

Rúben Silva Oliveira

# SISTEMA DE VISÃO ARTIFICIAL EM CENÁRIOS NATURAIS: SEGMENTAÇÃO E DETEÇÃO DE LIMÕES

Tese de doutoramento em Engenharia Mecânica, ramo de Gestão e Robótica Industrial, orientada pelo Prof. Doutor José Manuel Torres Farinha e Prof. Doutor. Joaquim Norberto Pires, apresentada ao Departamento de Engenharia Mecânica da Faculdade de Ciências e Tecnologia da Universidade de Coimbra

Agosto, 2017



UNIVERSIDADE DE COIMBRA





C •

FCTUC FACULDADE DE CIÊNCIAS  
E TECNOLOGIA  
UNIVERSIDADE DE COIMBRA

Rúben Silva Oliveira

# SISTEMA DE VISÃO ARTIFICIAL EM CENÁRIOS NATURAIS: SEGMENTAÇÃO E DETEÇÃO DE LIMÕES

Tese de Doutoramento em Engenharia Mecânica, no ramo de Gestão e Robótica Industrial, apresentada ao Departamento de Engenharia Mecânica da Faculdade de Ciências e Tecnologia da Universidade de Coimbra para obtenção do grau de Doutor

Orientadores: Prof. Doutor José Manuel Torres Farinha  
e Prof. Doutor. Joaquim Norberto Pires

Coimbra, 2017



Aos meus avós.

*“It is not the critic who counts: not the man who points out how the strong man stumbles; or where the doer of deeds could have done them better. The credit belongs to the man who is actually in the arena, whose face is marred by dust and sweat and blood; who strives valiantly; who errs, who comes short again and again, because there is no effort without error and shortcoming; but who does actually strive to do the deeds; who knows great enthusiasms, the great devotions; who spends himself in a worthy cause; who at the best knows in the end the triumph of high achievement, and who at the worst, if he fails, at least fails while daring greatly, so that his place shall never be with those cold and timid souls who neither know victory nor defeat.”*

*Theodore Roosevelt*



# AGRADECIMENTOS

Em primeiro, à minha família e namorada por todo o apoio, acreditarem que era possível e por me motivarem a superar os obstáculos encontrados.

Aos orientadores pela árdua tarefa de “simplesmente” orientar o trabalho desenvolvido, permitindo-me progredir de forma independente e com raciocínio próprio. A eles se deve parte da evolução do meu conhecimento científico e industrial, conhecimento esse que vai além do puro conhecimento intelectual.

Ao Crédito Agrícola pelo Prémio CA atribuído, motivando o desenvolvimento de trabalhos de investigação para a concretização da ideia proposta.

À Infaimon Lda, em particular ao Eng<sup>o</sup> Vitor Medeiros, pelo apoio técnico prestado e pela cedência temporária de filtros óticos para aquisição de imagens dos frutos e de outro equipamento de visão artificial usado na deslocação ao Volkswagen Tracking Challenge 2013.

Ao Laboratório de Computação de Elevado Desempenho (LaCED) - ISEC/IIA/IPC, em particular ao Prof. Miguel Couceiro, pelas horas de computação cedidas gratuitamente para realização das simulações no *cluster* de elevado desempenho.

À Prof. Justina Franco, docente na Escola Superior Agrária de Coimbra (ESAC), pelo conhecimento partilhado acerca dos frutos em geral, e da colheita de citrinos em particular. Agradeço ainda a sua prontidão em facilitar acesso às plantações da ESAC para recolha de imagens dos frutos.

Ao Centro de Engenharia Mecânica, Materiais e Processos (CEMPRE), enquanto instituição de acolhimento dos trabalhos de investigação desenvolvidos.

À FCT – Fundação para a Ciência e a Tecnologia pelo financiamento através de uma Bolsa de Doutoramento com o contrato SFRH/BD/86816/2012. Este trabalho foi ainda financiado por fundos do FEDER através do programa COMPETE – Programa Operacional Factores de Competitividade e por fundos nacionais da FCT – Fundação para a Ciência e a Tecnologia, pelo projeto: UID/EMS/00285/2013



UNIÃO EUROPEIA  
Fundo Social Europeu

# ABSTRACT

This thesis is focused on the integration and evaluation of several methodologies aiming to detect lemons on trees in natural outdoor scenarios, without controlled light conditions.

Current approach is different from the previous ones identified on literature, because it addresses a fruit not evaluated before using a simple vision system, which only acquire RGB images, suitable for real tasks conditions.

Based on state of the art review, the supervised learning methods are more effective than the unsupervised ones to segment and detect fruits in uncontrolled light scenarios. Considering such previous findings, it is created a reference database of lemons fruits scenarios to train, validate and test SVM classifiers and also to evaluate the performance of Haar, HOG and LBP classifiers.

Segmentation of citrus fruits is performed by a decision tree and three different schemes of SVM classifiers: through a global classifier trained by the strategy of one vs all; and through two sets of SVM classifiers, trained differently, by the strategy one vs one. The final performance evaluation is a two class problem, because only interests to distinguish between the remaining scenario and lemons in each image.

To avoid constraints from light changes on RGB color space, all simulations are performed using color components from YCbCr, HSV and chromaticity color spaces. Additionally, they were created two color indexes derived from chromaticity color space. Evaluating all combinations of color components plus SVM parameters by brute-force methods, it is an inefficient process. So, it is proposed a method based on hill-climbing algorithm that converge to the optimal configuration on SVM parametrization. After each stage of training and aiming to increase the number of features evaluated by SVM, it is proposed a method that filter the best set of features for further evaluations based on AUC (Area Under the Curve) results.

Since the ground truth database is not extensive as desired and its population contains some noise, it is proposed another method to split the data to train, validate and test the classifiers, instead of traditional k-fold. According to achieved results, this proposal fits well on the presented case.

Using the SVM-based segmented images, they are evaluated the Haar, HOG and LBP classifiers to perform the fruits detection. In opposition to previous findings, Haar and HOG classifiers performs better than LBP.

For the 133 images used to test segmentation methods, the best response is achieved by the global classifier related to true-positive rate of 93.14 % and a false-positive rate of 2.34 %. For such configuration, the segmentation is performed by 145 ms on average.

About detection, the best *Recall* of 92.39 % corresponds to a *Precision* of 80.19 % achieved by Haar classifiers, executed in 36 ms. If the precision is relevant for a specific application, the best

result corresponds to the use of HOG classifiers that response with a *Precision* of 85.37 % to a *Recall* of 76.09 %, executed on 47 ms on average per image.

Current results achieved by a simple vision system and by acquiring only RGB images are similar to the state of the art results, except when compared to Deep Learning methods whose perform better but also more computationally demanding to train.

Notwithstanding promising results, the proposed methodology tends to fail when exposed to low light scenarios. In such conditions it is highly advisable to use controlled external light sources, not only to compensate the low light conditions, but also to enhance the contrast between fruits and the remaining scenario. Additionally, it may be useful the introduction of new methods in order to detect correctly individual fruits grouped.

**Keywords:** SVM; Precision Agriculture; Computer Vision; Citrus segmentation; Fruits detection; agricultural robotics.



# RESUMO

A presente tese foca-se na integração e avaliação de metodologias distintas com o objetivo de fazer a deteção de limões nas árvores em cenários naturais, ou seja, de cenários com iluminação não controlada.

A proposta apresentada dista das restantes abordagens, discutida na revisão bibliográfica, por ser endereçada a um fruto até agora não avaliado e por apostar num sistema de visão artificial simples, passível de ser aplicado em contexto real de trabalho.

De acordo com o estado da arte, os métodos de aprendizagem supervisionada são os mais eficazes na segmentação e deteção dos frutos nas condições precedentes. Com base neste pressuposto foi criada uma base de dados de referência, composta por cinco classes de elementos, para treino e avaliação dos classificadores SVM na segmentação das imagens. Adicionalmente, um outro conjunto de imagens é usado para avaliar a deteção dos frutos através de classificadores baseados nos métodos de *Haar*, HOG e LBP.

A segmentação inicial das imagens é realizada a partir de uma árvore de decisão, em que é avaliada a resposta do modelo a um classificador SVM denominado global, treinado segundo a estratégia um contra todos, e a outros dois modelos compostos por vários classificadores SVM, treinados com a estratégia um contra um. Em qualquer um destes casos, o objetivo é separar os *pixels* dos limões da restante imagem, pelo que, no final, a avaliação do desempenho dos modelos de classificação é realizada considerando apenas duas classes de elementos ( $C=2$ ), a classe dos limões e a classe do *background*.

Devido à elevada quantidade de combinações possíveis de características de cor e parametrizações do SVM é proposto um método que convirja para as soluções de classificação ótimas. Além disso, este método é responsável pela determinação das características, aparentemente mais significativas para a correta segmentação dos frutos, ao longo dos vários estágios de treino em que se incrementa o número de características a analisar.

Com as imagens segmentadas são avaliadas as respostas de diversos classificadores na deteção dos frutos nas árvores, onde se conclui que os métodos de *Haar* e HOG permitem melhores resultados do que os classificadores LBP, contrariamente ao que seria esperado, de acordo com a bibliografia revista.

A metodologia de segmentação dos frutos proposta, para as 133 imagens de teste, apresenta uma taxa média de TP = 93.14 % para uma taxa de FP = 2.34 %, através da aplicação do classificador SVM, designado de global. Este processo de segmentação tem um tempo de execução, em média, de 145 ms.

Relativamente à deteção dos limões, a configuração que resulta na maior deteção dos frutos visíveis nas 44 imagens apresenta como desempenho uma *Precision* = 80.19 % e *Recall* = 92.39 %,

atingidos por um classificador de *Haar* executado, em média com uma duração de 36 ms por imagem; porém, se para a aplicação final for mais relevante a menor ocorrência de falsos-alarmes, o melhor modelo de classificação é obtido com a metodologia HOG, executada em 50 ms, onde se obtém uma *Precision* = 85.37 % e *Recall* = 76.09 %.

Os resultados alcançados por um simples sistema de visão artificial, apenas com aquisição de imagens a cores no espectro visível, estão ao nível dos melhores resultados alcançados e identificados na literatura revista. Excetuam-se apenas os resultados conseguidos pela introdução de modelos de *Deep Learning*.

Apesar dos resultados alcançados, e em função dos modelos de cor usados para atenuar os impactos das variações das condições de luz, é admissível que a metodologia proposta possa falhar em condições de iluminação deficiente. Nestes casos é recomendável a utilização de iluminação artificial, que poderá até aumentar o contraste entre os frutos e os restantes elementos captados nas imagens. Além disso é benéfica a introdução de metodologias que permitam a correta de deteção de frutos que se encontrem parcialmente sobrepostos.

**Palavras-chave:** SVM; Agricultura de Precisão; Visão Artificial; Segmentação de citrinos; Deteção de frutos; Robôs agrícolas.

