (*Speeded-Up Robust Features*) [BTVG06]. Tem por base o SIFT, embora seja baseado nas respostas *wavelet* de primeira ordem de Haar em  $x$  e em  $y$  em vez do gradiente. Este é um descritor com 64 dimensões em contraponto com as 128 dimensões do HOG. Neste trabalho foi utilizada uma implementação do SURF para MATLAB da autoria de Dirk-Jan Kroon, obtida em [Kro10] que é baseada numa outra da autoria de Chris Evans, denominada OpenSURF [Eva09].

A nível de parâmetros a implementação utilizada apenas permite a variação do tamanho do descritor, permitindo a utilização de duas versões, a original de 64 e uma estendida com 128. Ambas as configurações foram propostas pelo autor do SURF [BTVG06].

#### 4.2.6 Daisy

O Daisy [TLF10] é um descritor que é eficiente ao ser calculado de forma densa como sucede neste trabalho. Este descritor é inspirado no descritor do SIFT [Low04], o HOG. Este descritor utiliza gradientes no seu cálculo, mas ao contrário do HOG utiliza uma grelha circular. A implementação do Daisy [TLF10] [TLF08] utilizada é disponibilizada por Engin Tola [Tol10] numa versão para MATLAB desenvolvida em 2010. Este descritor possui um grande conjunto de parâmetros que são sumariados na tabela 4.3. O primeiro argumento é o raio do descritor que consiste na distância entre o pixel

Tabela 4.3: Parâmetros do Daisy

Nome	Função	Valor por defeito
$R$	Raio	15
$Q$	Número de convoluções	3
$T$	Número de histogramas por camada	8
$H$	Número de direcções por histograma	8
$SI$	Interpolação Espacial	Falso
$LI$	Interpolação em Camadas	Falso
$NT$	Tipo de Normalização	2

central e o pixel mais distante, sendo medida em pixels. O  $Q$ , número de convoluções ou anéis, o número de histogramas por camada ( $T$ ) e o número de direcções de cada histograma ( $H$ ) apresentam valores numéricos. Os dois argumentos relativos a interpolações,  $SI$  e  $LI$  representam a possibilidade de interpolações no processo de cálculo do descritor. Na figura 4.2 apresentamos um exemplo de alguns dos parâmetros. O último parâmetro diz respeito à normalização utilizada. Existem 4 possibilidades de normalização que estão representados na tabela 4.4. A normalização parcial significa que cada um dos his-

Tabela 4.4: Tipos de normalização no Daisy

Tipo	Descrição
0	Sem Normalização
1	Normalização Parcial
2	Normalização Total
3	Normalização SIFT

togramas do Daisy é normalizado de forma independente, fazendo com que a L2-norm (apresentada no HOG na equação 4.1) de cada um seja igual a 1. A normalização total significa que o descritor é normalizado como um todo e a sua L2-norm é igual a 1. Por fim, a normalização SIFT indica que o descritor é normalizado de forma recursiva de modo a que a sua L2-norm seja 1 e que nenhum histograma individual tem um valor superior a um limiar tal como no SIFT [Low04].

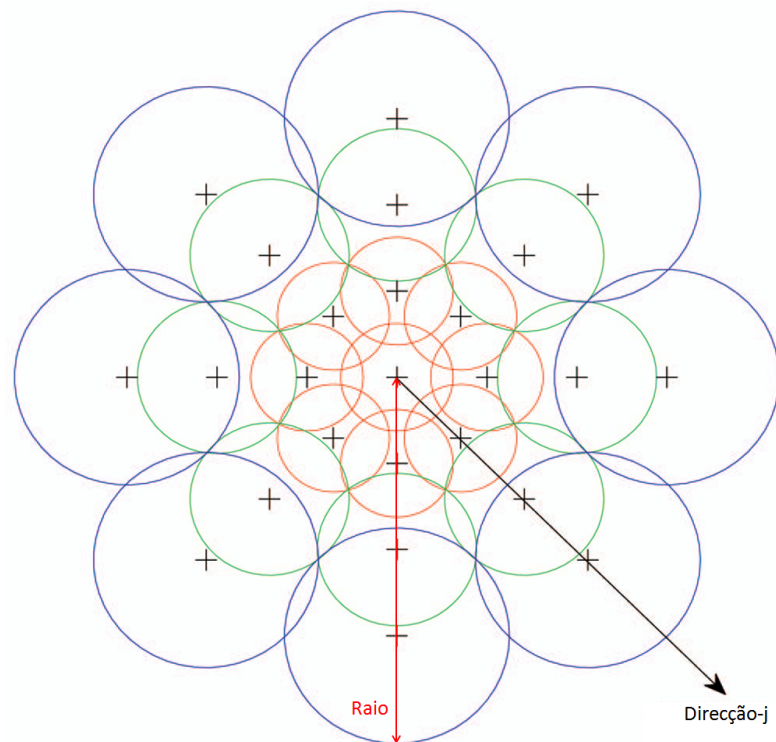


Figura 4.2: Descritor Daisy com a representação do raio, com 3 anéis ( $Q$ ) e com 8 histogramas por camada ( $T$ ).

### 4.3 Classificadores

Nesta secção são apresentados os classificadores que foram seleccionados para fazerem parte das experiências cujos resultados serão apresentados no capítulo 5. Aqui são descritos os classificadores, os seus parâmetros e a origem da sua implementação. A selecção foi feita com base nos resultados anteriores de cada classificador e com o objectivo de ter classificadores de diferentes tipos para verificar qual se adequa melhor a este problema. É importante referir que só classificadores que apresentem uma saída com um valor de probabilidade *à posteriori* associado ao resultado da classificação ou um *score* tendo em vista permitir escolher a localização mais provável do pedículo na área de pesquisa. Não foram considerados classificadores que digam apenas se um exemplo é positivo ou negativo nem foi explorado nenhuma versão multi-classe dado o problema ser binário (presença ou não de pedículo).

#### 4.3.1 Naive Bayes

O Naive Bayes é um classificador de Bayes que assume que as características do exemplo são independentes para uma dada classe. Neste trabalho este classificador será utilizado com o objectivo de verificar se é capaz de obter bons resultados na detecção dos

pedículos. O Naive Bayes apesar de ser um classificador simples de utilizar e de não ter parâmetros para otimizar ainda é capaz de apresentar bons resultados em algumas situações [DP97] [Ris01]. A implementação utilizada é a disponibilizada pelo MATLAB na *Statistics Toolbox*, nomeadamente a função *classify*. O seu processo de treino consiste em utilizar os exemplos supervisionados para criar o classificador e na avaliação dos resultados foi utilizada a probabilidade *a posteriori* da classe positiva de modo a ter um valor numérico para cada um dos exemplos passível de análise.

### 4.3.2 Máquinas de Suporte Vectorial

A máquina de suporte vectorial, SVM (*support vector machine*) foi outro dos classificadores seleccionados. As SVM são um classificador muito popular e com um grande espectro de utilização em diversos domínios de aplicação [DSG02] [Nob04] [KJPK02], como foi descrito no capítulo 2. A implementação das SVM utilizada foi o LIBSVM [CL01b] de Chih-Chung Chang e Chih-Jen Lin, nomeadamente a sua versão para MATLAB. O LIBSVM [CL01a] é uma biblioteca muito popular e possui um grande conjunto de opções que permitem explorar os diversos aspectos das máquinas de suporte vectorial. Nos documentos relativos à implementação da LIBSVM [CL01a] são apresentados com detalhe todos os parâmetros possíveis e aqui apenas serão referidos os mais importantes e que foram os explorados na detecção dos pedículos. O primeiro parâmetro é o tipo de SVM. As diversas possibilidades presentes na biblioteca estão sumariadas na tabela 4.5. Dos diferentes tipos disponíveis apenas foi utilizado o primeiro, C-SVM, que permite a

Tabela 4.5: Tipos de SVM disponíveis na LIBSVM

Tipo	Descrição
0	C-SVM
1	nu-SVM
2	SVM de uma classe
3	epsilon-SVR
4	nu-SVR

utilização de ponderação (*weighting*) que contribui na forma de lidar com o problema do desequilíbrio das classes. Os restantes não permitiam a utilização desta técnica. Outro parâmetro importante nas SVM é o tipo de *kernel* utilizado. Na tabela 4.6 são mostrados os *kernel* implementados pela biblioteca utilizada. Neste trabalho poderiam ser utilizados todos estes *kernel*, mas foi apenas utilizado o radial. Em versões futuras poderiam ser explorados outros *kernel* para verificar se algum é capaz de obter melhores resultados. No *kernel* radial existem dois parâmetros que necessitam de optimização, o  $C$  (custo) e o  $\gamma$ . Para o fazer é utilizada uma pesquisa em grelha com validação cruzada  $k$ -desdobrável (*k-fold cross-validation*) em que  $k = 10$ , e em que a métrica utilizada é o  $G$ -mean. O  $C$  e o  $\gamma$  são números reais e costumam ser utilizados em potências de

Tabela 4.6: Tipos de *kernel* disponíveis na LIBSVM

Tipo	Nome	Função
0	Linear	$u' * v$
1	Polinomial	$gamma * u' * v + coef0$
2	Radial	$exp(-gamma *  u - v ^2)$
3	Sigmóide	$tanh(gamma * u' * v + coef0)$

base 2 [Sta03]. A última opção explorada foi a utilização de ponderação (*weighting*), que permite atribuir maior peso a uma das classes quando o número de exemplos de uma das classes é superior ao da outra.

### 4.3.3 Redes Neurais

Outro classificador seleccionado para o teste foram as redes neuronais artificiais (*artificial neural networks* - RNA). As RNA são um classificador popular com diversas variantes como foi descrito no capítulo 2, sendo também nesse capítulo apresentados os seus fundamentos teóricos. A implementação das RNA utilizada foi a *Neural Network Toolbox*, presente no MATLAB. Foi utilizada a rede *pattertnet* que possui dois parâmetros principais, o número de neurónios na camada escondida e o algoritmo de treino a utilizar. Esta rede tem uma estrutura com 3 camadas, a da entrada em que o número de neurónios é igual ao número de dimensões do descritor, uma camada escondida cujo número de neurónio é parametrizável e uma camada de saída com dois neurónios (presença ou ausência de pedículo). Na figura 4.3 apresentamos um exemplo de rede. O número

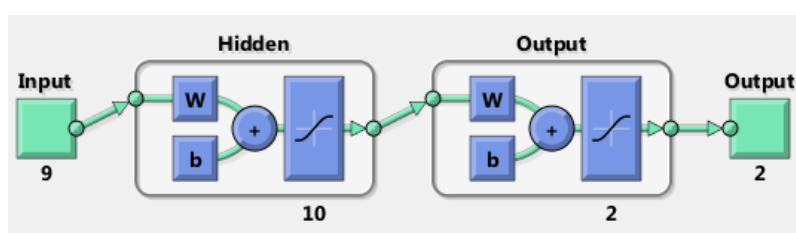


Figura 4.3: Exemplo de rede neuronal utilizada com três camadas.

de neurónios da camada escondida foi optimizado recorrendo a uma validação cruzada *k*-desdobrável (*k-fold cross-validation*) em que  $k = 10$ , de forma semelhante às SVM. A nível do algoritmo de treino esta biblioteca possui um grande número de hipóteses que podem ser consultadas no guia de utilizador [DBH10]. De acordo com esse documento em problemas de detecção de padrões deve ser utilizada uma de duas opções: a retropropagação resiliente (*trainrp*) ou a do gradiente conjugado escalado (*trainscg*) [DBH10].

## 4.4 Experiências

Nesta secção são descritas as experiências efectuadas com o objectivo de verificar qual a combinação entre descritor, classificador e seus respectivos parâmetros que melhor se adequa ao problema da detecção de pedículos abordado. Em cada uma das experiências será descrito o seu objectivo, os valores dos parâmetros que cada elemento utilizado utilizará e as métricas que serão avaliadas. Nestas experiências o conjunto de treino é constituído por 91 imagens (81%) e o de teste por 21 (19%). De referir que destes conjunto já não constam as imagens retiradas, pelo processo explicitado anteriormente, devido a sobreposições de artefactos com os pedículos. As imagens utilizadas são das vértebras lombares L1, L2 e L3 e nas quais foi explorado apenas o extremo inferior do pedículo. Nas radiografias planares utilizadas os pacientes sofriam de escoliose. O conjunto de treino foi usado para treinar o classificador dividindo cada imagem em blocos (como descrito no capítulo 3) e anotando os blocos como positivo quando é o mais próximo da extremidade do pedículo e como negativos nos restantes casos. Para cada imagem do conjunto de teste foi estimada a posição do pedículo utilizando o classificador previamente treinado com o conjunto de treino, sendo que para cada imagem foi calculado um descritor para cada pixel da região de interesse. Para o cálculo das métricas é escolhido o pixel com maior valor (*score*) do classificador. Os resultados obtidos nestas experiências vão ser apresentados no capítulo 5.

Para a avaliação dos resultados obtidos nestas experiências efectuadas foram criadas algumas métricas que serão agora descritas. A razão da sua criação é a dificuldade de avaliar se um determinado ponto é ou não um pedículo através da saída do classificador. Antes da determinação de cada uma destas métricas, existe um critério utilizado na selecção dos candidatos que consiste em remover os pontos cuja distância ao pedículo seja maior que um determinado valor. O número utilizado como limite da distância tem como base o tamanho da área de interesse determinado no trabalho de Moura *et al.* [MBBT09] [MBB<sup>+</sup>11] que utilizaram métodos semi-automáticos na determinação da área de pesquisa. Após este passo é seleccionado o candidato com o valor de saída mais elevado para a determinação das métricas:

- Distância do ponto detectado ao identificado pelo perito
- *Accuracy* da Detecção
- *Accuracy* em função do raio do erro do operador

A primeira calculada, a distância do ponto detectado ao identificado pelo perito, consiste no cálculo da distância euclidiana entre o candidato e o pedículo utilizando as coordenadas 2D na imagem. Na equação 4.5 mostra-se a forma de cálculo desta métrica em que  $x_{candidato}$  e  $y_{candidato}$  representam as coordenadas 2D do candidato e  $x_{perito}$  e  $y_{perito}$  as coordenadas 2D do pedículo identificado pelo perito.

$$distancia = \sqrt{(x_{perito} - x_{candidato})^2 + (y_{perito} - y_{candidato})^2} \quad (4.5)$$

A métrica *accuracy* é obtida para todo o conjunto de treino e representa a percentagem de imagens nas quais o máximo global da área de pesquisa se encontra dentro de um determinado limiar (*threshold*) relacionado com o raio do erro do operador. O seu valor varia entre 0% e 100%. Esta medida tem como objectivo verificar a capacidade do classificador em colocar o melhor candidato a pedículo numa área próxima à sua localização dada pelo técnico. A última métrica apresentada é a *accuracy* em função do raio do erro do operador (*threshold*) que é um gráfico semelhante à curva ROC. Nesta medida podemos verificar a variação da *accuracy* consoante o erro do operador considerado, com o objectivo de demonstrar qual seria o limiar em que a *accuracy* se torne 100%. Na figura 4.4 visiona-se um exemplo deste gráfico. Ainda com base neste gráfico podemos calcular a área abaixo

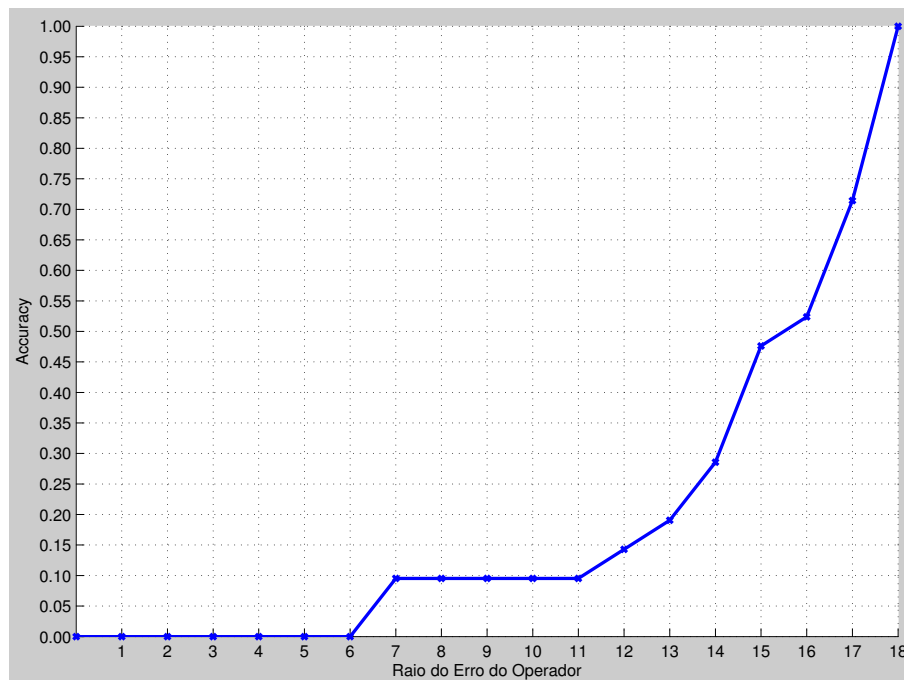


Figura 4.4: Exemplo do gráfico *accuracy* em função do raio do erro do operador.

dele (AUC) de uma forma semelhante ao método utilizado na curva ROC. Este cálculo permite transformar esta métrica numa medida única mais fácil de ser comparada e independente do valor do limiar (*threshold*), raio do erro do operador, cujo valor é difícil de determinar. O erro de um técnico especialista é de cerca de 1 mm [KCLL07]. As imagens utilizadas neste trabalho têm uma resolução de aproximadamente 0,4 mm/pixel, resultando num erro esperado de cerca de 7,5 pixels considerando a sua distribuição como normal. O valor considerado para o limiar (*threshold*) foi 6, sendo que caso a distância

entre a detecção e a referência seja  $\leq 6$  é considerado um verdadeiro positivo, caso contrário é um falso positivo.

Outro factor que pode ser explorado a nível das métricas é a suavização da saída do classificador, recorrendo-se a um filtro gaussiano. A suavização é feita com o objectivo de permitir eliminar extremos causado por algum ruído nos valores de saída. Na apresentação da experiência final vai ser utilizada este método, com a optimização do sigma, para obter os melhores resultados possíveis e para permitir outro tipo de análises.

Por fim, é proposta a utilização de uma métrica de ranking tendo em vista quantificar a associação entre valores altos na saída do classificador e proximidade à referência (*ground-truth*). Considerando  $P$  o conjunto dos candidatos e feita a sua selecção de modo semelhante às das outras métricas é criado um ranking,  $R$ , entre eles. Ele consiste numa lista ordenada pelo valor do resultado do classificador (*score*), sendo a primeira posição dada pelo valor mais alto, dado pelo classificador a cada um. O ranking é uma medida que consiste em seleccionar o elemento de  $P$  mais próximo do pedículo e determinar a sua posição em  $R$  obtendo assim o valor desejado. Esta métrica varia entre 1, ser o primeiro do ranking, e o número de elementos do conjunto  $P$ . Em termos de análise quanto menor o valor do ranking melhor é o resultado.

Para avaliar as métricas ao longo de todo o conjunto de teste são utilizadas várias estatísticas que serão utilizadas no capítulo 5. Estas incluem a utilização da média, do desvio-padrão, do mínimo, do máximo, da mediana e do percentil 25 e 75. Algumas das medidas podem ser visionadas recorrendo a diagramas de extremos e quartis (*box-plots*), que contêm a mediana, o mínimo, o máximo e os percentis 25 e 75.

### 4.4.1 Experiência 1: Avaliação dos descritores, classificadores e pré-processamento

Esta experiência consistiu em comparar os diferentes descritores seleccionados entre si nos mais diferentes cenários. Foram realizados testes para cada um dos classificadores, SVM, RNA e Naive Bayes com o objectivo de verificar qual das combinações descritor classificador obtinha melhores resultados. Ainda nesta experiência foi verificado o modo como os factores do ruído e da rotação da vértebra afectam as diferentes combinações utilizando três configurações possíveis: sem eliminação de ruído e sem eliminação da rotação, com eliminação de ruído e sem eliminação da rotação e com eliminação de ruído e com eliminação da rotação. Pretendeu-se observar como estes elementos afectam os diferentes descritores e classificadores no domínio deste problema. Para esta experiência foram definidos *a priori* os parâmetros de cada um dos descritores que se mantiveram inalterados ao longo deste primeiro teste. Os momentos invariantes e momentos estatísticos não possuem parâmetros a optimizar e para o MCO e o Medidas MCO foram utilizados os apresentados neste capítulo que foram as possibilidades consideradas adequadas para este problema. No caso do SURF foi utilizada a sua versão de 64 dimensões pela sua

maior celeridade de execução quando comparada com a versão de 128 dimensões. Os parâmetros do descritor Daisy foram seleccionados empiricamente e os seus valores foram 6 para o raio ( $R$ ), 3 para o número de convoluções ( $Q$ ), 6 para o número de histogramas por camada e 8 para o número de direcções por histograma. Não foi utilizada interpolação espacial, foi utilizada a interpolação em camadas e, por último, a normalização foi total (tipo 2). Relativamente ao HOG, seleccionou-se um bloco de  $3 \times 3$  ( $nX$  e  $nY$ ), 10 em relação ao número de orientações e normalização global para todos os blocos com o máximo (tipo 2). Nesta comparação ainda existem factores comuns a todos os descritores como o tamanho do bloco da imagem utilizado e o nível de sobreposição entre eles, para os quais foi seleccionado o valor de  $40 \times 40$  para o tamanho do bloco e sobreposição 0 (sem sobreposição de blocos). Finalmente foi necessária a selecção dos parâmetros explorados em cada um dos classificadores. Nas SVM foi feita uma pesquisa em grelha para o  $C$  e o  $\gamma$  com 441 possibilidades em que cada um variou no intervalo entre  $2^i$  com  $i = -10, -9, \dots, 9, 10$ . A nível da ponderação (*weighting*) foi escolhido empiricamente o peso de 33, 3 para a classe positiva. Para o outro classificador com parâmetros passíveis de optimização, a rede neuronal artificial, foi seleccionada a função retropropagação resiliente (*trainrp*) e para o número de neurónios escondidos o intervalo explorado variou entre um valor pequeno, 1 ou 10, dependendo do número de dimensões do descritor, até um valor igual ao dobro deste mesmo número de dimensões. Nesta experiência as métricas utilizadas foram: a *accuracy*, a distância e a AUC do gráfico da *accuracy* em função do raio do erro do operador.