# ÍNDICE

<b>LISTA DE FIGURAS.....</b>	<b>XV</b>
<b>LISTA DE TABELAS.....</b>	<b>XX</b>
<b>SIMBOLOGIA E ABREVIACÕES .....</b>	<b>XXI</b>
<b>1. INTRODUÇÃO.....</b>	<b>1</b>
<b>2. ESTADO DA ARTE .....</b>	<b>7</b>
<b>3. MÉTODOS DE APRENDIZAGEM.....</b>	<b>23</b>
3.1 Métodos de aprendizagem supervisionada e não supervisionada .....	24
3.2 Seleção de características.....	34
3.3 Treino e validação do modelo de decisão .....	35
<b>4. CLASSIFICAÇÃO POR SVM.....</b>	<b>39</b>
4.1 Métodos de otimização .....	43
4.2 Processamento em HPC .....	49
<b>5. DETEÇÃO .....</b>	<b>53</b>
<b>6. DADOS EXPERIMENTAIS.....</b>	<b>59</b>
6.1 Sistema de visão e base de dados.....	59
6.2 Normalização do modelo de cor .....	62
6.3 Quantidade de Dados .....	64
6.4 Métricas .....	66
<b>7. RESULTADOS EXPERIMENTAIS .....</b>	<b>69</b>
7.1 Classificador SVM global.....	73

7.1.1	Resultados dos Testes .....	75
7.2	Classificadores SVM independentes .....	77
7.2.1	Limões vs folhagem verde .....	78
7.2.2	Limões vs ramos .....	80
7.2.3	Limões vs elementos cinza.....	81
7.2.4	Limões vs elementos laranja .....	83
7.2.5	Limões vs folhagem seca .....	84
7.2.6	Resultados de teste .....	85
7.3	Classificadores SVM em cadeia .....	89
7.3.1	Limões vs folhagem verde .....	89
7.3.2	Limões vs ramos .....	91
7.3.3	Limões vs elementos cinza e laranja .....	94
7.3.4	Resultados de teste .....	97
7.4	Resultados da deteção dos frutos.....	99
<b>8.</b>	<b>ANÁLISE DE RESULTADOS .....</b>	<b>105</b>
<b>9.</b>	<b>CONCLUSÕES.....</b>	<b>115</b>
9.1	Perspetivas de desenvolvimento .....	117
9.2	Contributos .....	118
	<b>ANEXO A. PROCESSO DE SUBMISSÃO DE TAREFAS NO CLUSTER.....</b>	<b>119</b>
	<b>ANEXO B. COMBINAÇÕES DE CARACTERÍSTICAS COM MELHOR AUC DOS CLASSIFICADORES SVM.....</b>	<b>121</b>
1)	Classificador SVM global.....	121
2)	Classificadores SVM independentes.....	122
3)	Classificadores SVM em cadeia .....	124
	<b>ANEXO C. DESEMPENHO ISOLADO DOS CLASSIFICADORES SVM EM CADEIA</b>	<b>126</b>
1)	Limões vs folhagem verde .....	126
2)	Limões vs ramos .....	127
3)	Limões vs elementos cinza e laranja .....	129

<b>ANEXO D. PRODUÇÃO CIENTÍFICA .....</b>	<b>131</b>
<b>REFERÊNCIAS .....</b>	<b>135</b>



# LISTA DE FIGURAS

Fig. 2.1. Reflectância de citrinos vs folhagem. a) Análise de laranjas [10]; b) Análise de clementinas durante a maturação [11].	8
Fig. 2.2. Sistemas de visão multiespectrais: a) “Espelho” transmissão/reflexão [21]; b) Roda de filtro [17] c) Câmara JAI com dois sensores RGB+Nir [36].	17
Fig. 3.1. Fluxo de sinal na classificação.	23
Fig. 3.2. (esquerda) Imagem criada com ruído Gaussiano; (centro) histograma da imagem; (direita) máscara resultante do método de OTSU.	26
Fig. 3.3. (esquerda) Imagem em condições reais adquirida com filtro ótico 635nm; (centro) Histograma da imagem com estimativa da distribuição Gaussian; (direita) Máscara resultante do método de OTSU.	27
Fig. 3.4. (esquerda) Imagem simulada a partir da imagem real da segunda linha da Fig. 3.3; (centro) Histograma da imagem com parâmetros das distribuições Gaussianas; (direita) Máscara resultante do método de OTSU.	29
Fig. 3.5. (esquerda) Imagem simulada a partir da imagem real da quarta linha da Fig. 3.3; (centro) Histograma da imagem com parâmetros das distribuições Gaussianas; (direita) Máscara resultante do método de OTSU.	30
Fig. 3.6. Arquitetura da CNN LeNet-5 [89].	32
Fig. 3.7. Histograma de características para segmentação de limões.	34
Fig. 3.8. Distribuição da combinação das duas características da Fig. 3.7.	35
Fig. 3.9. Histograma das frequências relativas: (esquerda) elementos para treino e validação da classe dos limões; (direita) são acrescentados os elementos de uma subclasse do background.	36
Fig. 4.1. Classificação binário com SVM linear: (esquerda) diferentes hiperplanos possíveis; (direita) hiperplano correspondente à margem máxima entre as duas classes e respectivos vetores de suporte assinalados.	40
Fig. 4.2. Classificadores SVM para diferentes valores de C.	41
Fig. 4.3. Penalização relativa de más classificações pelo parâmetro c.	42
Fig. 4.4. Esquema representativo da estrutura da árvore de decisão.	43
Fig. 4.5. Esquema de geração de novas combinações.	45
Fig. 4.6. Algoritmo de geração de combinações.	45
Fig. 4.7. Exemplo de seleção de parâmetros pelo trepa colinas.	46
Fig. 4.8. Fluxograma do trepa colinas aplicado à determinação das parametrizações do SVM.	48
Fig. 4.9. Fotos dos bastidores do cluster Minerva, imagens cedidas pelo LaCED.	50
Fig. 4.10. Tempo processamento PC vs Cluster.	50

Fig. 4.11. Análise comparativa do tempo gasto na elaboração dos scripts e execução dos processos: PC i7 vs Cluster Minerva.....	51
Fig. 5.1. Exemplo da remoção de pequenas áreas: (esquerda) ROIs proposta pelo classificador; (direito) resultado da filtragem pela área. ....	56
Fig. 5.2. Exemplo da remoção ROIs com pouca área ativa: (esquerda) ROIs proposta pelo classificador; (direito) resultado da filtragem pela área ativa. ....	57
Fig. 5.3. Esquema da ROI resultante e da área a abranger.....	58
Fig. 5.4. Exemplo da união de ROIs que se interceptam: (esquerda) ROI proposta pelo classificador; (direito) resultado da união de ROI.....	58
Fig. 6.1. Esquema do sistema de aquisição de imagem. ....	59
Fig. 6.2. Sistema de visão experimental.....	60
Fig. 6.3. Exemplo de imagens captadas e respetiva máscara dos pixels relativo à classe dos limões. ....	62
Fig. 6.4. Matriz confusão genérica. ....	66
Fig. 6.5. Múltiplas deteções para o mesmo fruto: (esquerda) imagem segmentada; (direita) imagem RGB original.....	68
Fig. 7.1. Estrutura base da árvore de decisão.....	70
Fig. 7.2. Algoritmo de seleção das melhores combinações para elaboração de curvas ROC.....	72
Fig. 7.3. Desempenho nos dados de validação de classificadores SVM global treinados com cinco características: (esquerda) curva ROC; (direita) Acc correspondente. ....	73
Fig. 7.4. Desempenho nos dados de validação de classificadores SVM global para diferentes números de características e $\Delta t_{rel} \leq 500$ ms: (esquerda) curva ROC; (direita) Acc correspondente.....	74
Fig. 7.5. Detalhe da curva ROC da Fig. 7.4.....	75
Fig. 7.6. Comparação do desempenhos de classificadores SVM global ( $\Delta t_{rel} \leq 500$ m) com cinco características fixas vs quatro características determinadas pelo método iterativo: (esquerda) curva ROC; (direita) $\Delta t_{rel}$ correspondente. ....	75
Fig. 7.7. Curva ROC das operações morfológicas avaliadas após o classificador SVM global nas imagens de teste. ....	77
Fig. 7.8. Diagrama de extremos e quartis relativo à configuração selecionada para o classificador SVM global nos dados de teste: (esquerda) TPr; (centro) FPr; (direita) $\Delta t$ .....	77
Fig. 7.9. Estrutura da árvore de decisão usada na estratégia “um contra um”.....	78
Fig. 7.10. Desempenho de classificadores SVM independentes para folhagem até quatro características e $\Delta t_{rel} \leq 750$ ms: (esquerda) curva ROC; (direita) Acc correspondente.....	79
Fig. 7.11. Desempenho de classificadores SVM independentes para folhagem até quatro características e $\Delta t_{rel} \leq 500$ ms: (esquerda) curva ROC; (direita) Acc correspondente.....	79
Fig. 7.12. Tempo médio de processamento de cada configuração de classificadores SVM independentes para a folhagem: (esquerda) $\Delta t_{rel} \leq 500$ ms; (direita) $\Delta t_{rel} \leq 750$ ms. ....	80

Fig. 7.13. Desempenho de classificadores SVM independentes para ramos até três características e $\Delta t_{rel} \leq 500$ ms: (esquerda) curva ROC; (direita) Acc correspondente .....	81
Fig. 7.14. $\Delta t_{rel}$ dos classificadores SVM independentes para ramos de até três características.....	81
Fig. 7.15. Desempenho de classificadores SVM independentes para elementos cinza até cinco características e $\Delta t_{rel} \leq 750$ ms: (esquerda) curva ROC; (direita) Acc correspondente. ....	82
Fig. 7.16. $\Delta t_{rel}$ dos classificadores SVM independentes para elementos cinza de até cinco características. ....	82
Fig. 7.17. Desempenho de classificadores SVM independentes para elementos laranja até quatro características e $\Delta t_{rel} \leq 500$ ms: (esquerda) curva ROC; (direita) Acc correspondente. ....	83
Fig. 7.18. Desempenho detalhado dos classificadores SVM independentes para elementos laranja para $4\% \leq FPr \leq 6\%$ : (esquerda) curva ROC; (direita) $\Delta t_{rel}$ correspondente. ....	84
Fig. 7.19. $\Delta t_{rel}$ dos classificadores SVM independentes para elementos laranja de até quatro características. ....	84
Fig. 7.20. Desempenho de classificadores SVM independentes para folhagem seca até três características e $\Delta t_{rel} \leq 800$ ms: (esquerda) curva ROC; (direita) Acc correspondente. ....	85
Fig. 7.21. Curva ROC das configurações pré-selecionadas dos classificadores SVM independentes nos dados de teste. ....	86
Fig. 7.22. Curva ROC das operações morfológicas avaliadas após os classificadores SVM independentes nas imagens de teste.....	87
Fig. 7.23. Diagrama de extremos e quartis relativo à aplicação dos classificadores em independentes nos dados de teste: (esquerda) TPr; (centro) FPr; (direita) $\Delta t$ . ....	88
Fig. 7.24. Desempenho de classificadores SVM em cadeia para folhagem até quatro características e $\Delta t_{rel} \leq 500$ ms: (esquerda) curva ROC; (direita) Acc correspondente. ....	89
Fig. 7.25. Detalhe do desempenho de classificadores SVM em cadeia para folhagem até quatro características: (esquerda) curva ROC; (direita) $\Delta t_{rel}$ correspondente. ....	90
Fig. 7.26. Desempenho de classificadores SVM em cadeia para ramos até três características, para a Config A e $\Delta t_{rel} \leq 500$ ms: (esquerda) curva ROC; (direita) Acc correspondente. ....	91
Fig. 7.27. Desempenho de classificadores SVM em cadeia para ramos até três características, para a Config B e $\Delta t_{rel} \leq 500$ ms: (esquerda) curva ROC; (direita) Acc correspondente. ....	92
Fig. 7.28. Detalhe do desempenho de classificadores SVM em cadeia para ramos até três características, para a Config B: (esquerda) curva ROC; (direita) $\Delta t_{rel}$ correspondente. ....	92
Fig. 7.29. Desempenho de classificadores SVM em cadeia para ramos e $\Delta t_{rel} \leq 500$ ms em função da configuração do nível anterior: (esquerda) curva ROC; (direita) Acc correspondente. ....	93
Fig. 7.30. Detalhe do desempenho de classificadores SVM em cadeia para ramos com $\Delta t_{rel} \leq 500$ ms em função da configuração do nível anterior: (esquerda) curva ROC; (direita) $\Delta t_{rel}$ correspondente. ....	93
Fig. 7.31. Estrutura da árvore de decisão usada na estratégia “um contra um” em cadeia. ....	95