#### 4.4.2 Experiência 2: Avaliação do impacto dos parâmetros do HOG

A segunda experiência consistiu na selecção da melhor combinação obtida na experiência anterior que consistiu na combinação HOG e SVM, com eliminação de ruído e com a eliminação da rotação e comparar os diferentes parâmetros do descritor seleccionado. Cada um dos parâmetros será testado individualmente, mantendo os restantes fixos em relação à configuração original, com o intuito de verificar a sua variação independentemente dos outros. Foram experimentados todos os parâmetros do HOG, o seu bloco ( $nX$  e  $nY$ ), o número de orientações e a normalização. Relativamente ao primeiro foram testadas as configurações  $2 \times 2$ ,  $3 \times 3$ ,  $4 \times 4$  e  $5 \times 5$ , para o número de orientações 4, 8, 10, 12 e 16, e por fim, todas as normalizações já apresentadas. Neste teste foram também exploradas as variações do tamanho do bloco da imagem e do nível de sobreposição. Relativamente ao primeiro factor foram testadas três configurações,  $10 \times 10$ ,  $20 \times 20$  e  $40 \times 40$ . Existiu nesta comparação o cuidado de a área de varrimento de cada uma das configurações ser sempre a mesma,  $20 \times 20$  pixels, variando apenas a área coberta da imagem dependendo do tamanho do bloco utilizado. Em relação ao nível de sobreposição foram também exploradas três possibilidades, sem sobreposição, sobreposição de metade e sobreposição

de 3/4 do bloco anterior. Nesta experiência as métricas utilizadas foram as mesmas que na experiência 1: a *accuracy*, a distância e a AUC do gráfico da *accuracy* em função do raio do erro do operador.

### **4.4.3 Experiência 3: Avaliação do HOG com parâmetros otimizados e do pós-processamento**

A experiência final consistiu em utilizar os parâmetros que obtiveram melhores resultados na experiência 2 e comparar essa configuração com a obtida da primeira experiência. Foi feita uma análise mais extensa em relação à avaliação da saída do classificador calculando as mesmas métricas que na segunda experiência com a inclusão de uma outra, o Ranking. Nesta experiência foi também testado o impacto de aplicar um filtro gaussiano à saída da SVM. A suavização é feita com o objectivo de permitir eliminar extremos causado por algum ruído nos valores de saída. A experiência consistiu em testar diferentes valores de sigma da distribuição gaussiana e medir a performance da classificação através das métricas da *accuracy* da detecção e da média da distância, otimizando o valor de sigma.

## **4.5 Sumário**

Neste capítulo foram apresentados os detalhes de implementação da metodologia proposta. No início foram apresentados os materiais utilizados neste trabalho, incluindo a origem das imagens radiológicas e suas características. Neste capítulo foram também apresentados os sete descritores, bem como os três classificadores seleccionados para as experiências destinadas a verificar qual a melhor combinação dos vários factores. Para cada um destes elementos são apresentadas as origens das suas implementações, os parâmetros que possuem e a forma como estes podem ser variados. No final são descritas as três experiências efectuadas, com o seu objectivo, parâmetros testados e métricas utilizadas para a avaliação.

# Capítulo 5

## Resultados

Ao longo deste capítulo são relatados os resultados obtidos nas três experiências descritas no capítulo 4.

### 5.1 Experiência 1: Avaliação dos descritores, classificadores e pré-processamento

Nesta primeira experiência são testadas todas as combinações possíveis entre os classificadores e os descritores seleccionados. Nesta secção são apresentados gráficos da *accuracy* em função do raio do erro do operador, em que estão presentes todos os descritores em simultâneo para cada um dos três classificadores experimentados. São também mostrados diagramas de extremos e quartis (*box-plots*) relativos à distância do candidato ao pedículo para cada descritor, em que está representada a mediana, o máximo, o mínimo, o percentil 25, o percentil 75 e os valores discrepantes (*outliers*), representados pelo símbolo +, que são os valores que se encontram a uma distância superior a  $+/- 3\sigma$ . Em cada um dos classificadores foram ainda testados outros dois factores a presença de ruído e a rotação das vértebras, resultando em três cenários distintos:

- 1º Cenário: Sem eliminação de ruído e sem eliminação de rotação
- 2º Cenário: Com eliminação de ruído e sem eliminação de rotação
- 3º Cenário: Com eliminação de ruído e com eliminação de rotação

#### 5.1.1 Naive Bayes

##### 5.1.1.1 1º Cenário: Sem eliminação de ruído e sem eliminação de rotação

A figura 5.1 apresenta os resultados para o classificador Naive Bayes para o primeiro cenário em que não está presente a eliminação de ruído nem a eliminação de rotação. Nesta primeira comparação nota-se que o HOG obtém os melhores resultados de forma

## Resultados

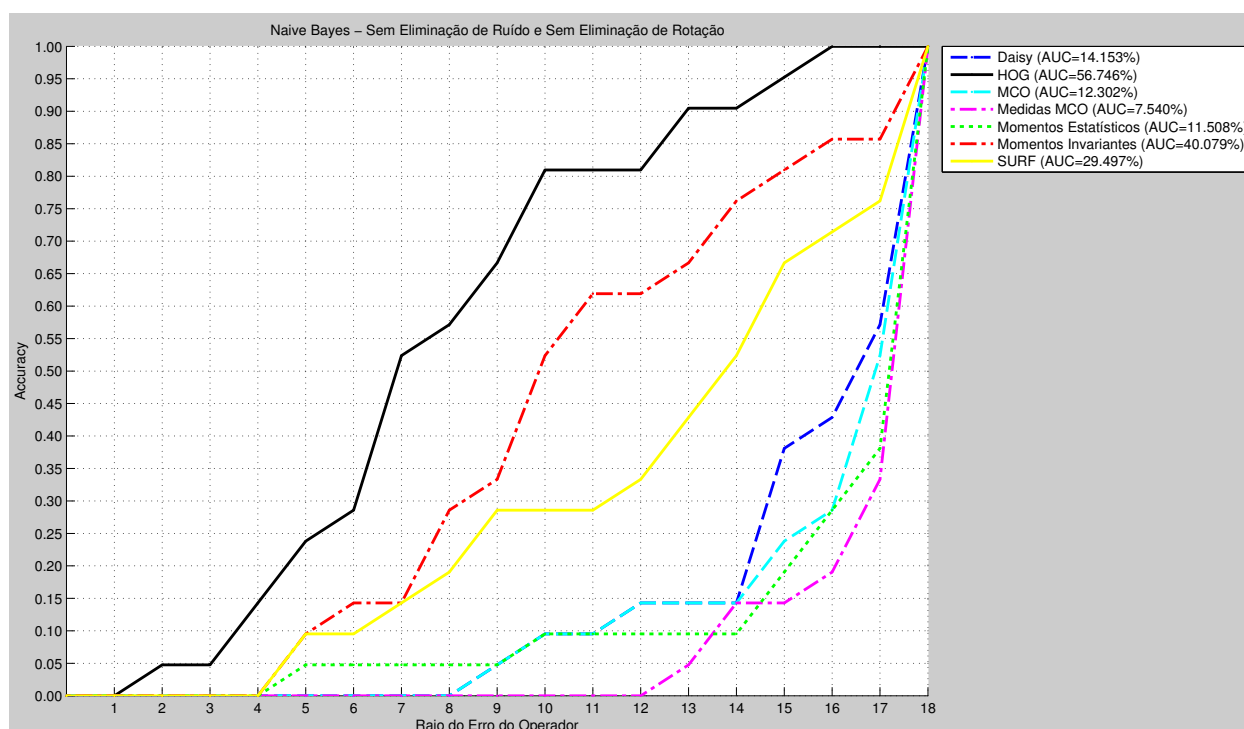


Figura 5.1: Gráficos da *accuracy* em função do raio do erro do operador para o Naive Bayes no primeiro cenário.

clara, tendo uma área debaixo da curva (AUC) de 56.7%. Obtém também a melhor *accuracy* nos valores próximos do erro do técnico (entre 5 e 7), sendo de 28.5% para erro de 6 pixels. O momentos invariantes e o SURF obtêm o segundo e terceiro lugar, respectivamente, embora já algo distantes do HOG. Obtêm valores de 14.0% e 9.5% para a *accuracy* para erro de 6 pixels. Relativamente aos outros descritores são na sua generalidade muito fracos, não ultrapassando os 15% na AUC. A nível de *accuracy*, também não ultrapassam os 5% e na sua maioria obtêm uma *accuracy* de 0% para um raio do erro do operador de 6 pixels. Na figura 5.2 é mostrada a comparação das distâncias obtidas pelos descritores nesta configuração, recorrendo a diagramas de extremos e quartis. Verificamos novamente que o HOG obtém os melhores resultados de forma clara, tendo uma mediana de 6.4 pixels e uma média de 7.6 pixels. O valor da mediana encontra-se mais próximo do percentil 25 e a distância entre os dois quartis é mais curta, quando comparada com os outros dois descritores com melhor mediana, o SURF e o momentos invariantes. Em paralelo com a ordem do gráfico da *accuracy* em função do raio do erro do operador, o momentos invariantes e o SURF são o segundo e terceiro melhor, tendo no entanto medianas de, respectivamente, 9.8 e 13.6 pixels. Todos os restantes descritores apresentam valores demasiado elevados, tendo sempre medianas superiores a 16 pixels. Neste cenário o HOG é claramente o melhor descritor segundo os critérios utilizados na avaliação e será ponto de comparação com os cenários seguintes.

## Resultados

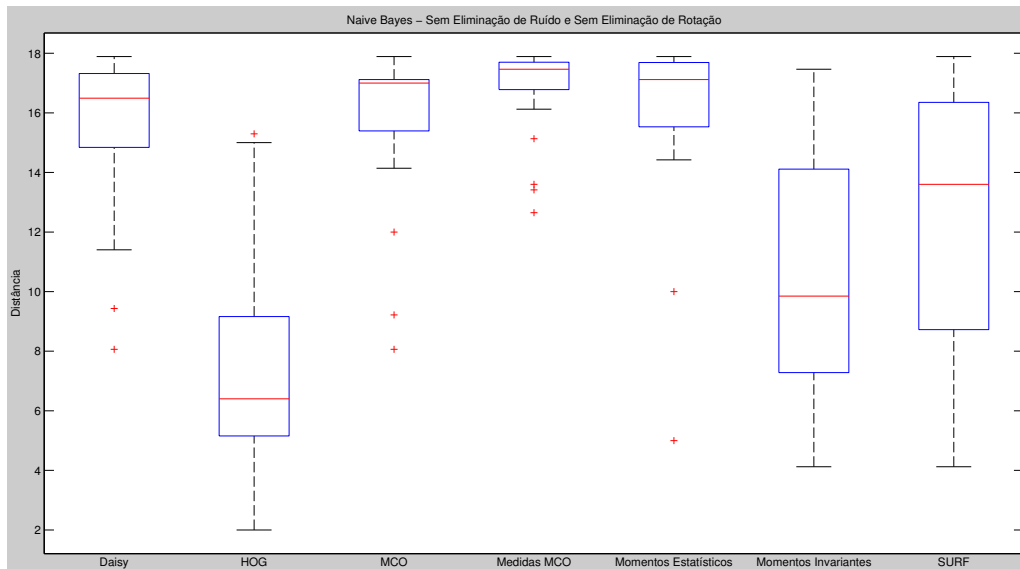


Figura 5.2: Diagramas de extremos e quartis da distância para o Naive Bayes no primeiro cenário.

### 5.1.1.2 2º Cenário: Com eliminação de ruído e sem eliminação de rotação

Na configuração em que existe eliminação de ruído, mas em que se continua a não eliminar a rotação apresenta-se na imagem 5.3 os resultados obtidos pelos descritores. Neste caso a liderança do HOG não é clara, tendo resultados muito semelhantes aos momentos invariantes. A nível de AUC, obtêm valores muito próximos, 49.6% para o HOG e 48% para os momentos invariantes. A nível de *accuracy* para um raio do erro do operador de 6 pixels os resultados são aproximados, mas com vantagem para os momentos invariantes com 28.6% contra os 23.8% do HOG. A generalidade dos restantes descritores tem resultados fracos, sendo o SURF o que apresenta o melhor desempenho com AUC de 32.7%. Na figura 5.4 mostra-se a comparação das distâncias obtidas pelos descritores nesta configuração, recorrendo a diagramas de extremos e quartis. A nível da distância vemos que é o HOG que obtém os melhores resultados, com uma mediana de 8.0 pixels e média de 9.0 pixels. Continua próximo dos momentos invariantes cuja mediana é de 9.4 pixels, mas possui uma ligeira vantagem contrariamente ao que se passava na *accuracy*. Os restantes descritores têm um comportamento semelhante ao descrito no gráfico, com resultados mais modestos com um ligeiro destaque para o SURF. As suas medianas variam entre 12.2 pixels do SURF e os 17.0 pixels do Daisy. Numa comparação feita com a configuração em que existia a presença de ruído verificamos que os resultados pioram ligeiramente para os melhores descritores e melhoram ligeiramente para os descritores com piores resultados. Analisando os diagramas de extremos e quartis verificamos que a mediana sobe na maior parte dos casos e que o mínimo diminui em quase todos.

## Resultados

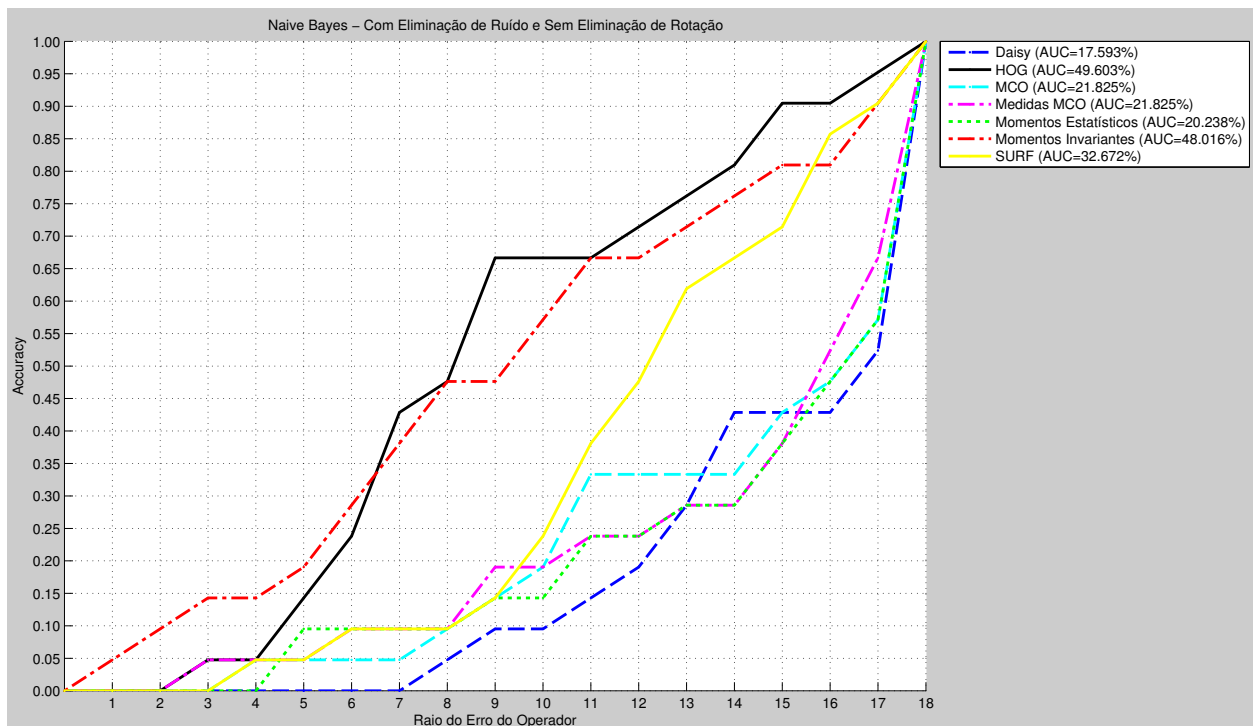


Figura 5.3: Gráficos da *accuracy* em função do raio do erro do operador para o Naive Bayes no segundo cenário.

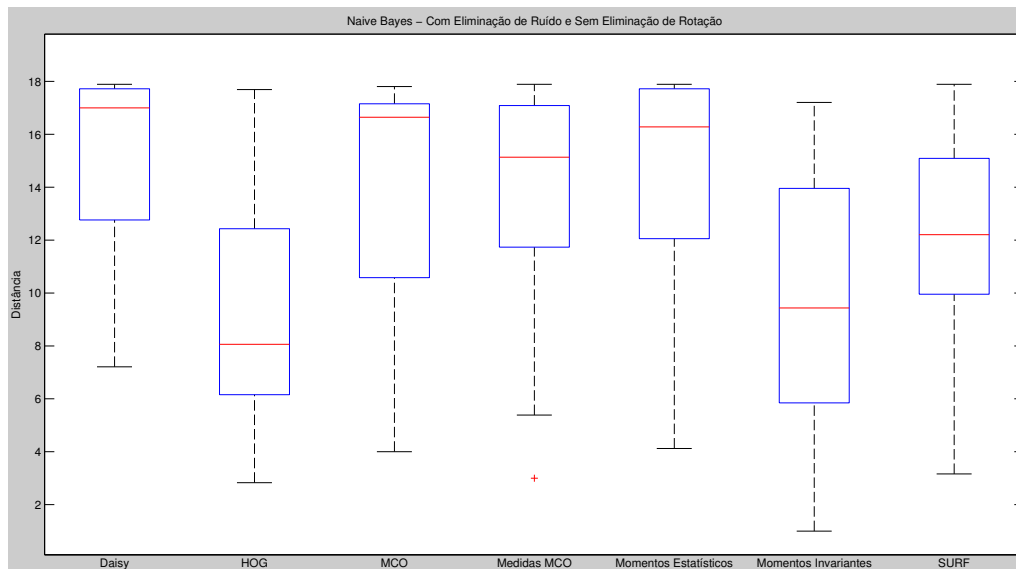


Figura 5.4: Diagramas de extremos e quartis da distância para o Naive Bayes no segundo cenário.

### 5.1.1.3 3º Cenário: Com eliminação de ruído e com eliminação de rotação

No último cenário para o classificador Naive Bayes, em que é feita a eliminação do ruído e a eliminação da rotação, apresentam-se os resultados da comparação na figura 5.5.

## Resultados

Observando a figura 5.5 verificamos novamente a liderança clara do HOG. Este descritor

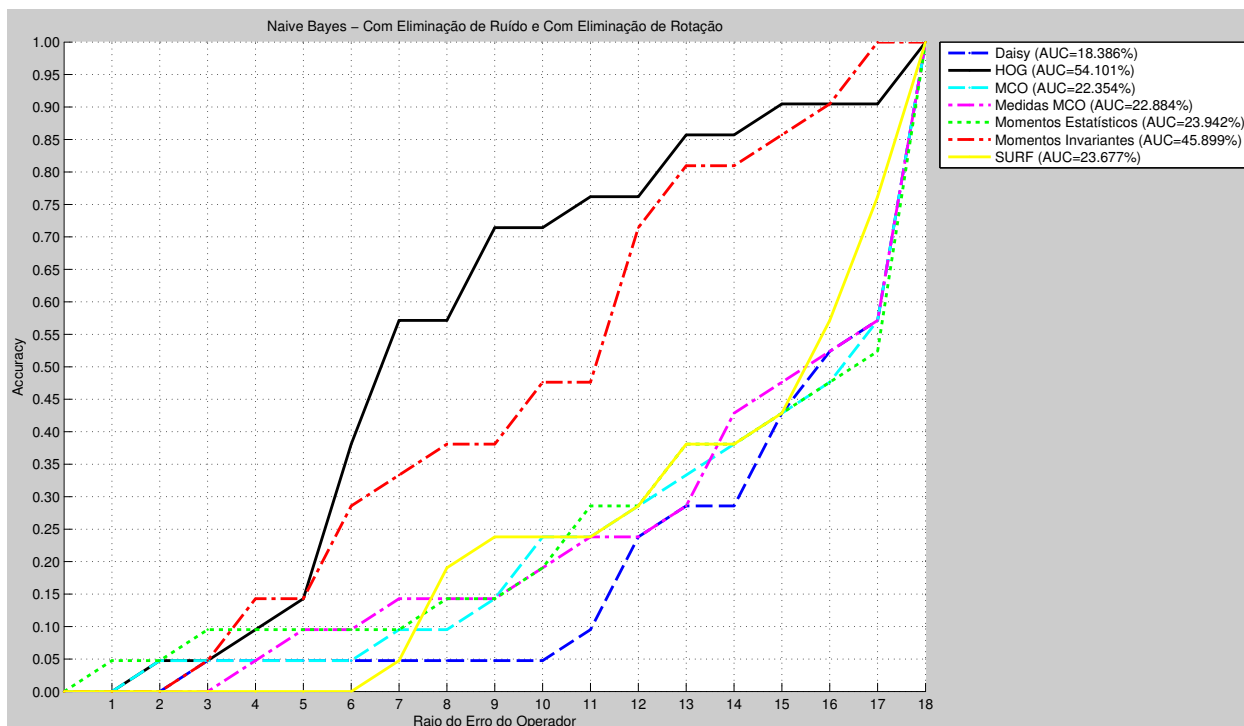


Figura 5.5: Gráficos da *accuracy* em função do raio do erro do operador para o Naive Bayes no terceiro cenário.

obtém uma área de 54.1%, em comparação com os 45.9% obtidos pelo segundo melhor, o momentos invariantes. A nível da *accuracy* o HOG atinge 38.1% para um raio do erro igual a 6 pixels e o momentos invariantes, para o mesmo valor do raio do erro, obtém 28.6%. Para os valores de erro próximos do limiar, o HOG leva vantagem em todos, com a exceção do valor de erro de 5 pixels em que existe um empate. Os restantes descritores obtêm valores mais baixos, sem que nenhum deles se destaque quando comparado com os seus concorrentes (AUC entre 18.4% e 23.7%). A comparação entre as distâncias para este cenário está presente na imagem 5.6. Estudando a imagem 5.6 verificamos novamente uma vantagem do HOG relativamente aos restantes, obtendo uma mediana de 6.3 pixels e uma média de 8.3 pixels. O segundo melhor descritor neste critério continua a ser o momentos invariantes obtendo uma média de 9.7 pixels e uma mediana de 11.2 pixels. Os restantes descritores são comparativamente piores apresentando medianas entre os 15.3 e os 17.0 pixels. De notar que o mínimo da distância é obtido pelo descritor momentos estatísticos. Fazendo uma comparação entre os diferentes cenários deste descritor verificamos que o HOG obtém os melhores resultados a par com o momentos invariantes. É importante realçar a diferença alcançada pelo HOG e pelo momentos invariantes relativamente aos restantes, sendo que desses o SURF foi sempre o terceiro melhor descritor. Comparando os diferentes cenários conclui-se que o melhor resultado é obtido no

## Resultados

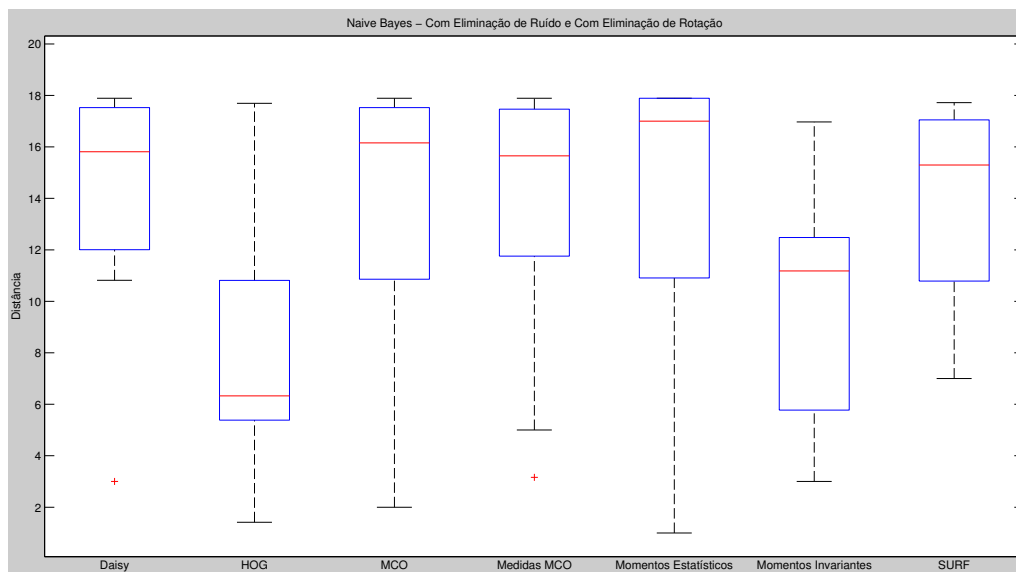


Figura 5.6: Diagramas de extremos e quartis da distância para o Naive Bayes no terceiro cenário.

primeiro cenário com a presença de ruído e sem a eliminação da rotação para o HOG com 56.7% para a AUC contra os 54.1% obtidos no último cenário. A nível de *accuracy* o melhor resultado é obtido no último cenário com 38.1%, contra os 28.5% do primeiro cenário, para um raio de erro do operador igual a 6 pixels. Por fim, no critério da distância o primeiro cenário obtém os melhores valores com mediana de 6.4 pixels e uma média de 7.6 pixels, em comparação com a mediana de 6.3 pixels e média de 8.3 pixels do último cenário. Neste classificador o cenário com melhores resultados foi o primeiro e será a base de comparação com os outros classificadores.

### 5.1.2 Redes Neurais Artificiais

#### 5.1.2.1 1º Cenário: Sem eliminação de ruído e sem eliminação de rotação

Na figura 5.7 são apresentados os resultados para o primeiro cenário do classificador Redes Neurais Artificiais (RNA). Analisando estes resultados verificamos que o HOG e o SURF são os que obtêm melhores resultados. O HOG tem vantagem na AUC com 53.7% em comparação com os 50.9% do SURF. A nível da *accuracy* a vantagem é do SURF com 38.1% contra os 28.6% do HOG, para um valor de 6 pixels para o raio do erro do operador. Avançando para os restantes descritores verificamos que o melhor é o momentos invariantes com AUC de 39.6%, destacando-se dos restantes cuja AUC varia entre 14% e 25%. Os resultados são semelhantes aos melhores do Naive Bayes em termos de valores e com o HOG e o SURF novamente em destaque. Na figura 5.8 mostra-se a comparação das distâncias obtidas pelos descritores neste cenário. Da análise da figura

## Resultados

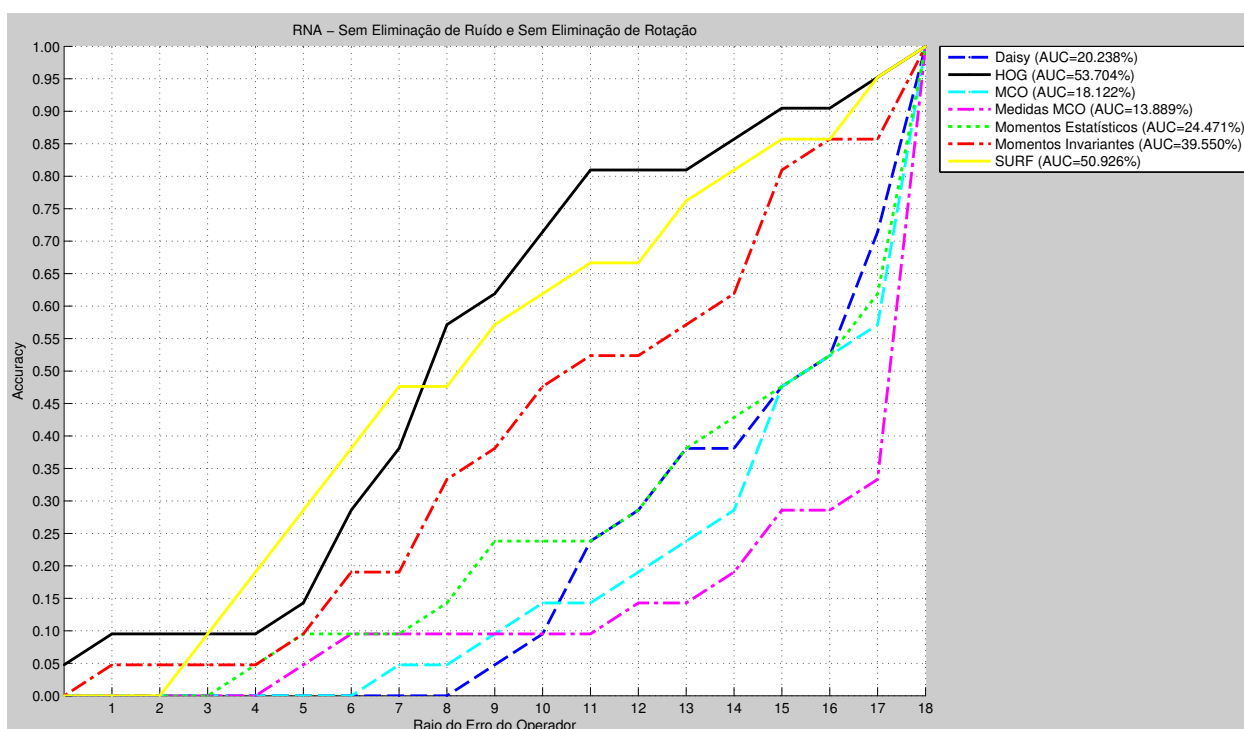


Figura 5.7: Gráficos da *accuracy* em função do raio do erro do operador para o RNA no primeiro cenário.

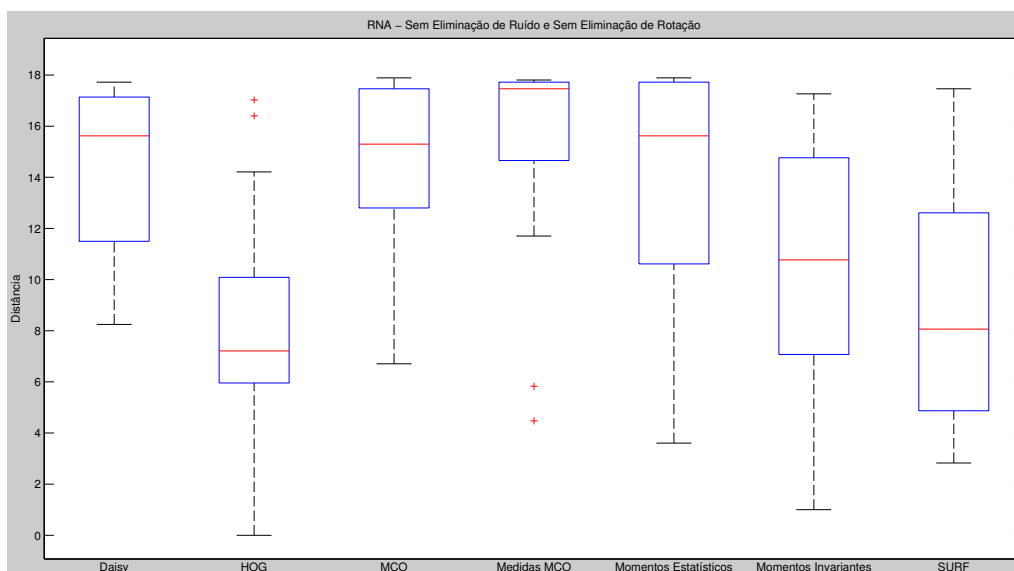


Figura 5.8: Diagramas de extremos e quartis da distância para o RNA no primeiro cenário.

5.8 concluímos que são novamente o SURF e o HOG que obtêm os melhores resultados. O HOG leva vantagem alcançando uma mediana de 7.2 pixels e uma média de 8.2 pixels melhores que os 8.1 pixels para a mediana e os 8.9 pixels para a média alcançados pelo

## Resultados

SURF. Analisando os restantes descritores verificamos que têm resultados mais modestos, em que é novamente o momentos invariantes que se destaca obtendo uma média de 10.8 pixels, sendo que a dos outras varia entre os 13.7 (momentos estatísticos) e os 15.6 pixels (matriz co-ocorrências). Os resultados obtidos neste cenário são semelhantes aos melhores obtidos pelo classificador Naive Bayes, com exceção do critério da distância em que o melhor do Naive Bayes se destacou.

### 5.1.2.2 2º Cenário: Com eliminação de ruído e sem eliminação de rotação

Apresenta-se na figura 5.9 os resultados dos gráficos da *accuracy* em função do raio do erro do operador para o segundo cenário do classificador RNA. Nesta configuração

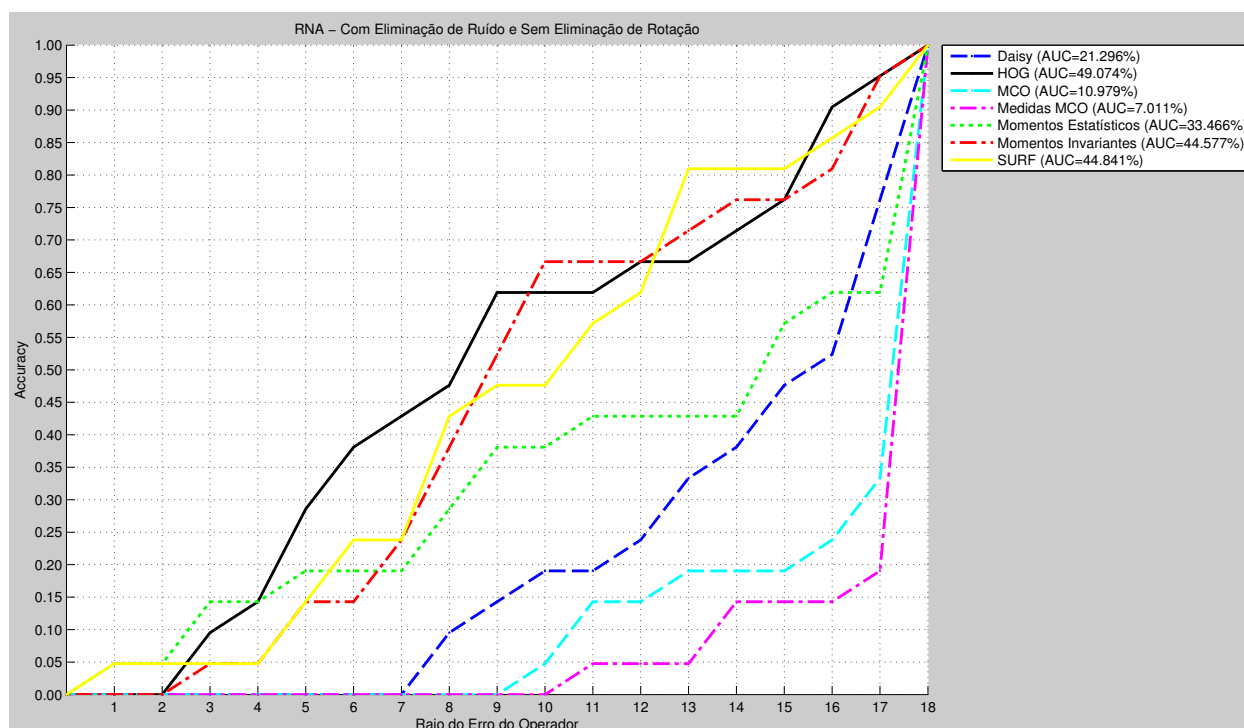


Figura 5.9: Gráficos da *accuracy* em função do raio do erro do operador para o RNA no segundo cenário.

o descritor que obtém melhores resultados é o HOG. Obtém um valor de AUC igual a 49.1% e uma *accuracy* de 38.1% para raio do erro do operador com valor de 6 pixels. O segundo lugar é praticamente compartilhado entre o momentos invariantes e o SURF, com o valor de AUC a ser, respectivamente, 44.6% e 44.8%. Em relação à *accuracy* para um erro de 6 pixels o SURF leva vantagem com 23.8% superiores aos 14.2%, tendo inclusive existido um melhor resultado de 19% por parte do descritor momentos estatísticos. Nos restantes descritores também existe uma pequena diferenciação entre o momentos estatísticos e o Daisy que são melhores que os restantes, embora se mantenham distantes

## Resultados

dos melhores (AUC de 33.5% e 21.3%, respectivamente). Estes resultados aproximam-se dos do melhor obtido com o Naive Bayes. Na imagem 5.10 vemos a comparação entre as distâncias neste cenário. Com base nos diagramas de extremos e quartis da figura 5.10

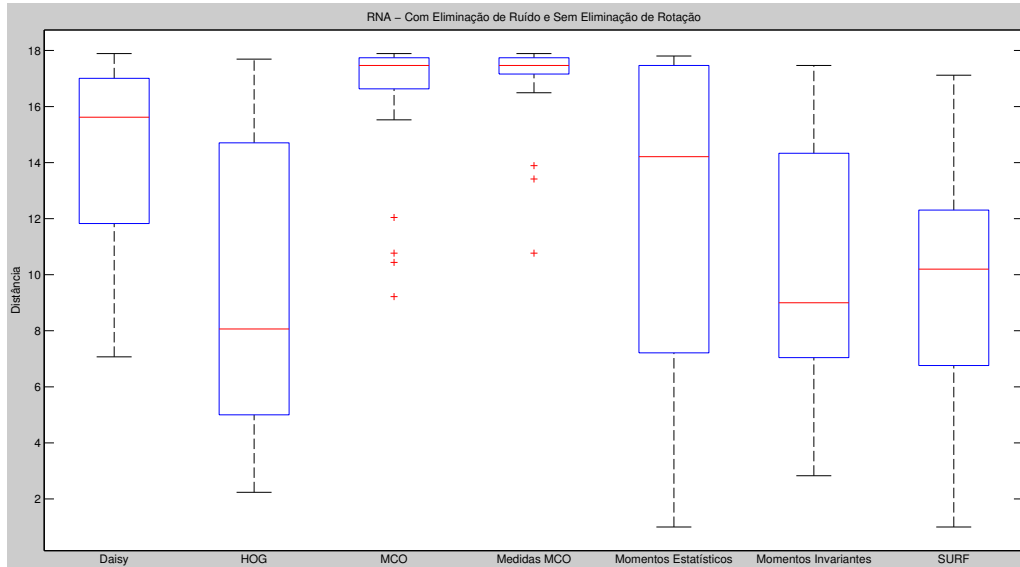


Figura 5.10: Diagramas de extremos e quartis da distância para o RNA no segundo cenário.

verificamos que o HOG tem o melhor resultado, embora com menor diferença para os restantes, nomeadamente o momentos invariantes e o SURF. A nível de medianas o HOG apresenta 8.1 pixels, o momentos invariantes 9.0 pixels e o SURF 10.2 pixels e de médias os valores são, respectivamente, 9.2, 10.0 e 9.9 pixels. Analisando os outros descritores verifica-se a mesma diferença que nos gráficos com o momentos estatísticos e o Daisy a obterem melhores resultados (média de 11.9 e 14.2 pixels) que o MCO e o medidas características da MCO que apresentam resultados muito modestos. Os resultados obtidos neste cenário são inferiores ao cenário anterior, nomeadamente em relação à distância, logo também piores em relação aos melhores obtidos pelo classificador Naive Bayes.