Fig. 7.32. Desempenho de classificadores SVM em cadeia para elementos cinza e laranja, até cinco características, para a Config B, Config C e $\Delta t_{rel} \leq 500$ ms: (esquerda) curva ROC; (direita) Acc correspondente. ....	96
Fig. 7.33. Detalhe do desempenho de classificadores SVM em cadeia para elementos cinza e laranja até cinco características, para a Config B e Config C: (esquerda) curva ROC; (direita) $\Delta t_{rel}$ correspondente. ....	96
Fig. 7.34. Exemplos da classificação resultante da aplicação do método de classificadores SVM em cadeia: (primeira linha) imagem RGB; (segunda linha) máscara de referência relativa aos limões; (terceira linha) máscara resultante da árvore de decisão; (quarta linha) máscara resultante das operações morfológicas. ....	98
Fig. 7.35. Curva ROC das operações morfológicas avaliadas após os classificadores SVM em cadeia nas imagens de teste. ....	99
Fig. 7.36. Diagrama de extremos e quartis relativo à aplicação do método de classificação em cadeia nos dados de teste: (esquerda) TPr; (centro) FPr; (direita) $\Delta t$ . ....	99
Fig. 7.37. Detecção pelo método de Haar: (esquerda) Precision vs Recall; (direita) $\Delta t_{rel}$ correspondente. .	100
Fig. 7.38. Detecção pelo método de HOG: (esquerda) Precision vs Recall; (direita) $\Delta t_{rel}$ correspondente. .	101
Fig. 7.39. Detecção pelo método de LBP: (esquerda) Precision vs Recall; (direita) $\Delta t_{rel}$ correspondente. .	101
Fig. 7.40. Casos de falha na detecção com ROIs TD a verde e ROIs FA a vermelho: (esquerda) imagem segmentada; (direita) imagem RGB original. ....	103
Fig. 8.1. Curvas ROC das metodologias de classificação dos limões nos dados de teste. ....	105
Fig. 8.2. Diagramas e extremos e quartis das três metodologias de classificação nos dados de teste: (esquerda) TPr; (centro) FPr; (direita) $\Delta t$ . ....	106
Fig. 8.3. Curvas Precision vs Recall de comparação dos métodos de Haar, HOG e LBP na detecção dos frutos. ....	108
Fig. 8.4. Tempo médio de processamento por imagem dos métodos de Haar, HOG e LBP na detecção dos frutos. ....	109
Fig. 8.5. Comparação da metodologia proposta com o estado da arte. ....	110
Fig. 8.6. Exemplos de sucesso na detecção de frutos na presença de um flare e limão verde: (esquerda) imagem segmentada; (direita) imagem RGB original. ....	112
Fig. 8.7. Exemplos de sucesso na detecção de frutos: (esquerda) imagem segmentada; (direita) imagem RGB original. ....	113
Fig. A.1. Ficheiro batch de submissão do processo. ....	120
Fig. A.2. Ficheiro com tarefas a submeter ao cluster. ....	120
Fig. B.3. Comportamento da métrica AUC. ....	121
Fig. B.4. Combinações de características com melhor AUC para o classificador SVM global. ....	122
Fig. B.5. Combinações de características com melhor AUC para a classe da folhagem. ....	122
Fig. B.6. Combinações de características com melhor AUC para a classe dos ramos. ....	123

Fig. B.7. Combinações de características com melhor AUC para a classe dos elementos cinza. ....	123
Fig. B.8. Combinações de características com melhor AUC para a classe dos elementos laranja.....	123
Fig. B.9. Combinações de características com melhor AUC dos classificadores SVM independentes para a classe da folhagem. ....	124
Fig. B.10. Combinações de características com melhor AUC dos classificadores SVM independentes para a classe dos ramos.....	124
Fig. B.11. Combinações de características com melhor AUC dos classificadores SVM independentes para a classe dos elementos cinza e laranja.....	125
Fig. C.12. Desempenho de classificadores SVM em cadeia para folhagem até quatro características e $\Delta t_{rel} \leq 500$ ms: (esquerda) curva ROC; (direita) Acc correspondente.....	126
Fig. C.13. Detalhe do desempenho de classificadores SVM em cadeia para folhagem até quatro características: (esquerda) curva ROC; (direita) $\Delta t_{rel}$ correspondente.....	127
Fig. C.14. Desempenho de classificadores SVM em cadeia para ramos até três características, para a Config A e $\Delta t_{rel} \leq 500$ ms: (esquerda) curva ROC; (direita) Acc correspondente.....	128
Fig. C.15. Desempenho de classificadores SVM em cadeia para ramos até três características, para a Config B e $\Delta t_{rel} \leq 500$ ms: (esquerda) curva ROC; (direita) Acc correspondente.....	128
Fig. C.16. Detalhe do desempenho de classificadores SVM em cadeia para ramos até três características, para a Config B: (esquerda) curva ROC; (direita) $\Delta t_{rel}$ correspondente.....	129
Fig. C.17. Desempenho de classificadores SVM em cadeia para elementos cinza e laranja, até cinco características, para a Config B, Config C e $\Delta t_{rel} \leq 500$ ms: (esquerda) curva ROC; (direita) Acc correspondente.....	129
Fig. C.18. Detalhe do desempenho de classificadores SVM em cadeia para elementos cinza e laranja, até cinco características, para a Config B, Config C: (esquerda) curva ROC; (direita) $\Delta t_{rel}$ correspondente.....	130

# LISTA DE TABELAS

Tabela 2.1. Resultados alcançados por abordagens anteriores .....	22
Tabela 6.1. Quantidade de instâncias para treino e validação dos classificadores .....	64
Tabela 7.1. Resultados da segmentação dos primeiros níveis da árvore de decisão.....	71
Tabela 7.2. Comparação dos resultados de validação e teste com diferentes configurações do classificador SVM global. ....	76
Tabela 7.3. Configuração selecionada para o classificador SVM global.....	77
Tabela 7.4. Configurações pré-selecionadas dos classificadores SVM independentes. ....	86
Tabela 7.5. Configurações dos classificadores SVM independentes usadas na avaliação do impacto das operações morfológicas.....	87
Tabela 7.6. Configurações selecionada para os classificadores SVM independentes e resultados de teste. ....	88
Tabela 7.7. Configurações selecionadas de limões vs folhagem.....	90
Tabela 7.8. Supressão de instâncias da classe ramos. ....	91
Tabela 7.9. Configuração selecionada de limões vs ramos.....	94
Tabela 7.10. Supressão de instâncias da classe elementos cinza.....	94
Tabela 7.11. Supressão de instâncias da classe elementos laranja. ....	95
Tabela 7.12. Supressão de instâncias da classe elementos cinza e laranja. ....	95
Tabela 7.13. Configuração selecionada de limões vs elementos cinza e laranja.....	97
Tabela 8.1. Detalhes das configurações pré-selecionadas dos modelos de detecção dos frutos. ....	110
Tabela 8.2. Comparação dos resultados alcançados com os anunciados em [35].....	111
Tabela C.1. Configurações selecionadas limões vs folhagem .....	127
Tabela C.2. Configuração selecionada limões vs ramos .....	128
Tabela C.3. Configuração selecionada limões vs elementos cinza e laranja .....	130

# SIMBOLOGIA E ABREVIações

Acc	<i>Accuracy</i>
AUC	<i>Area Under the Curve</i>
C	Número de classes de elementos consideradas.
C	Parâmetro SVM relativo ao peso das más classificações, designado por <i>box constraint</i> no Matlab 2016a
c	Parâmetro SVM associado ao custo relativo das más classificações, designado por <i>cost</i> no Matlab 2016a
CNN	<i>Convolutional Neural Networks</i>
CRF	<i>Conditional Random Field</i>
DCNN	<i>Deep Convolutional Neural Networks</i>
F1-score	Métrica de avaliação do desempenho dos métodos de segmentação (e deteção)
FA	Falsas deteções ( <i>false-alarms</i> )
FLDA	<i>Fisher Linear Discriminant Analysis</i>
FN	Falsos- negativos ( <i>false- negatives</i> )
FP	Falsos- positivos ( <i>false- positives</i> )
FPr	Taxa de falsos-positivos (FP)
fps	Taxa de aquisição de imagens expressa em imagens por segundo ( <i>framerate, frames per second</i> )
GigE Vision	Interface normalizada para transmissão a alta velocidade de dados de câmaras de visão industrial através do protocolo de comunicação <i>Gigabit Ethernet</i>
HOG	<i>Histogram of Oriented Gradients</i>
HPC	<i>High-Performance Computing</i>
HSI	Modelo de cor composto pelas componentes: <i>Hue; Saturation (Sat); Intensity</i> .
HSV	Modelo de cor composto pelas componentes: <i>Hue; Saturation (Sat); Value (Val)</i> .
IR	<i>InfraRed</i>
LBP	<i>Local Binary Pattern</i>
LDC	<i>Linear Discriminant Classifier</i>
LoG	<i>Laplacian of Gaussian</i>

MD	Falha na detecção de um fruto ( <i>missed-detections</i> )
ms	Milissegundos
NIR	<i>Near-Infrared Spectroscopy</i>
nm	Nanómetros
NN	<i>Neural Networks</i>
OVA	Onve-vs-all
OVO	<i>One-vs-one</i>
PCA	<i>Principal Component Analysis</i>
RBF	<i>Radial Basis Function</i>
<i>rgb</i>	Cromaticidades: <i>r; g; b.</i>
RGB	Modelos de cor original das imagens
ROC	<i>Receiver Operating Characteristic</i>
ROI	Região de interesse ( <i>Region of Interest</i> )
s	Segundos
SAE	<i>Sparse Autoencoder</i>
SOM	<i>Self-Organizing Map</i>
SVM	Máquinas de Vetor de Suporte ( <i>Support Vector Machine</i> )
TD	Frutos bem detetados ( <i>true-detections</i> )
TN	Verdadeiros-negativos ( <i>true-negatives</i> )
ToF	<i>Time of Flight</i>
TP	Verdadeiros-positivos ( <i>true-positives</i> )
<i>TPr</i>	Taxa de verdadeiros-positivos (TP)
YCbCr	Modelo de cor composto pelas componentes: Y; Cb; Cr.
$\Delta t_{rel}$	Tempo médio de processamento por imagem
$\sigma$	Parâmetro SVM relativo à função de núcleo RBF

# 1. INTRODUÇÃO

A escassez de alimentos e o crescimento da população mundial obriga a reavaliar os hábitos alimentares de todos nós e a melhorar a eficiência na produção agrícola.