### 5.1.2.3 3º Cenário: Com eliminação de ruído e com eliminação de rotação

O terceiro cenário para o classificador RNA, tem os seus resultados apresentados na figura 5.11. Verifica-se que o HOG continua a ser o melhor descritor com uma percentagem de área de 50.7% e uma *accuracy* de 28.8% para raio do erro do operador igual a 6 pixels. O segundo melhor descritor é o momentos invariantes com AUC de 44.3% e uma *accuracy* igual à do HOG. Nos restantes descritores destacam-se o SURF e o momentos estatísticos que obtêm uma AUC semelhante na ordem do 30%, o Daisy encontra-se próximo com cerca de 25% e os restantes apresentam valores inferiores a 16%. Estes resultados são semelhantes aos melhores até este momento a nível de AUC, apresentando

## Resultados

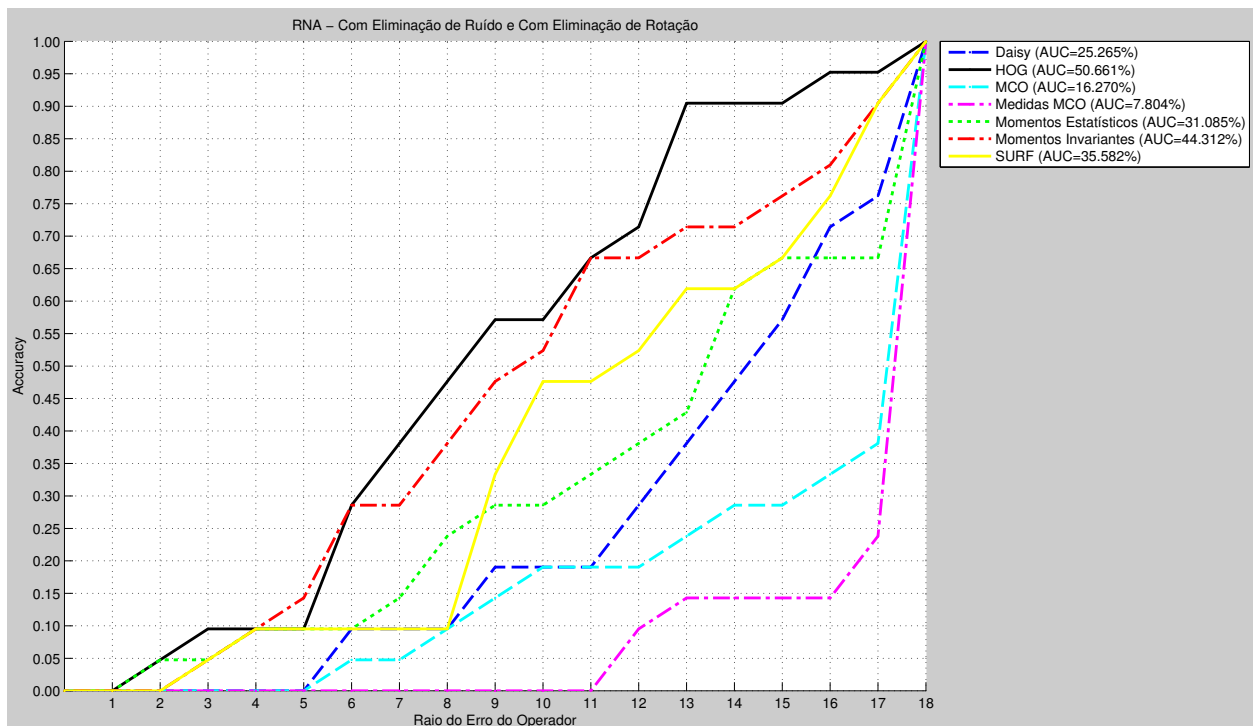


Figura 5.11: Gráficos da *accuracy* em função do raio do erro do operador para o RNA no terceiro cenário.

no entanto um valor mais baixo de *accuracy*. Em 5.12 vemos os resultados obtidos para a distância. Interpretando os valores presentes verificamos que o HOG se continua a des-

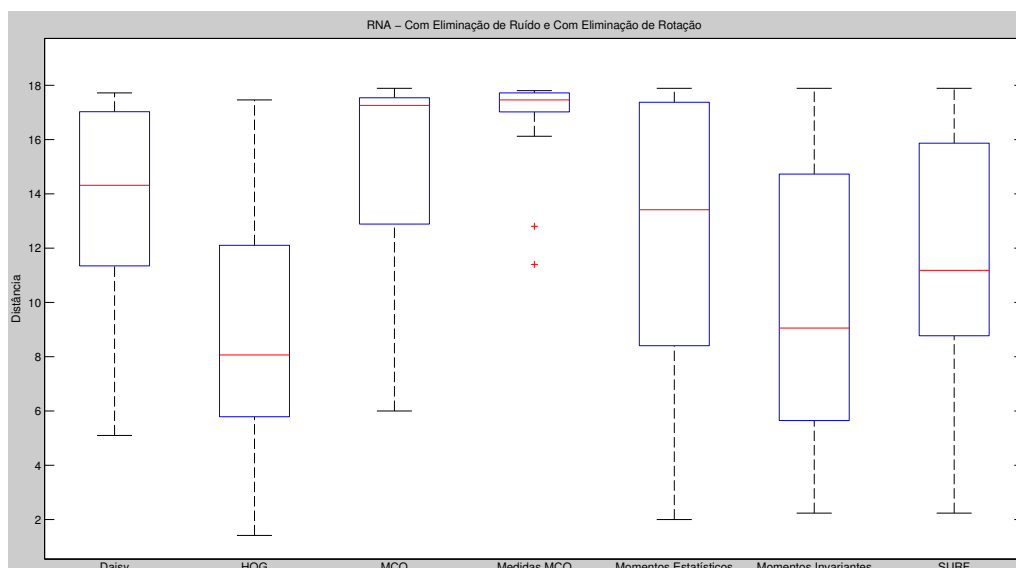


Figura 5.12: Diagramas de extremos e quartis da distância para o RNA no terceiro cenário.

tacar como o melhor. Apresenta uma mediana de 8.1 pixels e uma média de 8.8 pixels,

## Resultados

com performance superior aos 9.0 e 10.0 pixels de mediana e média do segundo melhor, o momentos invariantes. Nos outros descritores destaca-se o SURF, o momentos estatísticos e o Daisy relativamente próximos entre si, enquanto que o matriz de co-ocorrências e o descritor das medidas da MCO estão muito afastados dos restantes.

Finalizada a análise do segundo classificador verificamos que o melhor cenário continua a ser o primeiro do Naive Bayes, principalmente devido ao critério da distância na qual foi o melhor. Ao nível dos outros critérios o classificador Redes Neurais Artificiais apresenta resultados equiparados nomeadamente para o primeiro e último cenário.

### 5.1.3 Máquinas de Suporte Vectorial

#### 5.1.3.1 1º Cenário: Sem eliminação de ruído e sem eliminação de rotação

O classificador SVM para o cenário de existência de ruído e sem a eliminação da rotação obteve os resultados apresentados na figura 5.13. Numa análise ao gráfico verificamos

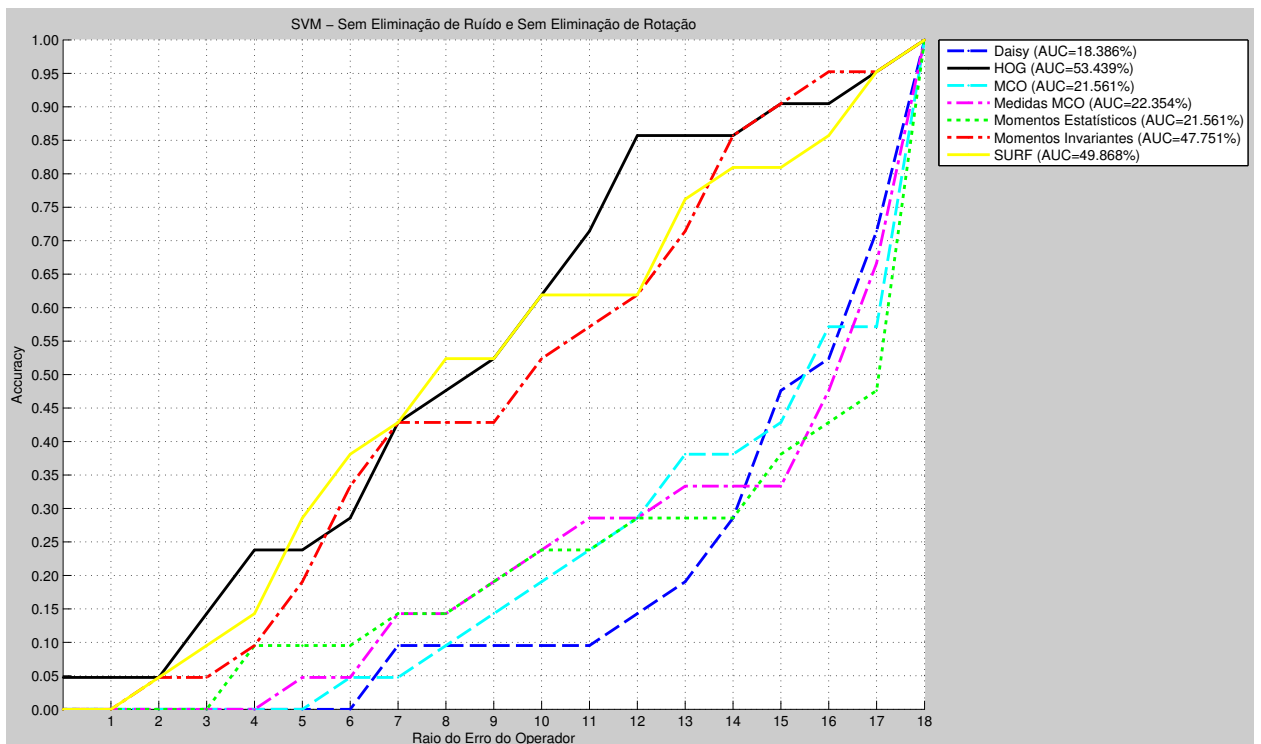


Figura 5.13: Gráficos da *accuracy* em função do raio do erro do operador para o SVM no primeiro cenário.

que existem três descritores que se destacam o HOG, o SURF e o momentos invariantes. Destes o HOG é o que apresenta o melhor valor de AUC com 53.4% contra os 49.9% do SURF e os 47.8% do momentos invariantes. Em termos de *accuracy* o SURF apresenta o melhor resultado com 38.1%, enquanto que o momentos invariantes obtém 33.3% e o

## Resultados

HOG 28.6%, todos para um erro do operador igual a 6 pixels. Os restantes 4 descritores têm resultados próximos com valores de AUC entre 18.4%, caso do Daisy, e 22.4%, no caso da matriz de co-ocorrências. Estes resultados são próximos do melhor cenário até este momento, obtido com o Naive Bayes, em termos de AUC e superiores em termos de *accuracy*. De seguida, na imagem 5.14 apresentamos os resultados da métrica distância. Estudando os vários diagrama de extremos e quartis verificamos que existe novamente um

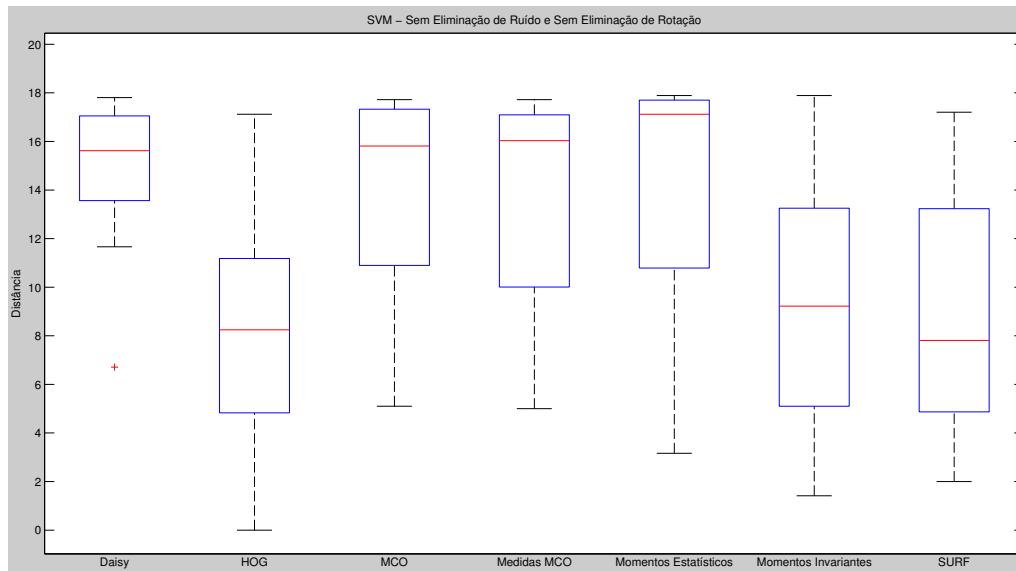


Figura 5.14: Diagramas de extremos e quartis da distância para o SVM no primeiro cenário.

destacar de três descritores, o SURF, o HOG e o momentos invariantes. Destes o SURF obtém a mediana mais baixa com 7.8 pixels, comparada com os 8.2 pixels do HOG e os 9.2 pixels do momentos invariantes. Em termos de média a mais baixa é a do HOG com 8.2 pixels, melhor que os 9.0 pixels do SURF e os 9.4 pixels do momentos invariantes. Os restantes obtêm resultados mais modestos com a média a variar entre 13.8 e os 14.7 pixels, obtidos pelo matriz de co-ocorrências e pelo Daisy, respectivamente. Comparando estes valores com os melhores obtidos até este momento verificamos que estes se aproximam, mas não atingem os melhores até agora do primeiro cenário do Naive Bayes.

### 5.1.3.2 2º Cenário: Com eliminação de ruído e sem eliminação de rotação

O segundo cenário para o classificador SVM tem os seus resultados apresentados na figura 5.15. Interpretando os resultados obtidos conclui-se que existem dois descritores se destacam relativamente aos restantes, o HOG e o SURF. A nível de AUC, o HOG alcança os 52.3% e o SURF os 50.9%, enquanto o terceiro melhor, momentos invariantes, se limita aos 42.5%. Comparando a *accuracy* atingida verificamos que o HOG atinge um valor

## Resultados

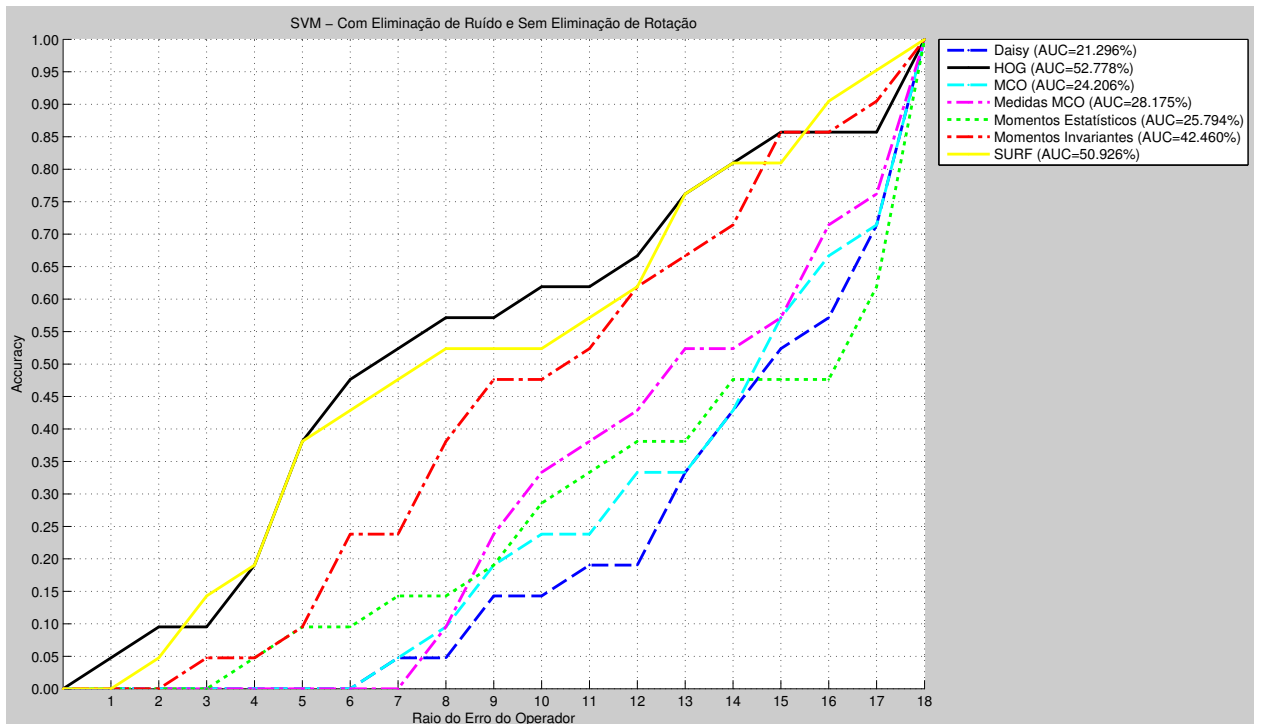


Figura 5.15: Gráficos da *accuracy* em função do raio do erro do operador para a SVM no segundo cenário.

próximo dos 50%, mais concretamente 47.6% e o SURF atinge 42.9%. O melhor dos restantes é novamente o momentos invariantes com 23.8%. Os outros descritores alcançam um valor baixo quer na AUC (entre 21% e 28%) quer na *accuracy* (entre 0% e 9.5%). Estes resultados são superiores ao melhor cenário até este momento, nomeadamente a nível da *accuracy*. Na figura 5.16 apresenta-se a métrica distância para este cenário. Volta-se a verificar que o HOG e o SURF se destacam em relação aos restantes descritores. O HOG obtém uma mediana de 6.3 pixels e uma média de 8.5 pixels, enquanto que o SURF tem como mediana 7.8 e como média 8.8 pixels. Nos outros descritores o momentos invariantes é novamente o melhor com média de 10.3 pixels e os outros quatro são próximos entre si com a média a variar entre 12.9 (matriz de co-ocorrências) e os 14.2 pixels (Daisy). Comparando estes resultados com o melhor até este momento verificamos que esta configuração apresenta uma melhor mediana e uma pior média. Combinando esta informação com a obtida do gráfico conclui-se que esta configuração é a superior até este momento e passa a ser a base de comparação.

## Resultados

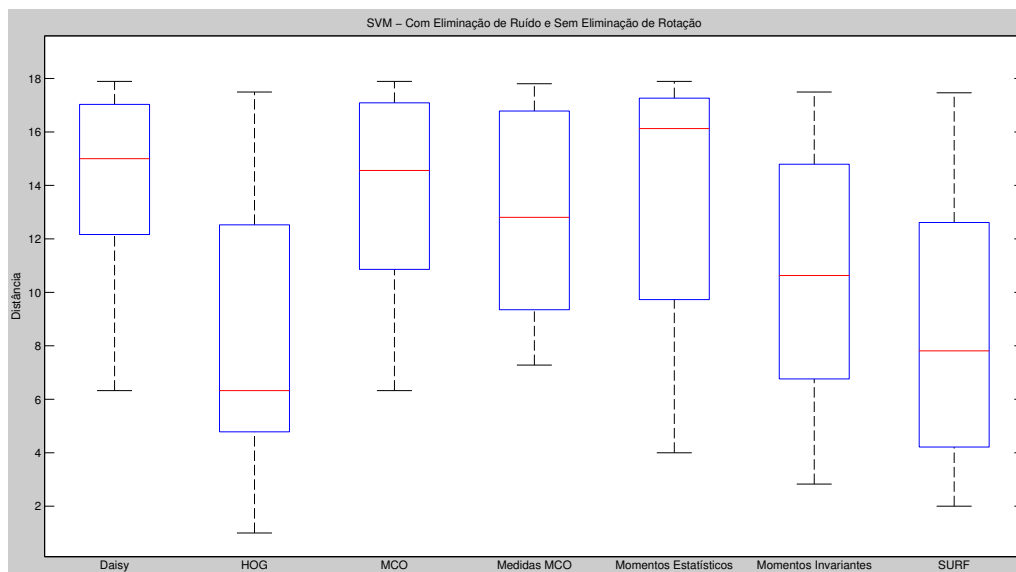


Figura 5.16: Diagramas de extremos e quartis da distância para a SVM no segundo cenário.

### 5.1.3.3 3º Cenário: Com eliminação de ruído e com eliminação de rotação

Na imagem 5.17 vemos os resultados relativos aos gráficos da *accuracy* em função do raio do erro do operador para o terceiro cenário para o classificador SVM. Nesta con-

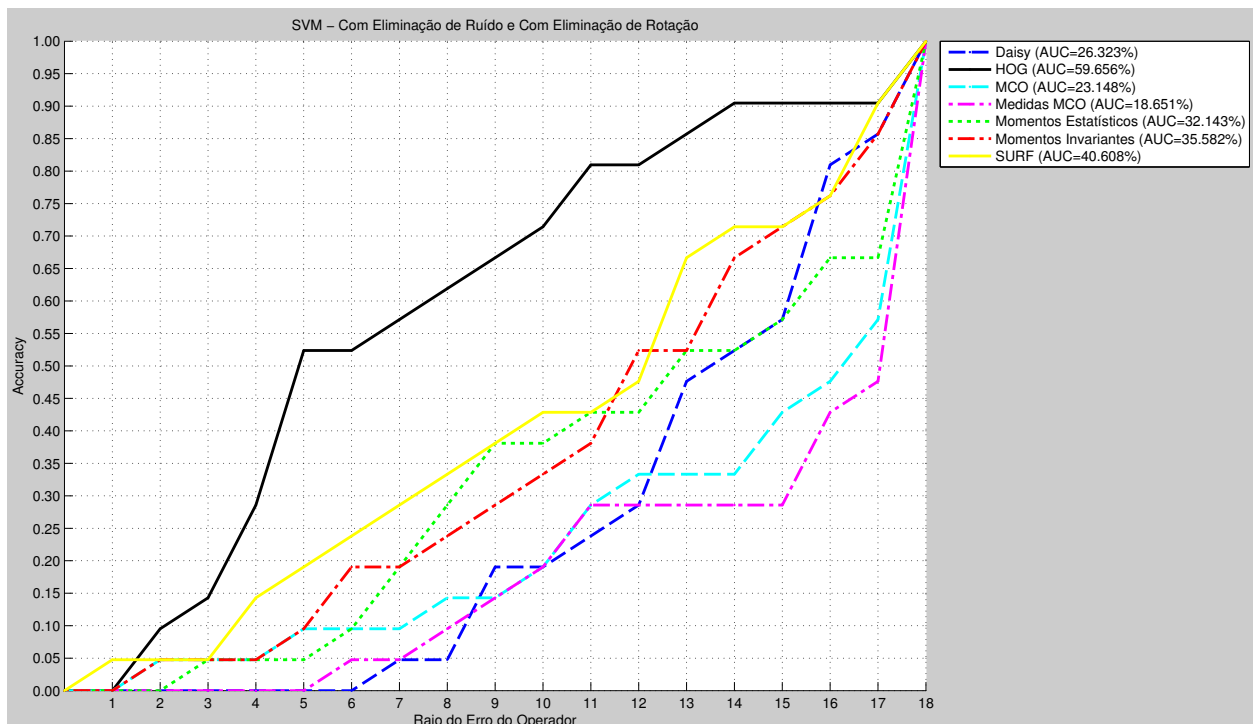


Figura 5.17: Gráficos da *accuracy* em função do raio do erro do operador para a SVM no terceiro cenário.