A automação tem contribuído para a melhoria do setor agrícola, desde a criação de sensores para a análise dos solos, para a determinação do tempo ideal da fase da sementeira, para a monitorização da produção e para a colheita e processamento pós-colheita. Contudo, ainda se verifica uma forte componente manual em algumas produções, e ou fases dessas produções, por falta de resposta do meio científico e tecnológico.

O foco da investigação e dos seus resultados, alvo da presente tese, é referente à fase de colheita de frutos, nomeadamente nas vertentes de segmentação e deteção dos frutos na árvore.

Atualmente existem equipamentos industriais aptos a realizar colheitas massivas de frutos de forma muito mais rápida que os processos tradicionais - os que requerem mão-de-obra intensiva; no entanto, essas máquinas não são aplicáveis a qualquer fruto - há frutos sensíveis a choques mecânicos onde a mínima mazela faz com que entrem em estado de decomposição com o sério risco de propagação aos frutos adjacentes.

Adicionalmente, a falta de mão-de-obra especializada para a época da colheita é um sério problema para os agricultores. O facto de, usualmente, ser um trabalho sazonal e, com as restrições às políticas migratórias em alguns países, torna difícil o investimento no aumento das explorações. Não compensa investir na produção para depois deixar os frutos a apodrecer nas árvores.

Há, neste setor, uma lacuna que a automação pode colmatar, tal como proposto na presente dissertação, através do desenvolvimento de um sistema para colheita individual dos frutos. A parte fundamental deste sistema é o módulo de recolha e tratamento de informação que, neste caso concreto, servirá para detetar os frutos na árvore. Sem esta informação não é possível obter as coordenadas dos frutos nem tão pouco determinar os movimentos necessários para os colher.

No passado foram tentadas várias abordagens a este problema, mas poucas atingiram um nível de sucesso meritório de implementação e comercialização a nível industrial. Exceção feita ao sistema de colheita de morangos Agrobot [1].

Uma das medidas impeditivas, mais vezes referenciada, corresponde ao tempo de ciclo de apanha de cada fruto ser demasiado elevado para haver retorno de investimento em tempo útil. Diferentes empresários neste setor de atividade recomendam diferentes tempos máximos de ciclo para que o sistema automatizado seja compensatório relativamente à mão-de-obra humana. Num estudo realizado na colheita de pepinos foi determinado que o ciclo deveria ser inferior ou igual a 10 segundos [2]; no entanto, o sistema testado demorava 45 segundos a completar cada ciclo.

No caso dos citrinos, os resultados referem-se à colheita de laranjas onde é referido em [3] um tempo máximo de ciclo de seis segundos para que o sistema seja economicamente viável. Um estudo de viabilidade económica focado num robô de colheita de citrinos, apresentado em [4], tem por base um tempo de ciclo de três segundos, onde se constata que o sistema a usar tem que ser muito eficiente para compensar a sua utilização. Face à juventude deste tipo de soluções no mercado e à sua previsível evolução, é recomendável refazer o estudo com as condições atuais para se obterem resultados fidedignos.

Da análise detalhada de publicações e projetos relacionados com esta temática, são ainda identificadas outras fragilidades que levam ao descrédito destas soluções em contexto de utilização real no terreno.

No caso específico dos citrinos, alvo da presente tese, a cultura dos citrinos é feita em campo aberto, mas mesmo nas culturas em estufa, estas estão sujeitas a grandes variações nas condições de luz ao longo do dia. Isto faz com que os sistemas de visão artificial tenham que ser projetados tendo em atenção estas especificidades, para não influenciar negativamente os resultados em função do momento do dia em que as imagens são adquiridas.

Outros desafios impostos neste cenário são as semelhanças dos frutos com outros elementos do meio envolvente e o facto dos frutos poderem ser captados em condições de ocultação parcial, ou mesmo total. Não são condições ideais, mas sim as condições reais de trabalho que influenciam as escolhas das metodologias a adotar.

Enquanto Harrel [4], na sua análise económica, define que o sistema de colheita possa falhar em cerca de 5% dos frutos (ou até 15% num cenário otimista) para ser rentável, Bac *et al.* refere que os produtores aceitam uma solução baseada em robótica colaborativa, na qual o sistema mecanizado faça a colheita de, pelo menos 60% dos frutos, deixando os casos mais complexos para os operários.

Das abordagens analisadas, conclui-se que é comum a utilização de sistemas de visão multispectrais, compostos por uma câmara e uma roda de filtros, com o intuito de limitar a aquisição de imagens num intervalo de comprimentos de onda específicos.

Num sistema destes, há a vantagem de se obterem diferentes assinaturas visuais para o mesmo cenário. Contudo, são também identificadas algumas limitações. As imagens são adquiridas para cada filtro e admite-se que o cenário é estático, o que poderá não corresponder à realidade.

Além disso, é despendida uma parcela de tempo relevante na aquisição de uma imagem com cada filtro para processamento posterior, de forma a segmentar o cenário e a detetar os frutos. Além disso, numa aplicação destas deve ser evitado o uso de partes móveis para evitar o encravamento dos sistemas mecânicos devido à presença de poeiras e obstáculos no ambiente de trabalho.

Devido às especificidades do fruto, e do contexto em que se insere, na presente dissertação é proposto um sistema monocular sem roda de filtros. Com esta metodologia obtém-se diretamente um sistema mais económico, mecanicamente mais simples e com um tempo de

aquisição de imagens mais rápido do que os multiespectrais tradicionais. No entanto, como resultado desta abordagem, é levantado um outro desafio, que é o de conseguir fazer a segmentação do cenário e a detecção dos frutos com taxas de sucesso semelhantes às das restantes abordagens com os melhores resultados atuais. Este cenário é mais exigente no treino dos modelos baseados em inteligência artificial, por haver menor número de características independentes a analisar. Porém, ao invés da resposta de cada filtro de cor do sistema de visão, apenas há uma imagem decomposta nas componentes Red (R), Green (G) e Blue (B).

Esta abordagem é aplicada ao limão, um fruto da classe dos citrinos. Este fruto foi escolhido por não ter sido identificado na literatura outros trabalhos dedicados à sua detecção, e por ser um fruto mecanicamente robusto. A robustez do fruto tem importância na continuação do trabalho aqui apresentado, onde será possível integrar a metodologia desenvolvida num sistema de colheita e, ao ser mecanicamente robusto, há um risco menor de danificar mecanicamente os frutos envolvidos no estudo. Depois de consolidado o conhecimento sobre esta abordagem, todas estas metodologias devem ser extrapoladas para a colheita de frutos sensíveis, onde é esperado um maior impacto económico na sua implementação.

A inexistência de uma base de dados de imagens de limões devidamente segmentados forçou à aquisição de imagens e ao respetivo tratamento manual, de modo a que, a partir desta nova base de dados criada, pudessem ser avaliadas diversas abordagens na segmentação dos frutos.

Da base de dados criada constam 133 imagens com condições de luz diferentes, captadas de vários pontos de vista, com um intervalo entre 40 cm e 120 cm de distância entre a câmara e os frutos mais próximos. A partir destas imagens foram marcadas manualmente seis classes distintas de objetos: limões; folhagem verde; ramos; elementos cinzentos; elementos alaranjados; e folhagem seca. Outras 44 imagens são usadas na validação dos modelos de detecção.

A segmentação de imagens é realizada por classificadores de Máquina de Vetores de Suporte (SVM - Support Vector Machines) que analisam apenas a cor de cada *pixel*. Por apenas interessar classificar o cenário em duas classes, frutos e não frutos, são testadas várias abordagens no treino dos classificadores SVM: uma em que o modelo de classificação é treinado com a classe dos frutos a concorrer contra todas as outras; e uma segunda abordagem onde o problema inicial, aparentemente complexo, é decomposto em vários mais simples, em que a segmentação dos frutos é realizada por uma árvore de decisão e cada nó corresponde a um classificador SVM.

Os vários modelos de inteligência artificial podem determinar resultados em frações de segundos, mas o seu treino e a escolha do modelo mais adequado não tem a mesma celeridade. Os classificadores SVM foram treinados com diferentes números de características (*features*), de escala da função de núcleo, da penalização de más classificações, do custo diferenciado para classificar um fruto como *background* em relação ao *background* classificado como fruto, e da percentagem de ruído presente na amostra.

Nas abordagens iniciais, a variação dos parâmetros de treino era dada na forma de uma grelha, permitindo inferir como cada parâmetro influenciava a resposta do sistema. Contudo, trata-se de

abordagens que despendem demasiados recursos computacionais. Para melhor desempenho do treino dos classificadores foi adotado o algoritmo intitulado “trepas colinas” para fazer variar alguns dos parâmetros pretendidos e para despende o mínimo de recursos em parametrizações com os resultados menos satisfatórios. Dada a quantidade de simulações a realizar com as diferentes parametrizações tornou-se imperiosa a implementação da metodologia de treino dos modelos num centro de computação de elevado desempenho (High Performance Computing - HPC).

Para que se possam comparar os resultados alcançados entre a metodologia proposta e outros autores, a avaliação da prestação de cada modelo é feita através de curvas ROC (Receiver Operating Characteristic), características da matriz de confusão (*confusion matrix*), e de grandezas derivadas.

As avaliações experimentais apresentadas nesta dissertação são baseadas na segmentação de citrinos em condições de luz natural, ao ar livre, sem sistemas de iluminação artificial. O objetivo corresponde à classificação de regiões com imagens que têm maior probabilidade de serem limões. As principais restrições de tais cenários são a iluminação não uniforme e o fundo desordenado. São aplicadas metodologias para atenuar os efeitos das mudanças de luz, como a crominância, em vez da imagem RGB original. Os *pixels* são classificados como fruta ou como fundo usando um modelo de árvore de decisão para os classificadores SVM.

Através da abordagem de segmentação apresentada nesta dissertação é atingida a taxa de *True-Positives* (TP) de 90,55% e uma taxa de *False-Positives* (FP) de 1,61%. Além disso, o método proposto tem a vantagem de ser ajustável, para retornar diferentes proporções entre TP e FP.

Quanto à deteção, esta é alcançada com uma *Precision* = 80.19 % e *Recall* = 92.39 %, atingidos através de um classificador de *Haar* executado, em média, em 36 ms por imagem. Por outro lado, se o requisito mais relevante for a menor ocorrência de falsos-alarmes, o melhor modelo de classificação é obtido com a metodologia HOG, onde se obtém uma *Precision* = 85.37 % e *Recall* = 76.09 %, para uma execução média por imagem de 50 ms.

A restante estrutura da tese é a seguinte:

- No capítulo 2 é apresentada a revisão bibliográfica dos desenvolvimentos mais significativos relativos a sistemas de visão aplicados à deteção de frutos no terreno. Verifica-se que não existe consenso nas metodologias aplicadas na segmentação e deteção dos frutos, nem no sistema de visão empregue. Há investigadores que recorrem a iluminação artificial, a sistemas de visão multispectrais, hiperespectrais e ainda, outros a sistemas monoculares. Foram também identificados estudos com sensores de visão 3D e câmaras termográficas.
- O capítulo 3 faz uma introdução aos métodos de aprendizagem não supervisionada, em particular do método de OTSU, e supervisionada focada em SVM, bem como as estratégias de treino dos classificadores. São ainda referidas: a relevância e cuidados a ter

na seleção das características usadas no treino dos classificadores; e a divisão dos dados para treino, validação e teste dos classificadores.