## Resultados

figuração nota-se um destaque do HOG relativamente aos restantes descritores. Atinge 59.7% na percentagem de área sob a curva e 52.4% na *accuracy* para raio do erro igual a 6 pixels. Analisando os outros descritores verificamos que o SURF ( $AUC = 40.6\%$ ), o momentos invariantes ( $AUC = 35.6\%$ ) e o momentos estatísticos ( $AUC = 32.1\%$ ) se superiorizam aos restantes em que nenhum alcança um valor superior a 27%. Em confronto com as anteriores configurações esta é comparativamente superior, quer a nível de AUC e principalmente a nível de *accuracy* para o limiar de 6 pixels. O gráfico de comparação da métrica da distância desta experiência está presente na figura 5.18. Analisando os di-

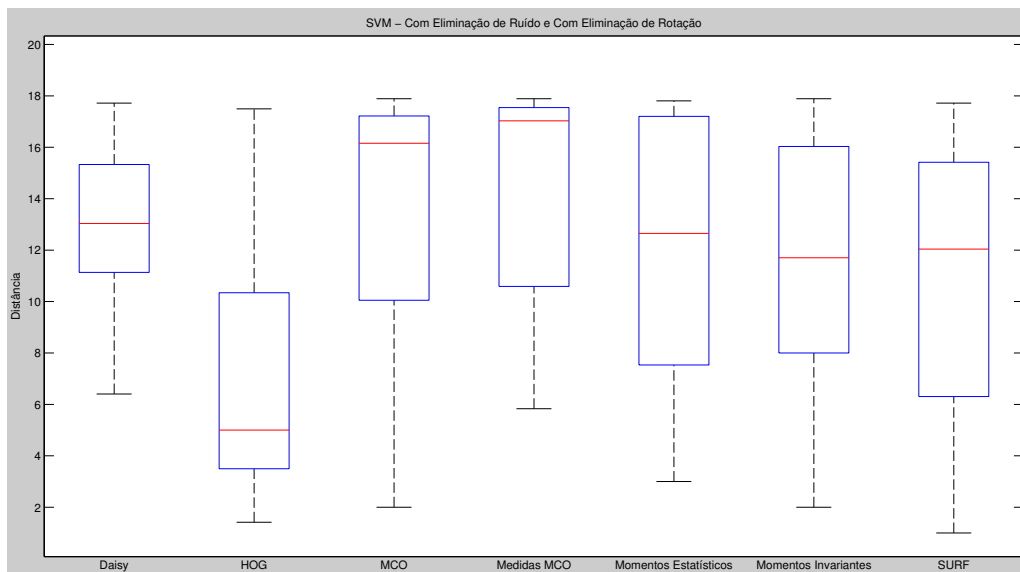


Figura 5.18: Diagramas de extremos e quartis da distância para a SVM no terceiro cenário.

agramas de extremos e quartis anteriores verificamos que o HOG é o melhor com alguma vantagem em relação aos restantes. Obtém uma mediana de 5 pixels e uma média de 7.2 pixels que é visivelmente melhor quando comparado com os 11.7 pixels de mediana e os 11.7 pixels de média alcançados pelo segundo melhor, o momentos invariantes. Entre os restantes nenhum se destaca particularmente, sendo que o MCO e o das medidas características do MCO se encontram num patamar inferior.

Nas tabelas 5.1, 5.2 e 5.3 estão sumariados os resultados obtidos nesta experiência para os classificadores Naive Bayes, RNA e SVM, respectivamente.

Tabela 5.1: Sumário dos resultados obtidos na primeira experiência para o classificador Naive Bayes. As medidas apresentadas são a *accuracy* para raio do erro do operador igual a 6 pixels (ACC), média da distância (MD) e percentagem da área sob a curva (AUC).

Cenário	Daisy			MCO			Medidas MCO			M. Estatísticos			M. Invariantes			HOG			SURF		
	ACC	MD	AUC	ACC	MD	AUC	ACC	MD	AUC	ACC	MD	AUC	ACC	MD	AUC	ACC	MD	AUC	ACC	MD	AUC
1	0.00	15.5	14.2	0.00	15.7	12.3	0.00	16.7	7.54	4.76	16.0	11.5	14.3	10.71	40.1	<b>28.6</b>	<b>7.61</b>	<b>56.7</b>	9.52	12.6	29.5
2	0.00	14.8	17.6	4.76	14.1	21.8	9.52	14.0	21.8	9.52	14.3	20.2	<b>28.6</b>	9.3	48.0	23.8	<b>9.04</b>	<b>49.6</b>	9.52	12.0	32.7
3	4.76	14.7	18.4	4.76	14.0	22.4	9.52	13.9	22.9	9.52	13.8	23.9	28.6	9.69	45.9	<b>38.1</b>	<b>8.29</b>	<b>54.1</b>	0.00	13.8	23.7

Tabela 5.2: Sumário dos resultados obtidos na primeira experiência para o classificador Redes Neurais Artificiais. As medidas apresentadas são a *accuracy* para raio do erro do operador igual a 6 pixels (ACC), média da distância (MD) e percentagem da área sob a curva (AUC).

Cenário	Daisy			MCO			Medidas MCO			M. Estatísticos			M. Invariantes			HOG			SURF		
	ACC	MD	AUC	ACC	MD	AUC	ACC	MD	AUC	ACC	MD	AUC	ACC	MD	AUC	ACC	MD	AUC	ACC	MD	AUC
1	0.00	14.4	20.2	0.00	14.7	18.1	9.52	15.6	13.9	9.52	13.7	24.5	19.0	10.8	39.6	28.6	<b>8.23</b>	<b>53.7</b>	38.1	8.85	50.9
2	0.00	14.2	21.3	0.00	16.1	11.0	0.00	16.8	7.01	19.0	11.9	33.5	14.3	10.0	44.6	<b>38.1</b>	<b>9.17</b>	<b>49.1</b>	23.8	9.90	44.8
3	9.52	13.4	25.3	4.76	15.1	16.3	0.00	16.	7.80	9.52	12.5	31.1	<b>28.6</b>	10.1	44.3	<b>28.6</b>	<b>8.8</b>	<b>50.7</b>	9.52	11.6	35.6

Tabela 5.3: Sumário dos resultados obtidos na primeira experiência para o classificador Máquinas de Suporte Vectorial. As medidas apresentadas são a *accuracy* para raio do erro do operador igual a 6 pixels (ACC), média da distância (MD) e percentagem da área sob a curva (AUC).

Cenário	Daisy			MCO			Medidas MCO			M. Estatísticos			M. Invariantes			HOG			SURF		
	ACC	MD	AUC	ACC	MD	AUC	ACC	MD	AUC	ACC	MD	AUC	ACC	MD	AUC	ACC	MD	AUC	ACC	MD	AUC
1	0.00	14.9	18.4	4.76	14.1	21.6	4.76	13.8	22.4	9.52	14.1	21.6	33.3	9.4	47.8	28.6	<b>8.21</b>	<b>53.4</b>	<b>38.1</b>	9.00	49.9
2	0.00	14.2	21.3	0.0	13.6	24.2	0.00	13.0	28.2	9.52	13.3	25.8	23.8	10.3	42.5	<b>47.6</b>	<b>8.51</b>	<b>52.8</b>	42.9	8.78	50.9
3	0.00	13.1	26.3	9.52	13.8	23.1	4.76	14.6	18.7	9.52	12.3	32.1	19.0	11.7	35.6	<b>52.4</b>	<b>7.24</b>	<b>59.7</b>	23.8	10.6	40.6

### 5.1.4 Discussão

Após os vários testes verificou-se que a melhor configuração consiste na utilização do descritor HOG, o classificador SVM, aplicar o filtro de eliminação de ruído e aplicar a eliminação da rotação das vértebras. Esta combinação obteve praticamente 60% na percentagem da área sob a curva (AUC) e uma *accuracy* de 52.4% para um raio do erro do operador de 6 pixels. A nível da distância obteve uma mediana de 5 pixels e uma média de 7.2 pixels que foram as melhores atingidas em todos os testes efectuados nesta experiência. Nos outros classificadores o primeiro cenário do Naive Bayes e o último da Rede Neuronal atingiram igualmente bons resultados. Em termos da comparação dos dois factores testados, a presença de ruído e a rotação, verificou-se que o HOG beneficiou da eliminação do ruído e da eliminação da rotação, principalmente no SVM. Na generalidade a maioria dos descritores obteve resultados semelhantes nos três cenários, piorando ligeiramente com a eliminação do ruído. Notou-se também que alguns descritores que seriam invariantes à rotação como o momentos invariantes pioraram os seus resultados ao eliminar a rotação. Foram feitos pequenos testes no modo de aplicação da rotação experimentando diferentes modos de interpolação e os resultados mantiveram-se constantes, concluindo que a eliminação da rotação poderá ter introduzido algum ruído que afectou os resultados. Ao longo de todos os testes existiram sempre três descritores que se destacaram, o HOG, o SURF e o momentos invariantes nos diferentes cenários. A nível de classificadores as diferenças causadas não foram muito notórias resultando na conclusão que a selecção do descritor é mais crítica, verificando-se no entanto superioridade da SVM em relação aos outros dois. Esta experiência termina com a selecção da melhor configuração que será utilizada nos testes seguintes.

## 5.2 Experiência 2: Avaliação do impacto dos parâmetros do HOG

Nesta experiência são testados os diversos parâmetros do descritor HOG, utilizando o classificador SVM e a configuração que obteve os melhores resultados na experiência 1. Os critérios de avaliação são os mesmos utilizados nos testes anteriores. Os parâmetros do descritor testados são o bloco do HOG, o número de orientações e a normalização utilizada. Para além disso são testados outros dois factores, o tamanho do bloco da imagem e o seu nível de sobreposição que serão apresentados em primeiro lugar.

### 5.2.1 Tamanho do bloco da imagem

O tamanho do bloco da imagem é um factor que indica a quantidade de pixels utilizados pelo descritor no seu cálculo. As configurações testadas têm o cuidado de garantir que a área de varrimento onde é procurado o pedículo se mantém constante ao longo de todos os testes. Na figura 5.19 apresenta-se a primeira comparação. Analisando os resul-

## Resultados

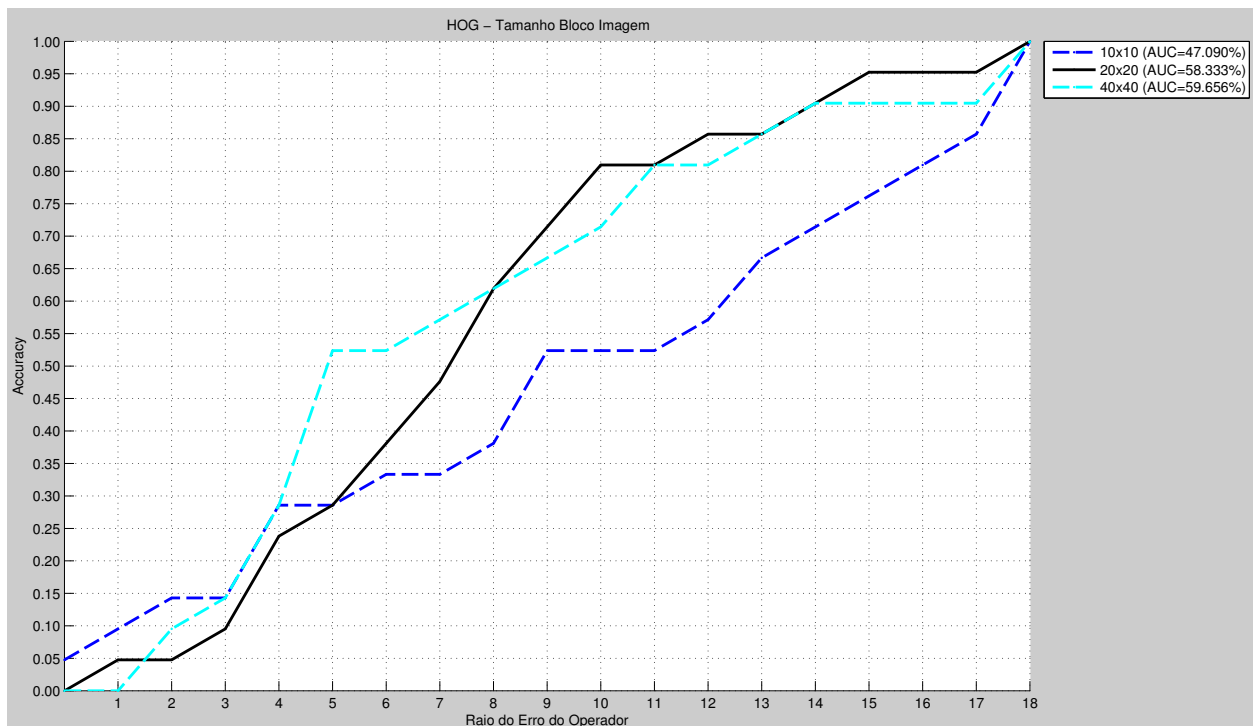


Figura 5.19: Gráficos da *accuracy* em função do raio do erro do operador para a comparação entre o tamanho do bloco da imagem.

tados obtidos verificamos que a configuração de  $40 \times 40$  obteve os melhores resultados, atingindo uma percentagem de área de 59.7% e uma *accuracy* de 52.4% para um erro do operador de 6 pixels. Um bloco de  $20 \times 20$  também obtém bons resultados, com uma AUC semelhante, 58.3%, mas com uma *accuracy* inferior com o erro próximo (5, 6 e 7 pixels) do valor de referência para o erro do técnico. A outra configuração em estudo obtém resultados mais modestos, sugerindo que não é capaz de capturar o padrão do pedículo. Na imagem 5.20 está presente o próximo critério a distância. Interpretando os resultados obtidos verificamos novamente que a configuração  $40 \times 40$  obtém os melhores resultados. A sua mediana é de 5 pixels e a média é de 7.2 pixels. Nas duas outras combinações o bloco de tamanho  $20 \times 20$  destaca-se obtendo mediana de 7.1 pixels e média de 7.5 pixels. A configuração restante obtém resultados piores com uma mediana com valor mais elevado e existindo uma grande variação entre o percentil 25 e o percentil 75. Neste parâmetros do HOG verificamos que o tamanho  $40 \times 40$  é o melhor e será o utilizado na experiência final.

### 5.2.2 Nível de Sobreposição

O próximo parâmetro a analisar é o nível de sobreposição e os resultados relativos ao gráfico da *accuracy* em função do raio do erro do operador encontram-se na figura

## Resultados

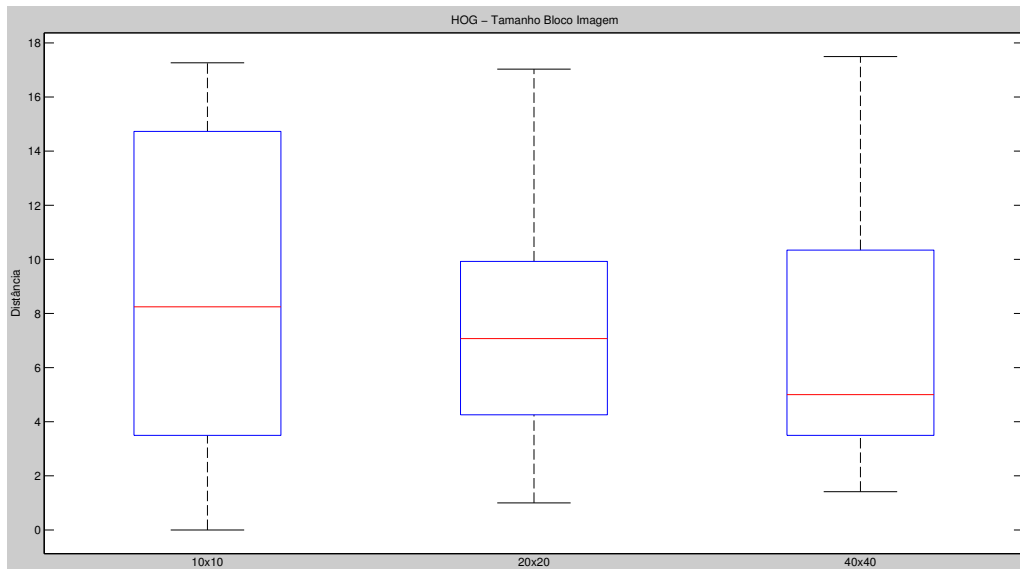


Figura 5.20: Diagramas de extremos e quartis da distância para a comparação entre o tamanho do bloco da imagem.

5.21. Analisando os três gráficos verificamos que a configuração sobreposição 4 obtém o

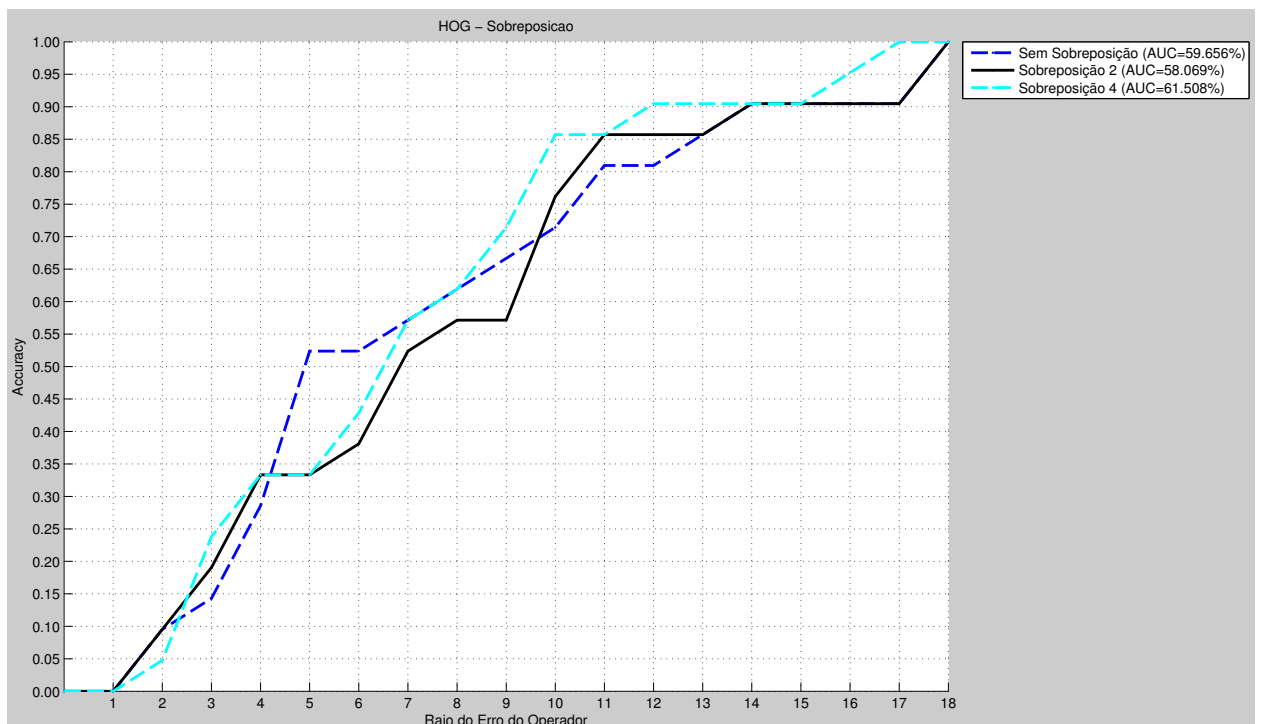


Figura 5.21: Gráficos da *accuracy* em função do raio do erro do operador para a comparação entre o nível de sobreposição do bloco da imagem.

melhor resultado na percentagem de área sob a curva (AUC). Nesta configuração existe