- O capítulo 4 foca-se nos classificadores SVM, nas estruturas em árvore avaliadas e na metodologia de otimização desenvolvida para o treino dos classificadores com recurso a um *cluster* HPC.
- No capítulo 5 são apresentadas as metodologias base dos classificadores ensaiados para a deteção dos frutos após a segmentação por SVM.
- O modo de aquisição dos dados experimentais, a composição do sistema de visão, assim como a caracterização da população para treino, validação e teste dos classificadores são apresentadas no capítulo 6. Neste mesmo capítulo são também explicitados os modelos de cor usados com o objetivo de dar robustez às metodologias aplicadas face às variações das condições de luz e, por fim, são apresentadas as métricas aplicadas na avaliação do desempenho das metodologias implementadas.
- Os resultados individuais de cada metodologia avaliada na segmentação e deteção dos frutos é apresentado no capítulo 7, complementado com outros detalhes relativos aos resultados da segmentação por classificadores SVM, apresentados no Anexo B e Anexo C.
- No capítulo 8 é feita a análise dos resultados alcançados e a comparação com outros resultados identificados na bibliografia revista.
- Finalmente, no capítulo 9, são apresentadas as conclusões e os desenvolvimentos futuros propostos para melhoria dos resultados e integração dos desenvolvimentos desta dissertação numa máquina de colheita de frutos.



## 2. ESTADO DA ARTE

Este capítulo apresenta os resultados alcançados até à presente data, por outros investigadores, assim como as metodologias implementadas. Porém, mais que um resumo dos resultados alcançados em cada projeto, importa fazer a ponte com as metodologias aplicadas e o cenário onde são aplicadas, para que, aprendendo com as experiências anteriores, seja possível criar uma metodologia para um novo cenário com um elevado grau de confiança no seu sucesso.

Da revisão bibliográfica, importa avaliar o método de aquisição de imagens, as metodologias aplicadas ao processamento dos dados, o número de classes de segmentação e as metodologias utilizadas para a deteção dos frutos. No final deste capítulo é apresentada uma análise crítica a cada uma destas características.

As aplicações baseadas em visão por computador para cenários exteriores, onde a iluminação não é controlada e não há forma de uniformizar as condições de aquisição de imagem, são mais exigentes em termos das metodologias e algoritmos implementados, para que a resposta do sistema não seja fortemente afetada.

Neste contexto, é de estranhar que algumas das abordagens identificadas na literatura, nas quais os modelos de cor utilizados são fortemente afetados por variações na intensidade de luz, não mencionarem qualquer sistema de iluminação artificial para contrabalançar essas variações de luz.

Outras abordagens, apesar não incluírem nenhum sistema de iluminação artificial, recorrem a modelos de cor onde a intensidade de luz é isolada numa componente diferente dos dados relativos à cor. Estas apresentam maior robustez a alterações de intensidade de luz; no entanto, é expectável que falhem em condições cuja intensidade de luz é baixa. Neste último caso, a solução é recorrer a sistemas de iluminação artificial.

Várias pesquisas têm sido realizadas na procura de soluções de colheita mecanizada fruto a fruto, para diferentes espécies de frutos; abordagens alternativas, como a aplicação de químicos para promover a queda da fruta, a agitação dos troncos, o recurso a varejadores ou a sopradores de folhagem, são referenciadas como ineficientes e algumas provocam danos nas árvores ou até na própria fruta, [3].

Os estudos de viabilidade económica de uma solução de colheita mecanizada fruto a fruto [2-4] referem que no que concerne ao tempo de ciclo de colheita de fruto, as soluções conseguidas exigem tempos demasiado longos para se tornarem viáveis. Contudo, estes equipamentos podem trabalhar mais do que as 8 h/dia de um operário, permitindo compensar a falta de mão-de-obra disponível e podem ser autónomos. Nesta perspetiva, e em função da dimensão de cada exploração agrícola, pode compensar o recurso a um sistema de colheita mecanizado.

Os sistemas de visão hiperespectrais permitem obter uma resposta espectral em pequenos intervalos de comprimentos de onda, ou seja, há mais detalhes do que uma simples imagem RGB, ou numa análise multiespectral. Na agricultura, a aposta na utilização destes sensores tem-se focado na determinação dos parâmetros que avaliam a qualidade dos produtos frescos, em substituição das técnicas destrutivas para determinação dos mesmos parâmetros, [5-7]. Além disso, a análise hiperespectral pode ser usada para detetar defeitos na fruta, [8, 9].

Quando se trata de sistemas para segmentação e deteção de fruta, a análise hiperespectral serve para recolher a assinatura espectral, quer dos frutos quer dos restantes elementos do ambiente real. Estas assinaturas servem para determinar os intervalos do espectro eletromagnético onde há maior diferença entre cada elemento, bastando, posteriormente captar imagens nesse(s) intervalo(s) para distinguir os elementos presentes em cada cenário.

Esta abordagem foi seguida em [10], onde foi realizada uma análise hiperespectral aos citrinos (laranjas) e à folhagem para identificação dos intervalos de comprimento de onda eletromagnética de maior diferença entre estes dois elementos (Fig. 2.1 a). A partir deste estudo, os autores identificaram um conjunto de seis filtros óticos passíveis de ser aplicados ao sistema de visão multiespectral, através de uma roda de filtros.

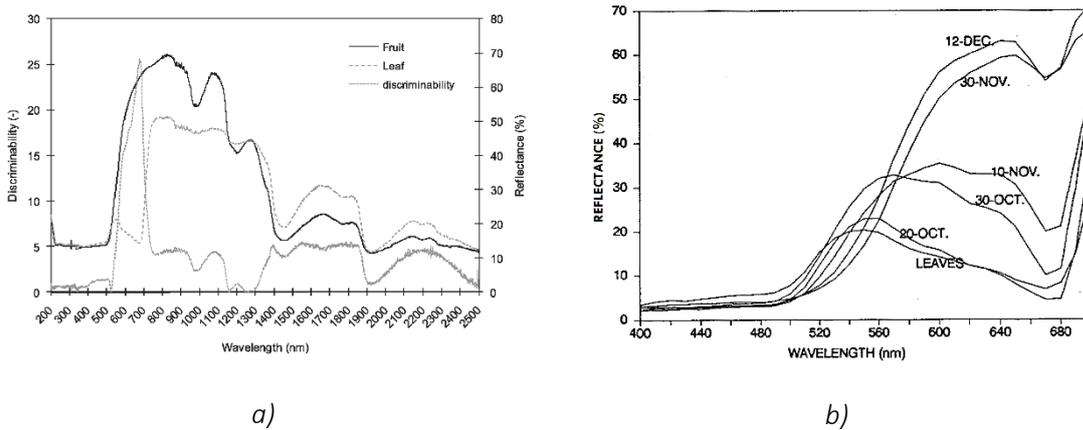


Fig. 2.1. Reflectância de citrinos vs folhagem. a) Análise de laranjas [10]; b) Análise de clementinas durante a maturação [11].

Pla e Juste [11] também fizeram um análise hiperespectral durante o crescimento de um fruto da famílias dos citrinos, neste caso em clementinas, onde verificaram que, com a maturação dos frutos a diferença para a folhagem aumenta (Fig. 2.1 b).

Apesar dos frutos analisados em [10] e [11] serem diferentes, as regiões do espectro onde se verifica uma maior diferença entre os frutos e a folhagem são as mesmas, acima dos 570 nm até cerca dos 700 nm, conforme é possível verificar na Fig. 2.1. Esta coincidência no intervalo de comprimentos de onda não é de estranhar, pois ambos os frutos pertencem à família dos citrinos, pelo que terão características semelhantes.

O intervalo de radiação infravermelha (IR - *InfraRed*) até aos 1100 nm é, normalmente, usado para analisar a presença de água nos frutos, e quanto mais água menor o grau Brix (°Bx). Como os frutos contêm maior percentagem de água do que a folhagem é expectável que hajam intervalos de radiação eletromagnética na região do infravermelho próximo do visível (NIR - *Near-Infrared Spectroscopy*) com diferenças acentuadas pela radiação refletida pelos frutos e folhagem.

Fernández *et al.* [12] apresenta uma análise à assinatura espectral do solo, folhas e pedúnculo de uvas para determinar quais os filtros óticos que podem ser utilizados para distinguir estes elementos. Neste estudo, os autores optaram por distinguir a fruta dos restantes elementos a partir de imagens RGB. Além disso, como seria de esperar, as folhas, como contêm maior percentagem de água do que os dois elementos analisados, refletem maior radiação no intervalo dos 700–900 nm.

É incomum usar um sistema de visão hiperespectral na deteção de frutos devido: à sua sensibilidade à variação da luz ambiente quando o sensor não está próximo do alvo; à complexidade do mecanismo para captação de um cenário por ser um sensor de linha e não de área; e ao seu custo.

Contudo, Okamoto e Lee [13] investigaram a deteção de três variedades de citrinos ainda verdes no terreno, portanto sem controlo da intensidade de luz natural, com recurso a um sistema de visão hiperespectral com uma demora entre 22 e 65 segundos para capturar uma imagem completa do cenário pretendido. Os autores tinham como principal objetivo estimar mapas de produtividade das plantações, mas a presença de demasiados falsos-positivos (FP), nos resultados alcançados, faz com que os autores duvidem da metodologia avaliada.

De acordo com a análise espectral realizada em [11] (Fig. 2.1 b) constata-se que é mais exigente a distinção de citrinos verdes da folhagem do que em fase de maturação avançada, recorrendo unicamente à percentagem de radiação refletida na região do NIR.

Antecipa-se desde já que a vantagem em analisar o intervalo de radiação IR até aos 1100 nm é a distinção dos elementos que contêm na sua constituição porções de água diferentes. Como é relatado nos capítulos seguintes é difícil distinguir os limões da folhagem seca recorrendo apenas a imagens captadas no intervalo de radiação visível, pois apresentam tonalidades de cor semelhantes. Neste caso, perspectiva-se que a análise desta região do espectro permita uma distinção mais fácil e precisa dos frutos e da folhagem seca, ou de outros elementos que contêm porções de água diferentes na sua constituição.

Esta é a filosofia seguida nas abordagens multiespectrais, em que são avaliadas imagens captadas com diferentes filtros óticos para separar os frutos dos restantes elementos de forma indiferenciada, uma abordagem designada por frutos *versus background*, onde apenas são consideradas duas classes de elementos ( $C=2$ ). Também é possível usar a mesma metodologia para distinguir as várias classes de elementos que compõem o *background* entre si, resultando num sistema com mais de duas classes ( $C>2$ ), em que, tipicamente, é criada uma classe para os

frutos, folhas, ramos e solo. O detalhe na discriminação do cenário, isto é, o número de classes de elementos, é determinado em função das necessidades de cada aplicação.

Apesar das desvantagens associadas a um sistema de visão multiespectral para a detecção de frutos nas plantações, o seu poder em obter imagens em vários intervalos do espectro eletromagnético faz com haja um maior número de características independentes a descrever cada cenário, o que se traduz numa maior probabilidade de sucesso na segmentação. Por esse motivo, este é o sistema mais popular na literatura revista. Contudo, no que respeita às metodologias aplicadas para fazer a segmentação e detecção dos frutos não há consenso; a comunidade científica ainda não determinou uma metodologia que seja economicamente viável para os produtores agrícolas.

Identificada a falta de resposta do meio científico e tecnológico para um sistema de colheita fruto a fruto e ao focar o estudo num fruto que, no conhecimento do autor, não foi alvo de abordagem anterior, importa referir e discutir o sucesso de diferentes metodologias desenvolvidas para situações passíveis de extrapolação, as quais serviram para pré-selecionar a abordagem realizada na presente dissertação.

Em 2009, Hannan *et al.* [14], foi elaborado um estudo para a detecção de laranjas utilizando apenas imagens captadas por uma câmara a cores. A primeira parte dessa publicação focou-se no método a usar para segmentar cada imagem em duas classes: fruta (laranjas); e *background*. Após converter o modelo de cor RGB para cromaticidades ( $r$ ,  $g$  e  $b$ ), os autores identificaram a componente  $r$  como a de maior significância na segmentação das duas classes. A tentativa de segmentação com o método de OTSU, originalmente apresentado em [15], para determinar o limiar (*threshold*) de intensidade de  $r$ , foi excluída devidos aos resultados insatisfatórios em situações cuja distribuição da cromaticidade  $r$  apresentava uma forma unimodal.

Como alternativa, os autores desenvolveram uma equação para determinar o limiar de intensidade de  $r$  em função da média e do desvio padrão desta cromaticidade em cada imagem, influenciado ainda por um terceiro parâmetro que foi determinado por tentativa e erro. Esta metodologia de segmentação testada num conjunto de 20 imagens reportou uma taxa de verdadeiros-positivos (TP) superior a 85% e de FP inferior a 5%.

Ainda relativamente à análise da cromaticidade  $r$ , o mesmo estudo [14] avaliou a sua resposta durante 24 horas para um mesmo cenário, usando iluminação artificial no período de menor intensidade de luz solar, e imagens adquiridas com e sem *flash* no interior da copa das árvores de onde concluíram que não há variação significativa na intensidade média de  $r$ . Quanto à detecção dos frutos, foi realizada com base no perímetro das regiões provenientes da segmentação com taxas de TP de 93 % e de FP de 4 %, para um total de 110 imagens, demonstrando ser eficaz também na detecção de frutos agrupados e parcialmente sobrepostos, assim como em frutos parcialmente ocultos.

Bulanon *et al.* [10] socorrem-se de um sistema multiespectral para a detecção de laranjas numa plantação sujeita a iluminação natural do sol. Após obter a resposta espectral da folhagem e dos frutos, selecionaram um conjunto de seis filtros óticos passa-banda. Posteriormente, avaliaram os

resultados da segmentação através de redes neuronais (NN – *Neural Networks*) e de um classificador linear (LDC – *Linear Discriminant Classifier*). Como a resposta inicial, usando NN e as imagens dos seis filtros, não era satisfatória (FP = 38,4 %), aplicaram a metodologia de análise de componentes principais (PCA - *Principal Component Analysis*) com o objetivo de processar apenas as principais características diferenciadoras da folhagem em relação aos frutos (C=2).

Com as imagens dos três filtros identificados como principais (600 nm, 650 nm e 750 nm) da análise PCA, foi criada uma nova imagem por cenário, com apenas um canal, a ser segmentada por NN e LDC. Desta avaliação resultou um melhor desempenho das NN, que ainda foi melhorado ao usar diretamente as três imagens dos filtros principais, com uma taxa de TP = 91 % para FP= 15 %. Depreende-se, da análise PCA, que esta pode ser usada para identificar as características que melhor distinguem as duas classes e que devem ser usados os dados originais, isto é, as imagens captadas com cada filtro, ao invés de procurar reduzir o tamanho da amostra ao criar uma imagem de um canal apenas a partir do conjunto de imagens originais.

Por fim, para realizar a detecção de frutos na árvore foi aplicada a transformada de *Watershed* à imagem resultante da segmentação por NN. A localização de cada fruto é obtida pelo centro de massa de cada região individual proveniente da transformada de *Watershed*. As taxas de sucesso desta abordagem foram as referidas anteriormente, não estando disponibilizadas as taxas referentes unicamente à segmentação, mas sim ao resultado final de todas as metodologias empregues.

Além dos resultados atingidos, os autores, [10], referem ter adquirido imagens em 80 cenários diferentes; mas, devido a alterações do cenário durante a aquisição de imagens para cada filtro (cenário dinâmico), só foram usados 51 cenários no estudo apresentado; este problema também foi referenciado em [16, 17].

Uma outra abordagem multiespectral à segmentação e detecção de citrinos é proposta pela mesma equipa de investigadores em [18-21]. O sistema de visão proposto consiste num espelho, responsável por transmitir parte da radiação eletromagnética a uma câmara e refletir outra parte para uma segunda câmara. Nesta proposta, o plano do espelho encontra-se a 45° com o plano de imagem de cada câmara e é preparado para refletir a radiação NIR para uma câmara e transmitir a radiação visível à outra [18], conforme é ilustrado na Fig. 2.2 *a*. Este sistema, com a aquisição de imagem sincronizada, evita os problemas de movimento do cenário nos sistemas de visão multiespectrais por roda de filtros.

Apesar do sistema de visão apresentar, conceptualmente, a vantagem em adquirir imagens em simultâneo em dois intervalos do espectro, a restante abordagem impossibilitou a avaliação desse desempenho, essencialmente por dois motivos:

- 1) Recurso a um filtro ótico de 800 nm para captação de imagens na região do IR, enquanto trabalhos científicos anteriores identificam o intervalo de 570 nm até cerca dos 700 nm como a maior diferença entre a fruta e a folhagem (tal como atrás mencionado);
- 2) Utilização do modelo de cor RGB para a segmentação das imagens.

Este é um modelo de cor influenciável por alterações na intensidade de luz, como os próprios autores referem em [18], pelo que é de estranhar a persistência deste modelo num espaço de iluminação não controlada, havendo modelos alternativos (ver seção 6.2 desta dissertação).

Relativamente ao sistema de aquisição de imagens, o autor testou o uso de filtros de densidade neutra e polarizador na câmara responsável pela aquisição na região visível do espectro: O filtro polarizador requer ajuste relativamente à posição e orientação da fonte de luz predominante, neste caso o sol, pelo que se torna uma solução pouco prática em contexto real; Quanto ao filtro de densidade neutra, este demonstrou atingir menor taxa de FP, mas também menor taxa de TP relativamente aos mesmos cenários adquiridos sem qualquer filtro [19]. Quanto às metodologias de segmentação de imagem, pelos motivos acima descritos, não há confiança suficiente na discussão dos seus resultados quanto às suas taxas de sucesso. No entanto, os métodos FLDA (*Fisher Linear Discriminant Analysis*) e SOM (*Self-Organizing Map*) tendem a não representar uma solução viável a esta situação, devido ao excessivo tempo de processamento, [19, 21].

Fernández *et al.* [12] usa também um sistema multiespectral com uma roda de filtros para classificar elementos de um cenário vinícola em cinco classes distintas ( $C > 2$ ): fruta (uva); folhagem; pedúnculos; ramos; solo e restante *background*. Enquanto a segmentação da fruta é realizada com imagens a cores, através do modelo de cor CIELAB, os restantes elementos são distinguidos por imagens adquiridas na região do NIR. Para tal, recolheram a assinatura espectral dos diferentes elementos e identificaram, a partir de uma análise PCA, as três regiões do espectro onde mais facilmente se conseguem distinguir uns elementos dos outros. Não obstante a resposta final do sistema ser para a classificação em mais do que duas classes, os autores procedem a várias classificações binárias até obterem a resposta final, numa estrutura idêntica a uma árvore de decisão.

Neste processo recorrem a classificadores baseados em *k-means*, um modelo de aprendizagem não supervisionado, resultando em taxas de TP = 68.3 % e FP = 1.1 % na classificação da classe dos frutos e uma precisão global de 75.8 %.

Este estudo foi atualizado em [17], no qual os autores acrescentam dados tridimensionais provenientes de um sensor ToF (*Time-of-Flight*) que são combinados com as imagens adquiridas (Fig. 2.2 b), sendo o método para classificação do cenário alterado. Nesta nova abordagem recorrem a classificadores SVM, um método de aprendizagem supervisionada, para classificar cada cenário nas mesmas cinco categorias do estudo anterior através de uma estrutura em árvore com quatro níveis. As ações adotadas permitiram melhorar a classificação de uvas para TP = 83.1 % e FP = 2.01 %. Além disso, o estudo foi alargado à segmentação de maçãs, onde conseguiram taxas de TP = 96.6 % e FP = 0.19%. Outro estudo que aborda a deteção de uvas é apresentado em [22].

É notório que a nova metodologia resultou numa melhoria dos resultados; no entanto, como foi alterado o método de classificação e acrescentados dados do sensor ToF, não foi possível concluir qual a influência de cada uma destas alterações na resposta do sistema.

Analisando a metodologia empregue em [12], os métodos de aprendizagem não supervisionada decidem, caso a caso, a determinação do limiar da classificação binária. Alterando o sinal de entrada, neste caso as imagens de cada cenário, é sempre forçada a classificação do cenário em cinco classes distintas, mesmo que alguma delas não exista em alguns cenários. A forma de ultrapassar esta limitação é através do recurso a métodos de aprendizagem supervisionada, em que é determinado o melhor modelo com base num conjunto de amostras, que, preferencialmente, deve incluir o máximo de variações possíveis do sinal de entrada. O recurso a classificadores SVM [17] não força a existência das cinco classes inicialmente previstas em cada cenário captado, mas, se as condições com o que o modelo de decisão foi treinado forem alteradas, não é assegurado o correto desempenho do mesmo. Nestes casos, é fulcral que a avaliação seja realizada em componentes de modelos de cor que não variem facilmente com a alteração das condições de luz, de forma a garantir um desempenho estável do sistema.

Também foram identificados vários estudos realizados com sistemas de visão monoculares para a deteção de diversos frutos nas árvores, quer na fase de colheita quer para frutos ainda verdes, para a elaboração de mapas de produtividade.

Okamoto e Lee, [23], apresentam um estudo com um sistema monocular, composto por uma câmara a cores, para a deteção de citrinos ainda verdes nas árvores. Como a cor dos frutos é muito semelhante à da folhagem, é realizada a segmentação não só pela cor como também pelos contornos das regiões da imagem. Numa primeira fase é segmentada a folhagem conjuntamente com os frutos dos ramos a partir de um limiar fixo definido para a componente H (*Hue*) do modelo de cor HSI. Os autores constataam que a superfície dos citrinos verdes apresenta uma superfície com menor textura que a folhagem, resultando num valor de gradiente mais elevado dos contornos dos frutos na imagem. Não obstante ser reportada uma elevada taxa de deteção dos frutos, usando o método de deteção de circunferências, a metodologia proposta foi avaliada unicamente em cinco imagens. É recomendável expandir os testes a um maior número de imagens para que se possa ter confiança suficiente nos resultados alcançados.