## Resultados

sobreposição de 3/4 do bloco. Atinge 61.5% na AUC, ligeiramente superior aos 59.7% da configuração sem sobreposição e aos 58.1% da configuração sobreposição de metade do bloco (sobreposição 2). Relativamente à *accuracy* verificamos que a configuração sem sobreposição obtém os melhores valores para erros 5, 6 e 7 pixels. Por exemplo para erro igual a 6, a hipótese sem sobreposição alcança 52.4% em confronto com os 42.9% da sobreposição 4, 3/4 do bloco, e os 38.1% da sobreposição 2, metade do bloco. Para a decisão de qual das configurações é a melhor é necessário ainda analisar o critério da distância, cujo gráfico comparativo se encontra na figura 5.22. No critério da distância

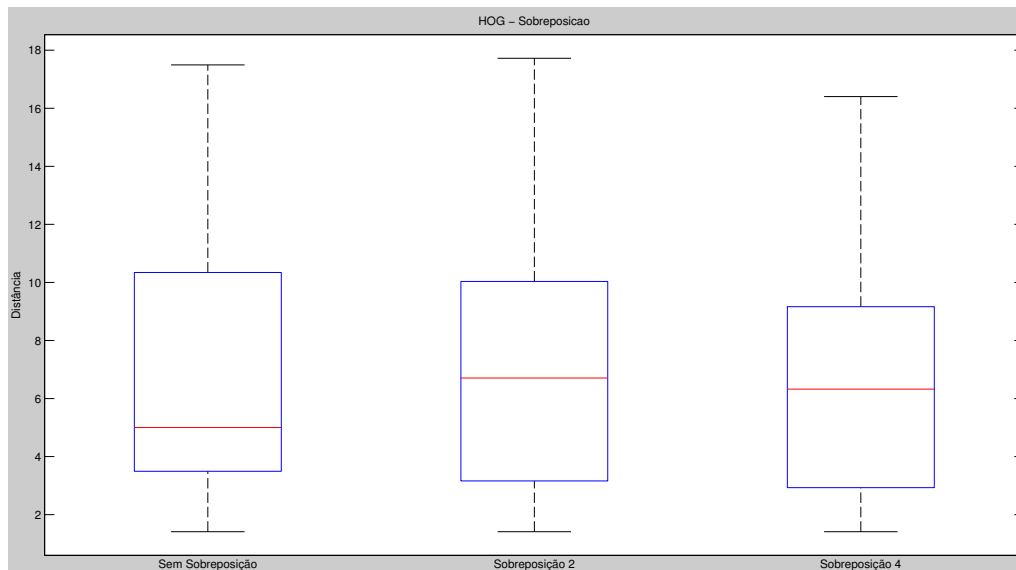


Figura 5.22: Diagramas de extremos e quartis da distância para a comparação entre o nível de sobreposição do bloco da imagem.

concluimos que a não sobreposição é a melhor configuração. Atinge uma mediana de 5 pixels e uma média de 7.3 pixels. As outras duas configurações atingem resultados próximos com uma mediana de 6.7 pixels para a sobreposição de metade do bloco (sobreposição 2) e de 6.3 pixels para a sobreposição de 3/4 do bloco (sobreposição 4). Neste critério reparamos que a não sobreposição é o que atinge os melhores resultados e será o explorado na experiência 3.

### 5.2.3 Tamanho do bloco do HOG

O primeiro parâmetro do HOG que vai ser testado é o tamanho do seu bloco. Na figura 5.23 temos o resultado relativo ao gráfico da *accuracy* em função do raio do erro do operador para este parâmetro. Analisado as diferentes combinações verificamos que existem duas que se destacam a com o tamanho de bloco de  $3 \times 3$  e a de bloco  $4 \times 4$ . Destas a melhor é a de  $4 \times 4$  com AUC de 63.4% superior aos 61.2% da configuração de

## Resultados

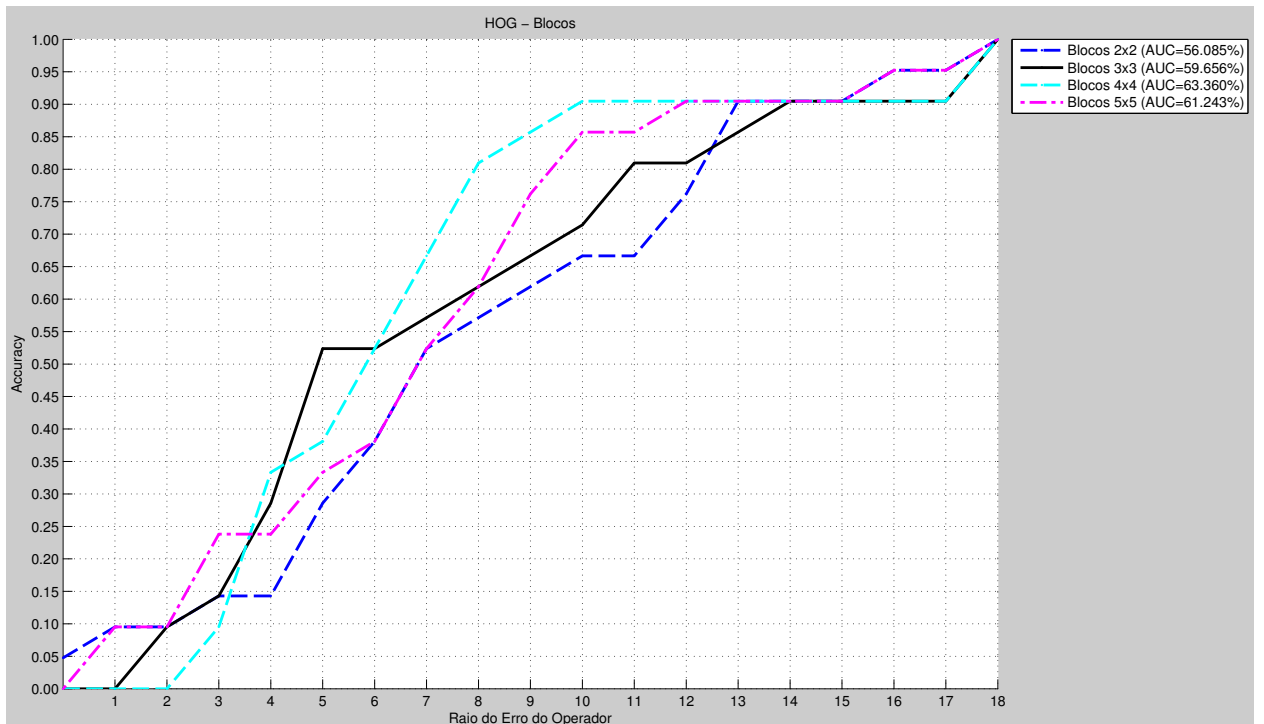


Figura 5.23: Gráficos da *accuracy* em função do raio do erro do operador para a comparação entre o tamanho do bloco do HOG.

$3 \times 3$ . Em termos de *accuracy* existe um empate para erro igual a 6 pixels com 52.4%. As restantes configurações obtêm resultados ligeiramente inferiores, embora em termos de *accuracy* não ultrapassem os 38.1% para o erro do operador igual a 6 pixels. Para terminar a análise deste parâmetro temos na imagem 5.24 a comparação do critério de distância. Comparando as diferentes combinações nenhuma se distingue de forma clara, com medianas relativamente próximas entre si: 7.0 pixels para o tamanho  $2 \times 2$ , 5.0 pixels para o tamanho  $3 \times 3$ , 6.0 pixels para o tamanho  $4 \times 4$  e, por fim, 6.7 pixels para o tamanho  $5 \times 5$ . Nas médias verificamos que a configuração de  $4 \times 4$  é a melhor com 6.6 pixels, seguido pelos 7.0 pixels da de  $5 \times 5$ , 7.2 pixels do tamanho  $3 \times 3$  e, por fim, os 7.9 pixels da de bloco  $2 \times 2$ . Neste critério, apesar da mediana ser mais alta, o bloco  $4 \times 4$  leva ligeira vantagem pela sua média e pela pequena diferença entre percentis. No critério do tamanho do bloco do HOG foram seleccionados duas opções o  $3 \times 3$  e o  $4 \times 4$  para utilização na experiência final, pelos seus resultados serem muito próximos incluindo um empate ao nível da *accuracy* para erro do operador igual a 6.

### 5.2.4 Tipo de Normalização do HOG

O segundo factor explorado do HOG foi a normalização utilizada e para a qual existem 5 possibilidades e cuja comparação ao nível do gráfico da *accuracy* em função do raio

## Resultados

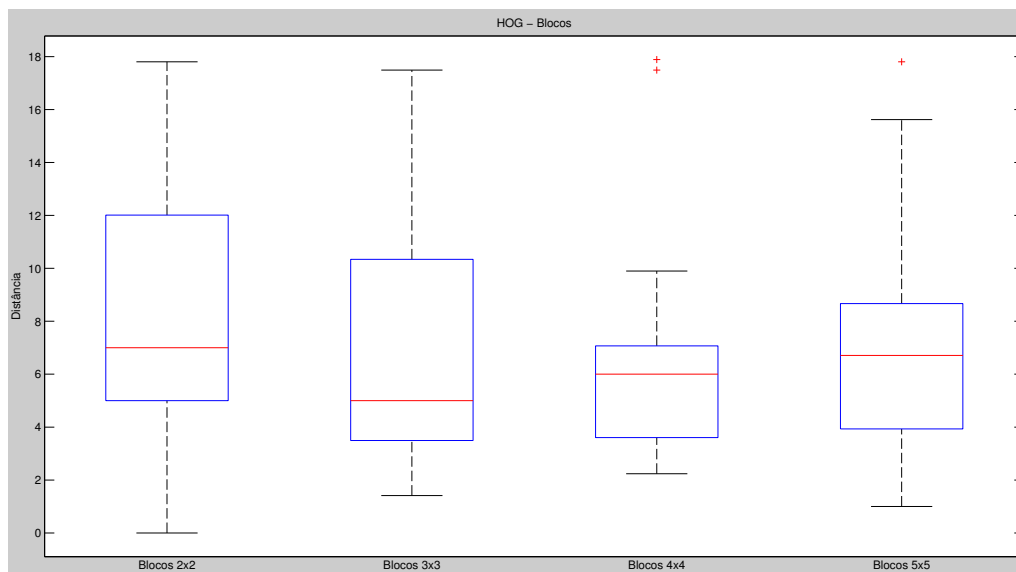


Figura 5.24: Diagramas de extremos e quartis da distância para a comparação entre o tamanho do bloco do HOG.

do erro do operador se encontra na figura 5.25. Numa análise da figura 5.25 verificamos

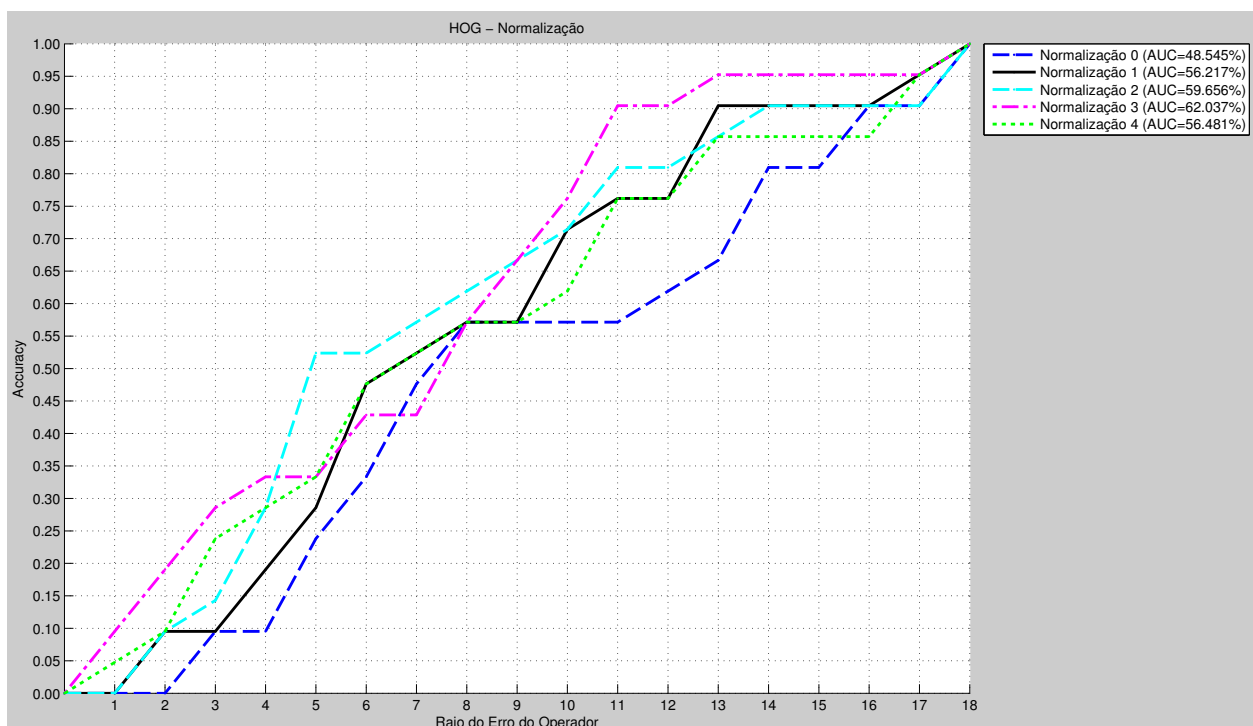


Figura 5.25: Gráficos da *accuracy* em função do raio do erro do operador para a comparação entre a normalização utilizada no HOG.

que existe algum equilíbrio entre praticamente todas as normalizações com exceção da

## Resultados

não utilização de normalização (0), com AUC de 48.5%. As restantes encontram-se entre os 56.2% da normalização local (1) e os 62% da normalização global utilizando 11-sqrt (3). A nível da *accuracy* verifica-se uma superioridade da normalização 2, global utilizando o máximo, nomeadamente nos valores de erro mais próximos da referência para o erro do operador. Atinge um valor de 52.4% para erro igual a 6 pixels, comparado com os 47.6% alcançados pelas duas segundas melhores normalizações, a normalização 4, global utilizando o l2-norm e a 1, normalização local. Passando para a comparação relativamente à distância, cujo gráfico está presente em 5.26. Observando a imagem 5.26

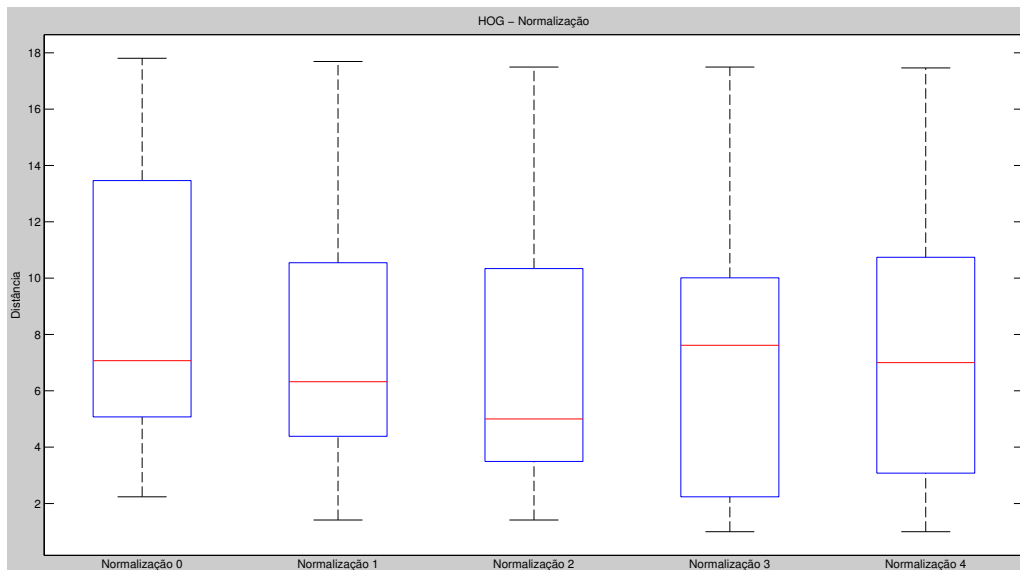


Figura 5.26: Diagramas de extremos e quartis da distância para a comparação entre a normalização utilizada no HOG.

volta-se a verificar um certo equilíbrio entre as várias possibilidades. A normalização 2, global utilizando o máximo, é a que obtém uma mediana mais baixa, com 5.0 pixels, e alcançando uma média de 7.2 pixels. As restantes normalizações obtém valores próximos com mediana de 7.1 pixels para a normalização 0, sem normalização, 6.3 pixels para a normalização local (1), 7.6 pixels para a normalização 3, global com a utilização de 11-sqrt, e 7.0 pixels para a normalização global utilizando l2-norm (4). Analisando a média, apenas a normalização global utilizando 11-sqrt (3) atinge um valor mais baixo, 6.9, tendo no entanto pior mediana e uma maior distância entre quartis. Conjugando os diversos critérios verificamos que a normalização 2, global com a utilização do máximo, alcança os melhores resultados, sendo a seleccionada para a experiência final.

### 5.2.5 Número de Orientações do HOG

O último parâmetro a ser testado para verificar qual o seu melhor valor é o número de orientações (*bins*), cujos resultados estão presentes na figura 5.27. Analisando os gráficos

## Resultados

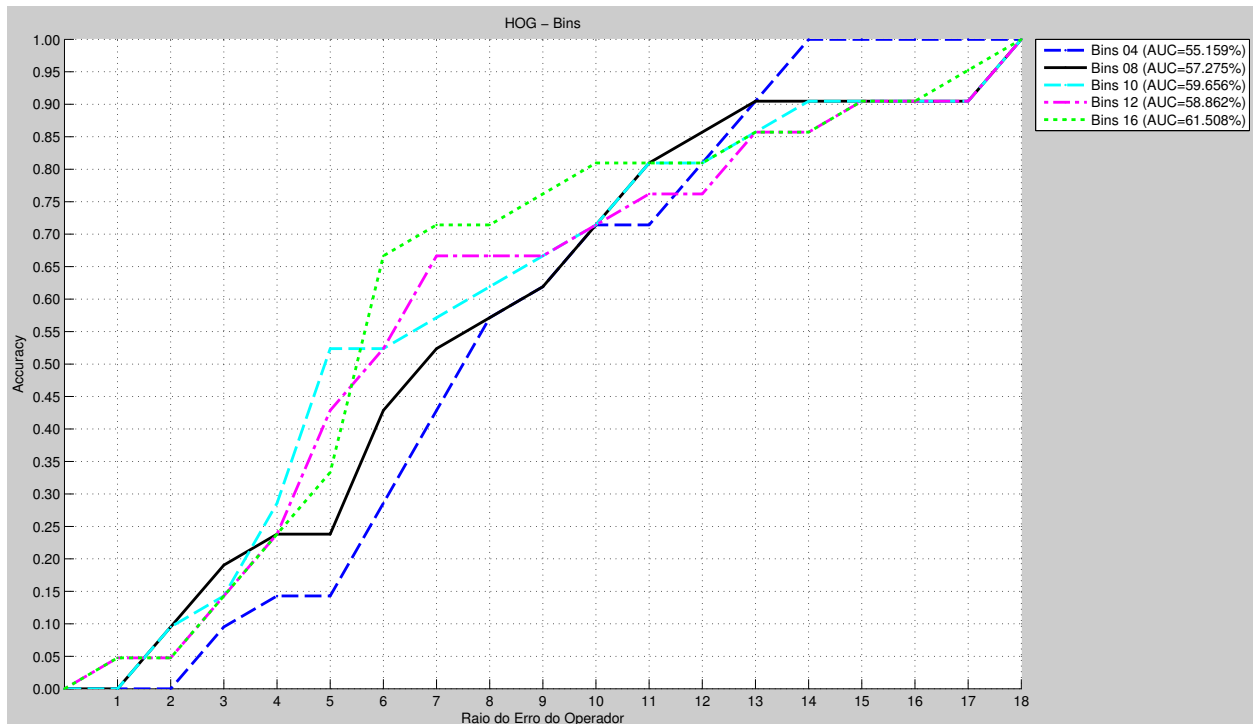


Figura 5.27: Gráficos da *accuracy* em função do raio do erro do operador para a comparação entre o número de orientações (*bins*) do HOG.

verificamos uma grande igualdade entre os diferentes número de orientações. O melhor valor de percentagem de área é obtido com 16 *bins*, com 61.5%. Os restantes obtêm valores relativamente próximos com 55.2% para 4 *bins*, 57.3% para 8 *bins*, 59.4% para 10 *bins* e 58.9% para 12. Ao nível da *accuracy* para o valor de erro de 6 pixels a vantagem da configuração de 16 *bins* é superior com 66.7%, comparados com os 52.4% alcançados pelas configurações de 10 e 12 *bins*. Para terminar a comparação deste parâmetros iremos analisar a métrica da distância, presente na imagem 5.28. Nesta medida já existe uma maior diferença entre as várias configurações, em que o número de orientações de 10, 12 e 16 levam vantagem. Apresentam medianas de 5.0 pixels para os 10 *bins*, 5.8 pixels para os 12 *bins* e 5.8 pixels para os 16 *bins*. Em termos da média a opção dos 16 *bins* leva vantagem com 7.1 pixels em relação aos 7.2 pixels dos 10 *bins* e os 7.5 pixels da opção de 12 *bins*. Neste parâmetro a opção seleccionada para a experiência final foi a utilização de 16 *bins* (número de orientações) devido a ter evidenciado a sua superioridade em quase todos os critérios estabelecidos para a comparação.

## Resultados

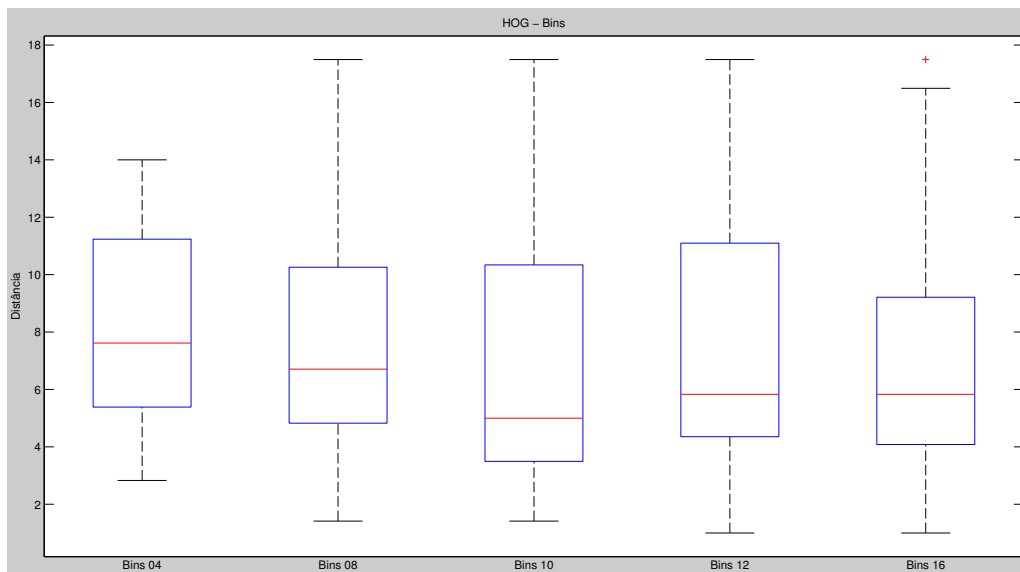


Figura 5.28: Diagramas de extremos e quartis da distância para a comparação entre o número de orientações (*bins*) do HOG.

### 5.2.6 Discussão

No final desta experiência verificamos que existem duas configurações que atingiram bons resultados. Uma primeira com um bloco de  $3 \times 3$  e uma segunda com bloco de  $4 \times 4$ . Em comum partilham o tipo de normalização que é global recorrendo ao máximo (tipo 2), um número de orientações igual a 16, um bloco de imagem de  $40 \times 40$  e a não utilização de sobreposição. Em termos de impacto verifica-se que todos os parâmetros têm algum impacto nos resultados e em termos de parâmetros independentes do descritor o com maior impacto é o tamanho do bloco da imagem. Em termos dos parâmetros do HOG aqui analisados destaca-se a influência do tamanho do bloco e do tipo de normalização.

### 5.3 Experiência 3: Avaliação do HOG com parâmetros otimizados e do pós-processamento

Nesta experiência final vão ser comparadas três diferentes configurações, a melhor obtida na primeira experiência e as duas obtidas na experiência anterior que consistem na utilização de um bloco de imagem de  $40 \times 40$  e na não utilização de sobreposição entre blocos em termos de parâmetros independentes do descritor. Ao nível dos parâmetros para o HOG, seleccionou-se um bloco de  $3 \times 3$  ou  $4 \times 4$ , a normalização global recorrendo ao máximo (tipo 2) e um número de orientações igual a 16. Na figura 5.29 apresenta-se a comparação feita recorrendo ao critério da AUC do gráfico da *accuracy* em função do raio do erro do operador. Analisando os resultados presentes no gráfico verificamos que

## Resultados

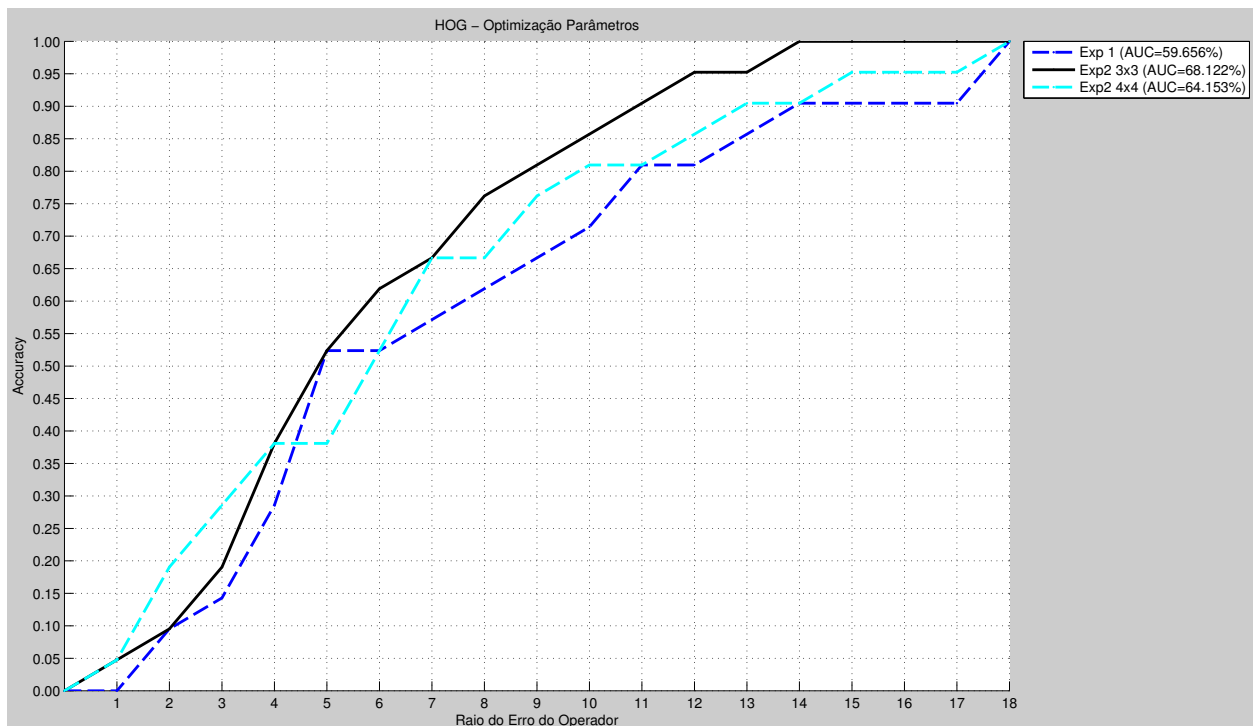


Figura 5.29: Gráficos da *accuracy* em função do raio do erro do operador para a comparação entre os melhores resultados obtidos.

a configuração da experiência 2 com bloco  $3 \times 3$  é a que obtém os melhores resultados. A percentagem de área sob a curva atinge os 68.1%, superiores aos 64.2% obtidos pela configuração com um bloco do HOG de  $4 \times 4$  e aos 59.7% obtidos pela configuração da primeira experiência. Os resultados da experiência 1 aqui apresentados são superiores aos apresentados anteriormente devido a ter sido feito uma suavização do *output* do classificador, em que o valor de sigma foi otimizado. Esta situação também foi utilizada nos outros dois cenários e posteriormente neste capítulo será apresentado um exemplo desse processo de optimização. Em termos de *accuracy* o melhor valor atingido são os 61.9% obtidos pela configuração com bloco  $3 \times 3$ , em comparação com os 52.4% obtidos pelas outras duas hipóteses. Analisando de seguida a distância, cujo gráfico comparativo se encontra presente na figura 5.30. Numa leitura a estes resultados verificamos que a configuração de bloco  $3 \times 3$  obtém novamente os melhores resultados. Apresenta uma mediana de 5.0 pixels e uma média de 5.9 pixels que pela primeira vez é inferior aos 6 pixels. As outras duas configurações apresentam medianas de 5.7 pixels e 5.0 pixels, para a de bloco  $4 \times 4$  e a obtida da primeira experiência, respectivamente. Em termos da média a da segunda experiência com bloco  $4 \times 4$  alcança 6.5 pixels e a outra 7.2 pixels. É importante salientar a robustez obtida pela melhor configuração, com um máximo de distância de 13 pixels, um mínimo de 1 pixel e apresentando um desvio-padrão de 3.3 pixels. A última métrica explorada nesta experiência é o ranking e na figura 5.31 é apresentado o

## Resultados

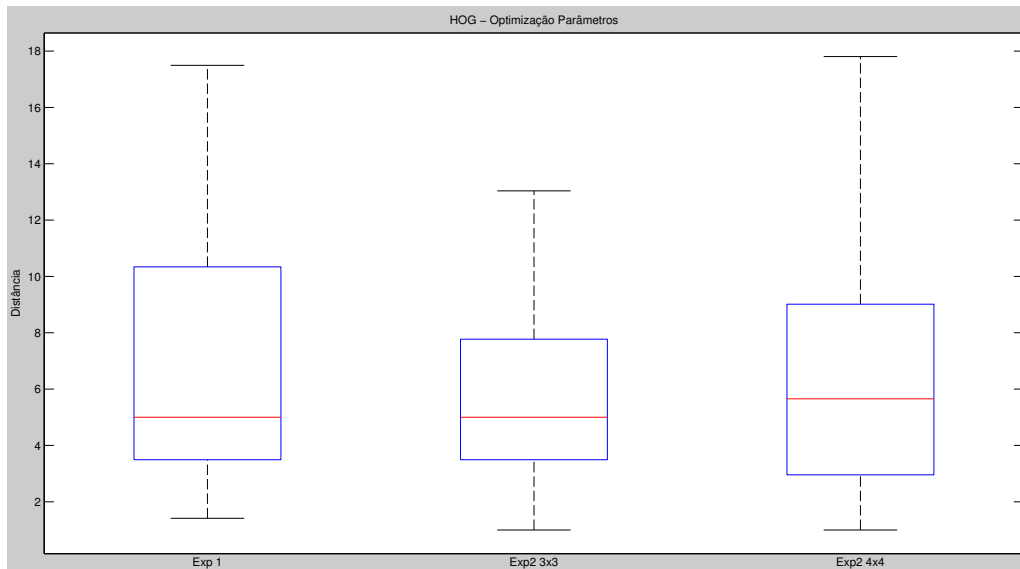


Figura 5.30: Diagramas de extremos e quartis da distância para a comparação entre os melhores resultados obtidos.

histograma para a melhor configuração (bloco de  $3 \times 3$ ). Analisando o histograma pre-

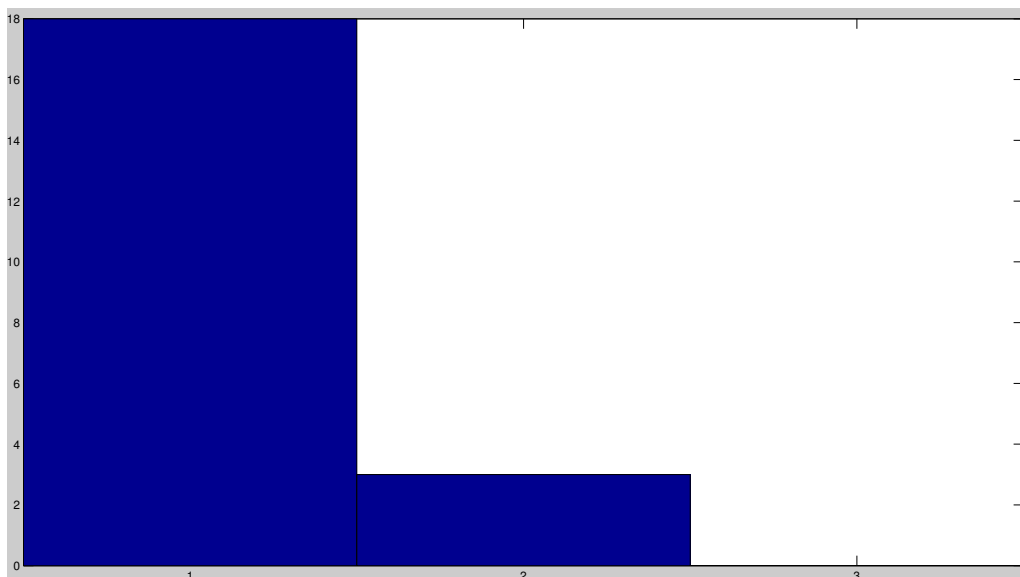


Figura 5.31: Histograma do Ranking para a melhor configuração alcançada.

sente em 5.31 verificamos que na maioria dos casos, 85.7%, o extremo mais próximo de pedículo tem o melhor ranking, logo o melhor resultado (*score*) dado pela SVM.

Nesta última experiência foi também explorado o efeito da suavização do *output* do classificador nos resultados obtidos. Foram testados diferentes valores de sigma utilizado no filtro gaussiano para a suavização, sob dois parâmetros: a *accuracy* para raio do erro do

## Resultados

operador de 6 pixels e a distância às coordenadas de referência do pedículo. O critério utilizado em primeiro lugar foi seleccionar o sigma para o qual a *accuracy* seja a melhor e no caso de existir um empate em relação à *accuracy* verificar a média da distância como factor de desempate. Nas imagens 5.32 e 5.33 apresenta-se os valores obtidos na optimização do sigma, em relação aos dois factores de decisão, a *accuracy* em 5.32 e a distância em 5.33, utilizada na obtenção dos resultados da melhor configuração. Observando as

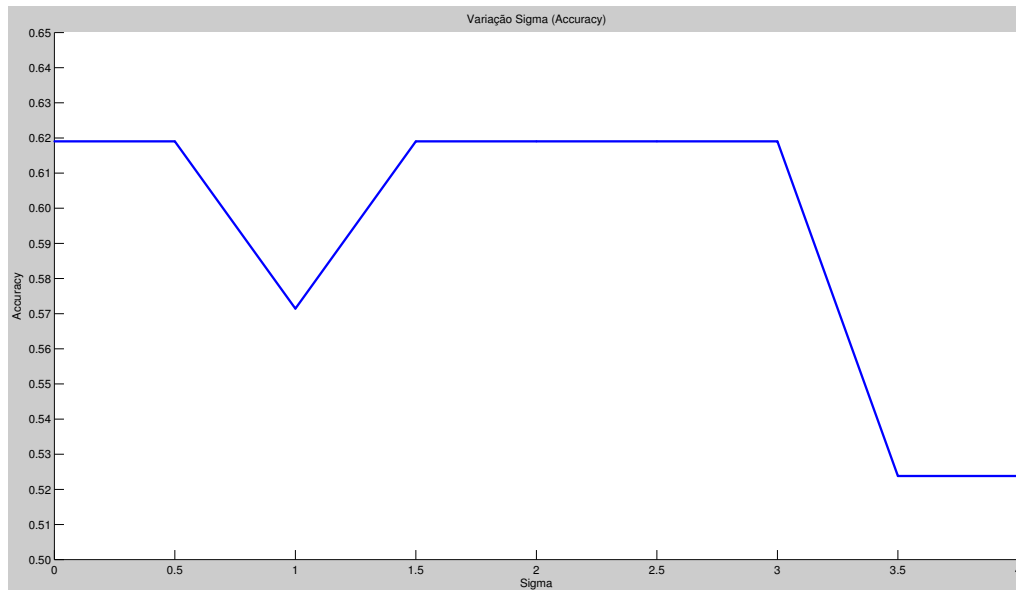


Figura 5.32: *Accuracy* em função do sigma no filtro de pós-processamento.

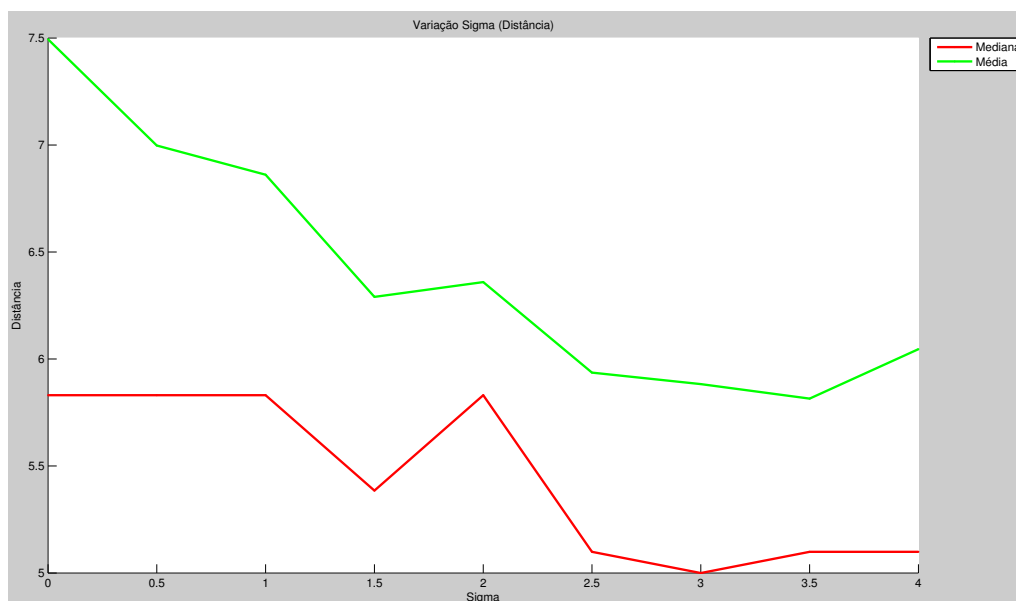


Figura 5.33: Média e Mediana da distância em função do sigma no filtro de pós-processamento.

imagens 5.32 e 5.33 verificamos que o sigma que obtém melhores resultados segundo o critério utilizado é o com valor 3. Em relação ao valor da *accuracy* existe um empate entre os valores 0.5, 1.5, 2, 2.5 e 3 de sigma e com a não aplicação de suavização e foi necessário proceder a uma análise da distância para estes valores de sigma. Neste critério verificamos que a média mais baixa é de 5.9 pixels para valor 3 de sigma. A mediana mais baixa é também neste valor de sigma atingindo os 5 pixels.

### 5.3.1 Discussão

Numa análise do efeito dos parâmetros do HOG e dos parâmetros independentes do descritor nos resultados conclui-se que têm um impacto positivo e demonstra a importância de basear a escolha dos parâmetros em testes exaustivos.

Em termos dos efeitos da suavização do resultado do classificador, fazendo uma comparação entre a não aplicação de suavização, sigma igual a 0, com o melhor sigma com valor 3, verifica-se que a técnica de pós-processamento melhora os resultados, principalmente a nível da distância, com uma redução na ordem de 1 pixel na mediana e de 1.5 pixels na média. Em termos de discussão é também de salientar que estes resultados estão sobreajustados ao conjunto de teste. Os parâmetros óptimos podem variar ao fazer variar o conjunto de teste e idealmente isto seria feito com uma validação cruzada *k*-desdobrável, mas que pelo facto de o tempo de execução ser demasiado elevado, tal método não foi utilizado. Os únicos parâmetros ajustados com validação cruzada foram os do classificador e apenas sobre o conjunto de treino.

## 5.4 Análise Temporal

Numa análise realizada em relação ao desempenho do método verificou-se que o processo de varrimento da imagem, incluindo o cálculo do descritor para cada pixel, a classificação de cada um e a obtenção do candidato com melhor resultado (*score*) demora uma média de 34,99 segundos por cada imagem. Neste tempo não está incluído o tempo necessário ao treino do classificador que é feito apenas uma vez e pode ser feito *a priori*. Os testes foram realizados num computador com um processador *Intel Core Duo 6400*, com 2.13 GHz com 2 GB de memória e o sistema operativo utilizado é o Windows Vista Ultimate SP2 de 32 bits. A fase mais demorada é a fase de varrimento e cálculo do descritor e ocupa cerca de 60% do total. A fase de classificação ocupa cerca de 35%, deixando menos de 5% para as restantes fases.

## 5.5 Sumário

Ao longo deste capítulo foram apresentados os resultados das três experiências efectuadas no âmbito deste trabalho. Na primeira foram comparadas as diversas combinações possíveis entre classificador e descritor, variante entre elas outros dois factores, a presença de ruído e da presença da rotação da vértebra. Nesta experiência verificou-se que a melhor configuração foi a utilização do HOG com a SVM e a aplicação de filtro de ruído e a eliminação da rotação nas vértebras. Na segunda experiência compararam-se os diferentes parâmetros de forma independente verificando qual o melhor valor para cada um deles. Na experiência final foram apresentados os resultados do melhor conjunto de parâmetros encontrado, numa análise com maior profundidade, recorrendo a uma nova métrica e a uma suavização do *output* do classificador. No final do capítulo foi realizada uma análise relativo ao desempenho obtido pelo método de detecção dos pedículos.