Uma metodologia cientificamente mais elaborada é apresentada em [24] para a deteção de citrinos verdes nas árvores, em ambientes sujeitos à luz natural do sol. Para atenuar os efeitos das variações de luz são usados os modelos de cor HSI e YCbCr aplicando a componente I, do modelo de cor HSI, e a transformada logarítmica para aumentar a intensidade das regiões mais escuras e a equalização do histograma desta componente. Estas transformações visam uniformizar as condições de luz entre as várias imagens a adquiridas.

Com base em amostras de imagens, os autores [24] sugerem excluir da análise parte dos *pixels* relativos à folhagem pela aplicação de um limiar fixo às componentes Cb, Cr e *Hue* dos modelos de cor analisados. Porém, da análise das distribuições dos frutos e da folhagem nestas componentes individuais, verifica-se não existir um ganho significativo devido à sobreposição quase total das distribuições das duas classes.

Quanto à detecção dos citrinos, é aplicado o método de Janela Deslizante, onde cada sub-região da imagem, proveniente deste método, é analisada individualmente por uma variação do método *eigenface*, popular na detecção de faces humanas, intitulado de “*eigenfruit*”, e uma versão circular do filtro de *Gabor* às componentes S e I do modelo de cor HSI. Não é explícita a metodologia usada para selecionar o uso destas componentes como características a analisar e não outras. Duas vantagens enunciadas da aplicação do método *eigenfruit* são: necessitar apenas de amostras da classe que se pretende detetar, neste caso as laranjas verdes; não necessitar de amostras com os frutos parcialmente ocultos, pois é verificada a semelhança com o formato ideal do fruto.

Relativamente a taxas de sucesso, são reportados valores de TP = 75.3 % e de FP = 27.3 % na detecção de citrinos verdes [24]. Adicionalmente, não foram registadas diferenças significativas de frutos não detetados no lado das árvores à sombra ou sujeitas a exposição solar. No entanto, na parte da copa mais iluminada das árvores foi registada uma taxa superior de falsas deteções (FP). Relativamente à localização dos frutos, a metodologia sugerida funciona na elaboração de mapas de produtividade, mas para sistemas de colheita autónoma é necessário refinar os resultados devido aos desvios serem significativos.

Nesta sequência, a mesma equipa de investigadores apresenta um estudo [25] com a mesma metodologia de extração de características, mas desta vez aplicada à detecção de pêsegos em fase de maturação para a criação de mapas de produtividade. A segmentação das imagens foi avaliada por sete classificadores distintos, entre os quais NN e SVM com função de núcleo linear, sendo a detecção feita por três métodos diferentes. Os melhores resultados foram atingidos com a segmentação realizada por NN, TP = 85 % e FP = 23 %. Com o método SVM foi atingida a mesma taxa de TP mas com mais casos de falsos-positivos (FP = 27 %).

Lu *et al.*, [26], também abordaram a detecção de laranjas usando um sistema monocular com imagens apenas da região do espectro visível, sendo atingida uma taxa de TP = 86.81 % e de FP = 2.25 %, numa amostra de 20 imagens.

Os estudos realizados na segmentação e detecção de frutos com sistemas monoculares não estão restritos apenas aos citrinos, e as metodologias aplicadas não são consensuais na bibliografia revista.

Há frutos mais fáceis de tratar apenas pela cor, como é o exemplo a detecção de frutos vermelhos [27]. Neste estudo, onde também é avaliada a segmentação de laranjas, é focado o tempo despendido pelas variações do método de OTSU na segmentação dos frutos em ralação aos restantes elementos (C=2). Contudo, não são apresentadas as taxas relativas à elaboração da matriz confusão para a comparação deste método com outros.

Wachs *et al.*, [28], propõem a combinação de imagens a cores com imagens termográficas para a detecção de maçãs em plantações. Neste estudo é avaliada a resposta de cinco métodos distintos para combinar as imagens RGB com as imagens termográficas, e realizada a detecção dos frutos através do método de *Haar*, proposto originalmente em [29]. Foi avaliada a resposta referente à aplicação método de *Haar* individualmente às imagens RGB e termográficas, assim como à

combinação das duas imagens. Além da configuração permitir detetar apenas 74 % dos frutos visíveis é reportado um valor considerável de falsas deteções para as várias configurações.

Uma abordagem distinta da deteção de maçãs vermelhas e maçãs verdes nas árvores é apresentada em [30], onde se combina informação relativa à cor com a textura. Relativamente à cor, os autores concluem que a avaliação do nível de tonalidade vermelha não deve ser feita diretamente pela componente R do modelo de cor RGB, mas através do índice  $r_{Zhao}$ , derivado do mesmo modelo de cor, conforme representado na equação seguinte:

$$r_{Zhao}=3R-(G+B) \quad (2.1)$$

Assim, a segmentação de maçãs é dada por uma resposta elevada do índice  $r_{Zhao}$  e pela ausência de contraste entre *pixels* na superfície da fruta. A separação de frutos contíguos e a deteção destes é realizada por um algoritmo de procura de círculos. Apesar da proposta de metodologia, não são apresentados resultados da sua aplicação num número considerável de imagens, não sendo por isso possível avaliar o desempenho do filtro. Não é ainda especificado como é determinado o limiar a partir do qual uma região é considerada fruto ou *background* a partir do seu valor de  $r_{Zhao}$ .

A deteção de maçãs da variedade Fuji, também é abordada em [31], em cenários com diferentes condições de luz. Neste estudo, a segmentação da imagem em duas classes ( $C=2$ ) é realizada pelo método de crescimento de regiões após ser aplicado um filtro de mediana. Assim, todas as regiões homogéneas e contíguas têm a mesma identificação.

Quanto à deteção dos frutos, os autores avaliam a resposta de um classificador SVM, sistema de aprendizagem supervisionada, ao analisar unicamente características relativas à cor extraídas do modelo de cor HSI, características relativas à forma, e uma terceira avaliação realizada à análise global de todas as características. Em cada uma das análises foi ainda avaliado o desempenho do classificador com três funções de núcleo distintas. Após treinar o modelo SVM com 150 imagens distintas e usar outras 50 imagens para testar a sua resposta, o melhor desempenho foi obtido quando são analisadas as características de cor e forma em simultâneo e usando a função de núcleo RBF (*Radial Basis Function*), sendo detetados 93.3 % dos frutos presentes nas imagens. O mesmo modelo SVM foi testado noutra pomar, também de maçãs, onde foi reportada uma taxa média de TP = 89 % e de FP = 11% em 100 imagens, com um tempo de processamento médio de 352 ms.

Uma abordagem focada na textura para deteção de outras variedades de fruto, como o abacaxi e o melão, é apresentada em [32]. São avaliados diferentes métodos de deteção de pontos de interesse e extração de descritores com classificadores SVM para a deteção de frutos. Como estes frutos são colhidos quando ainda apresentam tonalidade verde é mais fácil distingui-los dos restantes elementos do cenário pela textura do que pela cor.

Hung *et al.*, [33], aborda a segmentação de elementos de uma produção de amêndoas, classificando o cenário em cinco classes distintas a partir de imagens RGB e IR. Apesar das imagens serem captadas com a mesma câmara, os autores assumem que o cenário é dinâmico,

contrariamente à maioria das abordagens com sistemas de aquisição de imagem multiespectrais. Também a contrastar com as abordagens já referidas é a metodologia utilizada para extração das características que permite classificar os cenários captados. Foi utilizado um banco de imagens, diferente das imagens captadas dos frutos, e obtidas as características através do método *Sparse Autoencoder* (SAE).

A partir destas características, cada *pixel* do cenário é classificado na sua respetiva classe pelo método *Conditional Random Field* (CRF). Em 1600 imagens, a precisão global da resposta do sistema é de 88 % e, relativamente à classe das amêndoas, uma taxa de TP = 71.3 %. Constataram ainda que a análise de características das imagens RGB e IR melhoravam a resposta do sistema usando apenas imagens RGB. No entanto, a metodologia proposta não foi desenhada para ser executada com baixa latência; o intuito era a elaboração de mapas de produtividade em que o processamento era realizado em diferido. Assim, o tempo de cinco segundos reportado no processamento de cada cenário não representa um obstáculo à utilização desta metodologia, mas a mesma não é viável para sistemas de colheita de fruta sem quaisquer modificações.

Sa *et al.*, [34], realizam a segmentação de pimentos, verdes e vermelhos, numa plantação com a mesma metodologia aplicada por Hung *et al.*, [33]. Além do modelo SAE para determinação das características diferenciadoras, os autores testaram ainda a resposta do classificador CRF com características provenientes dos métodos de HOG (*Histogram of Oriented Gradients*) e LBP (*Local Binary Pattern*). Relativamente às imagens adquiridas, também foram usadas imagens RGB e NIR captadas por uma câmara com um prisma responsável por transmitir a radiação visível a um sensor e a IR a outro (Fig. 2.2 c), evitando assim a necessidade de processamento adicional para estabelecer as correspondências entre as duas imagens, permitindo ainda captar cenários dinâmicos.

Dos três métodos para a obtenção das características analisadas, o LBP é aquele que permite atingir uma melhor resposta nas 10 imagens analisadas no estudo, atingindo uma AUC (*Area Under the Curve*) = 73.0 %. Esta resposta é melhorada ao analisar não só as características provenientes do método LBP, mas também a cor, pelas componentes do modelo HSV, com uma AUC = 81.2 %.

A extensão dos estudos desenvolvidos é apresentada em [35], onde é considerado um maior número de imagens, adquiridas durante o dia e a noite, com diferentes combinações de características. Além de novas metodologias de segmentação é ainda aplicado um algoritmo para deteção dos pimentos.

De acordo com os resultados dos testes realizados, [35], é atingida uma melhor resposta do sistema para as imagens adquiridas à noite, ou seja, quando há menor influência da iluminação ambiente/solar. Dos três métodos para extração de características das imagens IR, o LBP continua a ser aquele que apresenta a melhor resposta com larga margem de vantagem relativamente aos restantes. Para os testes noturnos é possível melhorar em 21 % a AUC inicialmente atingida pelo LBP analisando as características dos três métodos e a cor.