## Capítulo 6

# Conclusão

Neste trabalho foi proposto um método para a detecção de estruturas anatómicas em imagens médicas, em particular para a detecção dos pedículos em radiografias planares da coluna vertebral. O método proposto é baseado na utilização de descritores de imagem e de classificadores supervisionados e possui três fases: o pré-processamento, o treino e a detecção das estruturas anatómicas. Foram explorados diferentes tipos quer de descritores de imagem, como por exemplo os baseados na distribuição do gradiente, quer de classificadores, como a SVM ou a RNA. Além de determinar qual a combinação entre descritor e classificador que obtinha melhores resultados foi também estudado o impacto de técnicas de pré-processamento e de pós-processamento na detecção, principalmente a eliminação de ruído e o factor rotação. O trabalho é realizado tendo por base as coordenadas de referência do pedículo identificadas por um perito.

Os resultados obtidos pelo método proposto para a detecção automática dos pedículos foram superiores às expectativas, na medida em que os melhores resultados obtidos alcançaram uma distância média, em relação às coordenadas de referência dadas pelo técnico, com um valor dentro dos limites esperados para o erro que o operador comete que é cerca de 1mm [KCLL07], resultando num erro esperado de cerca de 7,5 pixeis considerando a sua distribuição como normal. Conclui-se que a melhor combinação para este problema consiste na utilização do descritor HOG, do classificador SVM, da utilização da eliminação da rotação e da eliminação de ruído. Conclui-se, numa observação dos resultados, que os descritores baseados na distribuição do gradiente, como o HOG, obtiveram os melhores resultados, tal como previa a hipótese levantada no início desta dissertação. De realçar que o descritor HOG obteve resultados superiores ao momentos estatísticos que foram utilizados por Duong *et al.* no seu trabalho de detecção automática das curvas da coluna vertebral e que o segundo melhor descritor, o SURF, tem um princípio próximo ao do HOG. Nos classificadores verificou-se uma superioridade da SVM em relação aos restantes, embora a escolha do classificador se tenha revelado um aspecto menos crítico que a escolha do descritor. A optimização dos parâmetros da detecção tem um impacto

positivo nos resultados, mostrando a importância de basear a escolha dos parâmetros em testes exaustivos. Em termos de parâmetros independentes do descritor o tamanho do bloco da imagem é o que tem o maior impacto, sendo que o nível de sobreposição tem impacto menor. Nos parâmetros do HOG todos têm algum impacto, sendo o tamanho do bloco e o tipo de normalização os que mais influenciam os resultados. Em termos de remoção de ruído verificou-se que na maioria dos casos traduziu-se numa melhoria dos resultados, sendo que em algumas situações piorou ligeiramente os resultados obtidos. Uma razão para a remoção de ruído não melhorar sempre os resultados pode estar relacionada com a não optimização dos valores do filtro de eliminação de ruído utilizado, o BM3D [DFKE07]. Em relação ao problema da rotação das estruturas anatómicas, a sua eliminação melhora os resultados, principalmente ao nível do HOG, levando no entanto a piores resultados noutras situações, como o SURF. Um caso a notar é o piorar dos resultados em descritores que seriam invariantes à rotação como os momentos invariantes, facto que se manteve constante nas diferentes experiências feitas ao nível do algoritmo de rotação (interpolação bilinear ou bicúbica), possivelmente devido à introdução de ruído na aplicação da rotação às imagens. No caso de não ser possível a eliminação da rotação, o HOG continua a ser o melhor descritor, embora a diferença para o SURF e para os momentos invariantes não seja elevada, permitindo a utilização destes descritores sem grande prejuízo para os resultados. A hipótese levantada no início do trabalho de que o pré-processamento e o pós-processamento seriam factores importantes e que levavam a uma melhoria dos resultados verificou-se. Em termos de pós-processamento foi efectuada uma suavização da saída do classificador com um filtro gaussiano. Este tipo de análise tem um impacto positivo nos resultados pelo facto de eliminar possível ruído da classificação, permitindo uma melhor análise dos extremos. Em termos de desempenho a nível temporal concluímos que apesar de não ter sido dada ênfase a esta questão os tempos obtidos são promissores, cerca de 35 segundos por imagem, e será alvo de futuras melhorias. Outro factor a ter em conta é a possibilidade do método proposto para a detecção dos pedículos ser generalizável a outras estruturas anatómicas, como por exemplo os centros das plataformas vertebrais e seus limites.

### 6.1 Trabalho Futuro

Em termos de trabalho futuro, existe a possibilidade de se integrar nas experiências novos descritores de imagem, como o SSIM descrito no capítulo 2, com o objectivo de analisar o seu desempenho neste problema. Outro aspecto a explorar seria recorrer a novos tipos de classificador como os que utilizam *boosting*, como é o caso do AdaBoost ou a utilização do classificador K-Vizinhos mais próximos, sendo no entanto necessário propor novas formas de avaliar os resultados, dado que este classificador não permite o

cálculo de uma probabilidade associada de forma directa. Ainda em termos de classificador poder-se-ia aprofundar a utilização das RNA, utilizando outros tipos de estruturas de redes, explorando novos algoritmo de treino e fazendo mais experiências a nível de parâmetros, como o valor de aprendizagem, para verificar a capacidade de este classificador obter melhores resultados. Nas experiências efectuadas poderiam ser testados novos *datasets* compilados a partir de imagens radiológicas de pacientes com outras patologias ao nível da coluna vertebral que não a escoliose. Além disso, poderiam ser explorados outras vértebras de diferentes níveis que não a L1, a L2 ou a L3 (utilizadas neste trabalho) e outros pontos do pedículo para verificar se os resultados obtidos são consistentes. Outro aspecto que poderia ser mais aprofundado diz respeito ao filtro de eliminação de ruído utilizado, o BM3D [DFKE07]. Poderiam ser feitas experiências com os vários parâmetros com o objectivo de os otimizar para o problema da detecção dos pedículos.

Em termos de novas linhas de investigação seria interessante expandir o método de modo a analisar radiografias laterais e combinar estes resultados com os alcançados na detecção em radiografias frontais de modo a obter uma melhor posição para as extremidades do pedículo. A utilização de métodos estatísticos que permitam quantificar a probabilidade de um pedículo estar localizado em determinado pixel baseado no *input* de um utilizador poderia ser também um factor que pode permitir melhorar resultados, anulando área de pesquisa da análise devido à sua baixa probabilidade. Outra possibilidade seria um novo método de detecção, que também seria uma extensão ao proposto, baseado na detecção das duas extremidades, superior e inferior, do pedículo em simultâneo, treinando um classificador para cada extremidade, e utilizar o ângulo de rotação que pode ser obtido com o método de reconstrução da coluna [MBBT09] [MBB<sup>+</sup>11] como forma de verificar o desempenho da detecção. A nível da avaliação dos resultados novas métricas poderiam ser exploradas, fazendo a análise dos máximos locais e não apenas dos globais. Um exemplo de uma métrica seria uma que verificasse a percentagem de imagens nas quais era detectado um máximo local próximo das coordenadas de referência, ao contrário da métrica utilizada que apenas verifica o máximo global. Este medida analisa a capacidade de detecção de pontos interessantes que apesar de não serem o máximo global poderiam ser considerados como pedículos, aumentando o número de candidatos. A nível da performance temporal o estrangulamento encontra-se no tempo que demora o varrimento e o cálculo do descritor e é passível de melhoria através de duas formas, uma efectuando uma implementação do descritor numa linguagem compilada, como o C/C++, ou reduzindo a densidade da área de varrimento, como por exemplo analisar apenas de dois em dois pixels. Por fim, está em vista a integração deste método no sistema de reconstrução 3D da coluna vertebral [MBBT09] [MBB<sup>+</sup>11], tendo como objectivo melhorar as reconstruções 3D a partir dos resultados da detecção.

## Conclusão

# Referências

- [ADD<sup>+</sup>95] CE Aubin, JL Descrimes, J. Dansereau, W. Skalli, F. Lavaste e H. Labelle. Geometrical modeling of the spine and the thorax for the biomechanical analysis of scoliotic deformities using the finite element method. In *Annales de chirurgie*, volume 49, page 749, 1995.
- [AKJ04] R. Akbani, S. Kwek e N. Japkowicz. Applying support vector machines to imbalanced datasets. *Machine Learning: ECML 2004*, pages 39–50, 2004.
- [ATRB95] AP Ashbrook, NA Thacker, PI Rockett e CI Brown. Robust recognition of scaled shapes using pairwise geometric histograms. In *Proc. BMVC*, pages 503–512. Citeseer, 1995.
- [Bau00] A. Baumberg. Reliable feature matching across widely separated views. In *cvpr*, page 1774. Published by the IEEE Computer Society, 2000.
- [BMP02] S. Belongie, J. Malik e J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 509–522, 2002.
- [Bor09] J.S. Borges. Classificadores e máquinas de suporte vectorial em imagens multi-espectrais, 2009.
- [BPM04] G.E. Batista, R.C. Prati e M.C. Monard. A study of the behavior of several methods for balancing machine learning training data. *ACM SIGKDD Explorations Newsletter*, 6(1):20–29, 2004.
- [Bra97] A.P. Bradley. The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern Recognition*, 30(7):1145–1159, 1997.
- [BTVG06] H. Bay, T. Tuytelaars e L. Van Gool. Surf: Speeded up robust features. *Computer Vision–ECCV 2006*, pages 404–417, 2006.
- [CBHK02] N.V. Chawla, K.W. Bowyer, L.O. Hall e W.P. Kegelmeyer. SMOTE: synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16(1):321–357, 2002.
- [CL01a] C.C. Chang e C.J. Lin. Libsvm: a library for support vector machines. 2001.
- [CL01b] Chih-Chung Chang e Chih-Jen Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.

## REFERÊNCIAS

- [CPE02] VN Cassar-Pullicino e SM Eisenstein. Imaging in scoliosis: what, why and how? *Clinical radiology*, 57(7):543–562, 2002.
- [CTL<sup>+</sup>10] J. Chen, J. Tian, N. Lee, J. Zheng, R.T. Smith e A.F. Laine. A Partial Intensity Invariant Feature Descriptor for Multimodal Retinal Image Registration. *Biomedical Engineering, IEEE Transactions on*, 57(7):1707–1718, 2010.
- [CV01] T.F. Chan e L.A. Vese. Active contours without edges. *Image Processing, IEEE Transactions on*, 10(2):266–277, 2001.
- [DA00] J.S. Duncan e N. Ayache. Medical image analysis: Progress over two decades and the challenges ahead. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(1):85–106, 2000.
- [DBH10] H. Demuth, M. Beale e M. Hagan. Neural network toolbox. *Neural Network Toolbox User's Guide*, 2010.
- [DCL09] L. Duong, F. Cheriet e H. Labelle. Automatic Detection of Scoliotic Curves in Posteroanterior Radiographs. *Biomedical Engineering, IEEE Transactions on*, 57(5):1143–1151, 2009.
- [DDCC07] V. Doré, L. Duong, F. Cheriet e M. Cheriet. Towards segmentation of pedicles on posteroanterior x-ray views of scoliotic patients. *Image Analysis and Recognition*, pages 1028–1039, 2007.
- [DFKE07] K. Dabov, A. Foi, V. Katkovnik e K. Egiazarian. Image denoising by sparse 3-D transform-domain collaborative filtering. *Image Processing, IEEE Transactions on*, 16(8):2080–2095, 2007.
- [DML<sup>+</sup>03] R. Dumas, D. Mitton, S. Laporte, J. Dubousset, JP Steib, F. Lavaste e W. Skalli. Explicit calibration method and specific device designed for stereoradiography. *Journal of biomechanics*, 36(6):827–834, 2003.
- [DP97] P. Domingos e M. Pazzani. On the optimality of the simple Bayesian classifier under zero-one loss. *Machine learning*, 29(2):103–130, 1997.
- [DSG02] Harris Drucker, Behzad Shahrari e David C. Gibbon. Support vector machines: relevance feedback and information retrieval. *Information Processing & Management*, 38(3):305 – 323, 2002.
- [DT05] N. Dalal e B. Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 886–893. IEEE, 2005.
- [Eva09] Christopher Evans. Notes on the opensurf library. Technical Report CSTR-09-001, University of Bristol, January 2009.
- [FA91] W.T. Freeman e E.H. Adelson. The design and use of steerable filters. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 13(9):891–906, 1991.
- [Gab46] D. Gabor. Theory of communication: J. Inst. *Electr. Eng*, 93:429–457, 1946.

## REFERÊNCIAS

- [GD04] A. Greenspan e L.P. Duprey. *Orthopedic imaging: a practical approach*. Lippincott Williams & Wilkins, 2004.
- [GWE03] R.C. Gonzalez, R.E. Woods e S.L. Eddins. *Digital image processing using MATLAB*. Prentice Hall, 2003.
- [GWE04] R.C. Gonzalez, R.E. Woods e S.L. Eddins. *Digital image processing*. Prentice Hall Upper Saddle River, NJ, 2004.
- [Hay94] S. Haykin. *Neural networks: a comprehensive foundation*. Prentice Hall PTR Upper Saddle River, NJ, USA, 1994.
- [HDO<sup>+</sup>98] M.A. Hearst, ST Dumais, E. Osman, J. Platt e B. Scholkopf. Support vector machines. *Intelligent Systems and their Applications, IEEE*, 13(4):18–28, 1998.
- [Her02] R. Herbrich. *Learning kernel classifiers: theory and algorithms*. The MIT Press, 2002.
- [HL02] C.W. Hsu e C.J. Lin. A comparison of methods for multiclass support vector machines. *Neural Networks, IEEE Transactions on*, 13(2):415–425, 2002.
- [HS88] C. Harris e M. Stephens. A combined corner and edge detector. In *Alvey vision conference*, pages 147–152. Manchester, UK, 1988.
- [Hu62] M.K. Hu. Visual pattern recognition by moment invariants. *Information Theory, IRE Transactions on*, 8(2):179–187, 1962.
- [Jap00] N. Japkowicz. The class imbalance problem: Significance and strategies. In *Proceedings of the 2000 International Conference on Artificial Intelligence (ICAI'2000)*, volume 1, pages 111–117. Citeseer, 2000.
- [JH99] A.E. Johnson e M. Hebert. Using spin images for efficient object recognition in cluttered 3D scenes. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 21(5):433–449, 1999.
- [JMM96] A.K. Jain, J. Mao e K.M. Mohiuddin. Artificial neural networks: A tutorial. *Computer*, 29(3):31–44, 1996.
- [KCLL07] S. Kadoury, F. Cherié, C. Laporte e H. Labelle. A versatile 3d reconstruction system of the spine and pelvis for clinical assessment of spinal deformities. *Medical and Biological Engineering and Computing*, 45(6):591–602, 2007.
- [KHM97] M. Kubat, R. Holte e S. Matwin. Learning when negative examples abound. *Machine Learning: ECML-97*, pages 146–153, 1997.
- [KJPK02] K.I. Kim, K. Jung, S.H. Park e H.J. Kim. Support vector machines for texture classification. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(11):1542–1550, 2002.
- [KKK04] V. Kyrki, J.K. Kamarainen e H. Kalviainen. Simple Gabor feature space for invariant object recognition. *Pattern Recognition Letters*, 25(3):311–318, 2004.

## REFERÊNCIAS

- [KM97] M. Kubat e S. Matwin. Addressing the curse of imbalanced training sets: one-sided selection. In *MACHINE LEARNING-INTERNATIONAL WORKSHOP THEN CONFERENCE-*, pages 179–186. Citeseer, 1997.
- [Koh06] Teuvo Kohonen. Self-organizing neural projections. *Neural Networks*, 19(6-7):723 – 733, 2006. Advances in Self Organising Maps - WSOM'05.
- [Kro10] Dirk-Jan Kroon. Opensurf library for matlab. <http://www.mathworks.com/matlabcentral/fileexchange/28300-opensurf-including-image-warp>, 2010.
- [LdC08] A.C. Lorena e A.C. de Carvalho. Uma introdução às support vector machines. *Revista de Informática Teórica e Aplicada*, 14(2):43–67, 2008.
- [Low04] D.G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.
- [LSP05] S. Lazebnik, C. Schmid e J. Ponce. A sparse texture representation using local affine regions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1265–1278, 2005.
- [MAGB10] D. Marín, A. Aquino, M. Gegúndez e J. Bravo. A New Supervised Method for Blood Vessel Segmentation in Retinal Images by Using Gray-Level and Moment Invariants-Based Features. *IEEE transactions on medical imaging*, 2010.
- [MAT10] MATLAB. *version 7.10.0 (R2010a)*. The MathWorks Inc., Natick, Massachusetts, 2010.
- [MBB<sup>+</sup>11] D.C. Moura, J. Boisvert, J.G. Barbosa, H. Labelle e J.M.R.S. Tavares. Fast 3D reconstruction of the spine from biplanar radiographs using a deformable articulated model. *Medical Engineering & Physics*, (in press) 2011.
- [MBBT09] D. Moura, J. Boisvert, J. Barbosa e J. Tavares. Fast 3D Reconstruction of the Spine Using User-Defined Splines and a Statistical Articulated Model. *Advances in Visual Computing*, pages 586–595, 2009.
- [Mit97] Tom M. Mitchell. *Machine Learning*. McGraw-Hill, New York, 1997.
- [MS05] K Mikołajczyk e C Schmid. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1615–1630, October 2005.
- [Nob04] W.S. Noble. Support vector machine applications in computational biology. *Kernel methods in computational biology*, pages 71–92, 2004.
- [OPM02] T. Ojala, M. Pietikainen e T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 971–987, 2002.
- [PL04] F.J.G. Pitanga e I. Lessa. Sensibilidade e especificidade do índice de concidência como discriminador do risco coronariano de adultos em salvador, brasil. *Revista Brasileira de Epidemiologia*, 7(3):259–269, 2004.

## REFERÊNCIAS

- [PP00] C. Papageorgiou e T. Poggio. A trainable system for object detection. *International Journal of Computer Vision*, 38(1):15–33, 2000.
- [Pra01] W.K. Pratt. *Digital image processing*. John Wiley & Sons, Inc, NY, 2001.
- [Ris01] I. Rish. An empirical study of the naive Bayes classifier. In *IJCAI 2001 Workshop on Empirical Methods in Artificial Intelligence*, pages 41–46, 2001.
- [Ros58] F. Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386 – 408, 1958.
- [SI07] Eli Shechtman e Michal Irani. Matching local self-similarities across images and videos. In *IEEE Conference on Computer Vision and Pattern Recognition 2007 (CVPR'07)*, June 2007.
- [SS02] B. Scholkopf e A.J. Smola. *Learning with kernels: support vector machines, regularization, optimization, and beyond*. the MIT Press, 2002.
- [Sta03] C. Staelin. Parameter selection for support vector machines. *Hewlett-Packard Company, Tech. Rep. HPL-2002-354R1*, 2003.
- [Sto94] I.A.F. Stokes. Three-dimensional terminology of spinal deformity: a report presented to the Scoliosis Research Society by the Scoliosis Research Society Working Group on 3-D terminology of spinal deformity. *Spine*, 19(2):236, 1994.
- [SZ02] F. Schaffalitzky e A. Zisserman. Multi-view matching for unordered image sets. *Computer Vision—ECCV 2002*, pages 414–431, 2002.
- [TLF08] Engin Tola, Vincent Lepetit e Pascal Fua. A fast local descriptor for dense matching. In *Conference on Computer Vision and Pattern Recognition*, Alaska, USA, 2008.
- [TLF10] E. Tola, V. Lepetit e P. Fua. Daisy: an Efficient Dense Descriptor Applied to Wide Baseline Stereo. volume 32, pages 815–830, May 2010.
- [Tol10] Engin Tola. Daisy: Local descriptor for dense wide-baseline stereo matching. <http://cvlab.epfl.ch/~tola/daisy.html>, 2010.
- [TZCK09] Y. Tang, Y.Q. Zhang, N.V. Chawla e S. Krasser. Svms modeling for highly imbalanced classification. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 39(1):281–288, 2009.
- [Vap00] V.N. Vapnik. *The nature of statistical learning theory*. Springer Verlag, 2000.
- [VCC99] K. Veropoulos, C. Campbell e N. Cristianini. Controlling the sensitivity of support vector machines. In *Proceedings of the international joint conference on AI*. Citeseer, 1999.
- [VGMU96] L. Van Gool, T. Moons e D. Ungureanu. Affine/photometric invariants for planar intensity patterns. *Computer Vision—ECCV'96*, pages 642–651, 1996.

## REFERÊNCIAS

- [VR79] CJ Van Rijsbergen. Information retrieval. *Butterworths, London*, 1979.
- [WC03] G. Wu e E.Y. Chang. Class-boundary alignment for imbalanced dataset learning. In *Proceedings of the ICML*, volume 3. Citeseer, 2003.
- [ZW94] R. Zabih e J. Woodfill. Non-parametric local transforms for computing visual correspondence. *Computer Vision—ECCV'94*, pages 151–158, 1994.