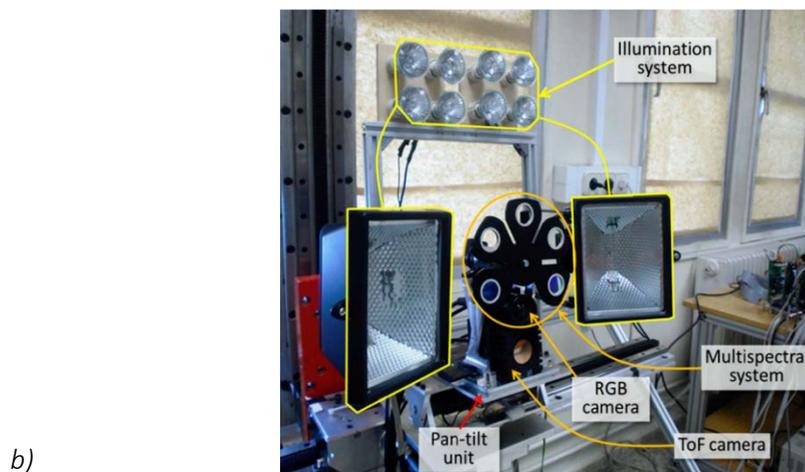
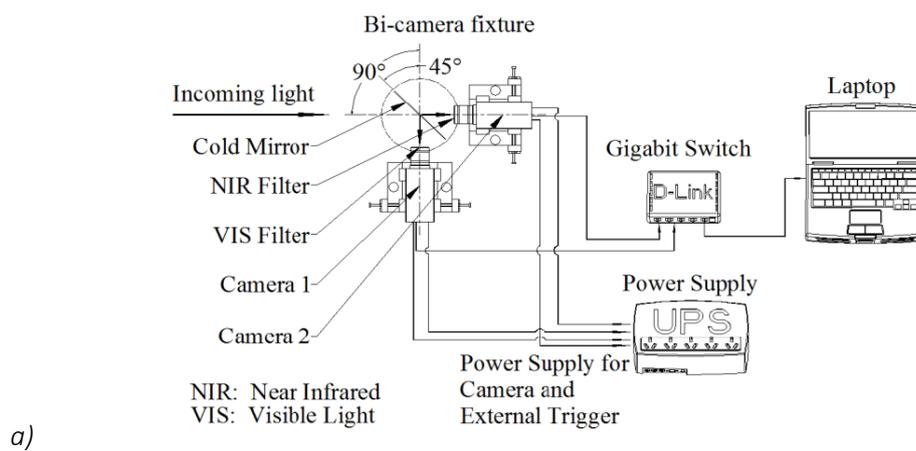


Fig. 2.2. Sistemas de visão multispectrais: a) “Espelho” transmissão/reflexão [21]; b) Roda de filtro [17] c) Câmera JAI com dois sensores RGB+Nir [36].

O facto de, em [35], as condições iniciais não serem apresentadas, dos resultados da combinação de características LBP com a cor não é possível fazer qualquer extrapolação a partir dos dados de [34] com o devido rigor. Pode-se assim afirmar que a combinação de características LBP com a cor [34] resultou numa melhoria de 11.2 %, relativamente à análise de, apenas, características LBP, enquanto para as imagens adquiridas durante o dia, [35], a combinação de características LBP, HOG, SAE e cor resultou numa melhoria de 13.7 % em relação às características LBP. Sendo o

diferencial pouco significativo, e não havendo registo do tempo gasto em processamento, é incauto avaliar com rigor o desempenho da aplicação desta metodologia a sistemas autónomos de colheita de fruta.

Quanto à deteção dos pimentos, os autores [35] optaram pelo método LoG (*Laplacian of Gaussian*) com o qual conseguiram detetar 69.2 % dos pimentos, com uma ocorrência média de 2.1 de falsos pimentos detetados por imagem. Adicionalmente, foi analisado o desempenho da metodologia de deteção em função da percentagem de área oculta de pimentos verdes e vermelhos individualmente, comparando ainda a resposta com a segmentação realizada apenas com características LBP, ou usando todas as características do estudo.

Ainda relacionado com os pimentos foi investigada a relação da posição relativa da câmara-pimento com a respetiva taxa de deteção [37]. Neste estudo atingiu-se uma taxa de deteção máxima de 69 % com apenas uma imagem e, quando combinando informação do mesmo cenário, mas com cinco imagens captadas em diferentes posições, a taxa de sucesso foi de 90 %.

Yamamoto *et al.*, [38], propõe uma metodologia baseada apenas em imagens RGB para estimar a produção de plantações de tomate. O desempenho registado por esta metodologia é uma precisão (*precision*) de 88 % e uma taxa de TP (também designada por *Recall*) de 80 %, relativa à deteção de tomates maduros e verdes num conjunto de 54 imagens. Apesar de haver sistemas de aquisição de imagem configurados de forma alternativa para melhor segmentar e detetar os frutos no terreno, os autores reforçam que essas soluções não são aplicáveis em contexto real de trabalho.

Essa abordagem [38] começa por segmentar os *pixels* de cada imagem em quatro classes de elementos ( $C > 2$ ) através de uma árvore de decisão cuja estrutura é determinada pela análise prévia das componentes de cinco modelos de cor. Seguidamente é realizada uma análise às regiões (*blobs*) dos *pixels* classificados como frutos, pelo método *random forest*, para determinar se essas regiões representam um fruto individual, um conjunto de frutos ou outros elementos. Posteriormente é utilizado um método diferente, em relação à restante bibliografia analisada, para proceder à separação dos frutos em grupo (*declustering*), *X-means*. Este método foi originalmente proposto por [39] e representa uma modificação do método *K-means*, em que não é necessário indicar o número de divisões pretendidas.

Relativamente à metodologia aplicada, importa salientar dois fatores que não são extrapoláveis a qualquer situação: O primeiro fator corresponde à análise do modelo de cor RGB, já referenciado como um modelo a evitar, mas que, neste caso, por ser utilizado um sistema de iluminação artificial e, por, nestas condições, a iluminação ambiente ser insignificante, não são esperadas variações significativas na resposta RGB; O segundo fator corresponde ao uso de dados relativos ao tamanho dos frutos, que é válido pelo fato da distância da câmara aos frutos ser sensivelmente constante nos ensaios realizados.

Em [40] é proposta uma metodologia para deteção de morangos com recurso a duas câmaras: uma fixa para determinar a região onde se encontra o fruto; a outra junto à garra de colheita para

uma análise mais detalhada do fruto e para determinar como colher o fruto. A segmentação do fruto é realizada numa primeira fase pela análise da cor, usando o modelo de cor OHTA [41]. Posteriormente, os autores admitem que a região próxima do cálice do morango assume uma tonalidade diferente da do ápice e assim identificam a região do pedúnculo ou cálice através da análise da componente *Hue* do modelo de cor HSI. Apesar desta metodologia recorrer ao modelo de cor OHTA, a sua formulação indica que será suscetível a alterações nas condições de iluminação, assim como o modelo RGB.

Em [42] é apresentado um sistema de visão com um feixe *laser*, diferente dos até agora analisados, para proceder à deteção de tomates em estufas. Esta metodologia permite segmentar os frutos da folhagem e dos ramos pelas componentes H e S do modelo de cor HSI, sendo indicado que todos os tomates foram detetados. No entanto, é uma metodologia que funciona, neste caso, devido à curta distância entre o emissor de feixe *laser* e os frutos, e por ser um ambiente que não está diretamente exposto à luz natural, contrariamente à produção de citrinos.

Outros estudos relacionados com a colheita dos frutos focam-se no desenvolvimento do mecanismo de colheita e nos movimentos a serem executados pelos braços robóticos.

Vários trabalhos foram identificados nesta temática para os citrinos: um conjunto de dois braços lineares que se deslocam numa guia linear colocada numa posição oblíqua [3]; na referência [43] é proposto um sistema baseado num cilindro pneumático para corte do pedúnculo, mas sem sistema de colheita dos citrinos; em [44] é apresentado um sistema com uma câmara fixa e outra colocada na garra de um robô de sete eixos, com o intuito de determinar a posição dos citrinos no espaço e avaliar o erro da posição para várias distâncias câmara-objeto.

Além dos citrinos foram identificados desenvolvimentos de sistemas de colheita robotizados para pepinos [2], kiwi [45, 46], espargos [47], maçãs [48-52], peras [53, 54], cerejas [55, 56], pimentos [57-60] e tomate [61]. Além da deteção dos frutos, existem temas complementares relativos à atividade agrícola onde também é utilizada a visão artificial, como a deteção de ervas [62, 63], aplicações de pulverização de fertilizantes ou pesticidas [64, 65], separação de frutos por calibre no terreno [66], ou a deteção de doenças [67].

De acordo com a bibliografia até agora revista, comprova-se que um bom desempenho na segmentação dos cenários captados é fundamental para a deteção dos frutos, seja para a elaboração de mapas de produtividade, ou para indicar aos equipamentos de colheita a localização dos frutos. Porém, não é exigido tanto rigor na deteção dos frutos para mapas de produtividade como nos sistemas robóticos relativamente à posição exata dos frutos. Os sistemas robóticos necessitam da posição exata dos frutos para a determinação do posicionamento do braço robótico, a fim de executar o movimento correto sem danificar o fruto nem embater em obstáculos naturais do cenário.

Para um sistema autónomo de colheita de frutos nem sempre é suficiente segmentar o cenário em duas classes ( $C=2$ ), frutos e restantes elementos; pode ser relevante segmentar o cenário em mais classes ( $C>2$ ) para detetar, por exemplo, o pedúnculo da fruta que deverá ser cortado para

separar a fruta da planta, tal como o estudo apresentado em [12] demonstra. Noutras abordagens interessa identificar outros elementos que não os frutos para aplicação de fertilizantes e ou pesticidas.

Em [68-70] são apresentados métodos distintos para a classificação de cenários agrícolas em várias classes ( $C > 2$ ). Enquanto em [69, 70] é realizada uma análise multiespectral; em [68] é feita a reconstrução 3D dos cenários captados a partir da aquisição sequencial de imagem. Dey *et al.*, [68], relatam que o classificador SVM atinge um melhor desempenho que o *Gaussian Mixture Model* (GMM) na classificação da classe dos *pixels*, sendo a resposta do sistema ainda melhorada quando a análise SVM é combinada com o método CRF, para analisar, não só cada *pixel* individualmente, mas também considerar os dados dos seus vizinhos.

Estes estudos poderão ser usados, posteriormente, para averiguar quais os métodos que têm melhor desempenho na segmentação dos elementos pretendidos em cada cenário, podendo até servir de base de dados para avaliação de outras metodologias, pois o trabalho de classificar manualmente os vários elementos em imagens é bastante moroso; contudo, é fundamental para uma avaliação de desempenho de diferentes metodologias.

Além dos estudos revistos neste capítulo, a revisão das metodologias aplicadas a outros frutos e os respetivos resultados podem ser consultados nas publicações [71-74]. Nestes artigos encontram-se abordagens que fazem uso de informação de profundidade e análises termográficas (uma da região do IR diferente do NIR) que não são detalhadamente analisados nesta monografia por recorrem a metodologias e sistemas físicos que não se enquadram na abordagem realizada.

A nível europeu foram identificados dois projetos que exploram a aplicação da robótica na colheita dos produtos de cada plantação. Enquanto o primeiro projeto realizado (CROPS) [75] serviu para avaliar diferentes metodologias em diferentes frutos, o projeto conseqüente (SWEEPER) [76], com execução prevista até ao ano de 2018, tem como objetivo implementar e avaliar o desempenho, em condições reais, de um sistema robótico numa plantação de pimentos. Constata-se que, destes projetos resultam publicações científicas relevantes para o desenvolvimento da robótica aplicada ao sector agrícola.

Dos estudos revistos há um melhor desempenho generalizado quando se empregam classificadores SVM ou NN na classificação dos cenários captados. Porém, a metodologia designada de *Deep Convolutional Neural Networks* (DCNN) começa a ganhar notoriedade na classificação e deteção de objetos em imagens. Contudo, um dos grandes impasses desta metodologia são os recursos computacionais necessários para o treino do classificador. Relativamente ao desempenho dos classificadores DCNN em aplicações agrícolas, já existem estudos publicados pelos quais se podem analisar os resultados na deteção de diferentes frutos [77-79] e na segmentação de ervas [80].

Da revisão bibliográfica conclui-se que o uso de um sistema multiespectral atinge melhor desempenho do que os sistemas que registam apenas imagens no intervalo visível do espectro.

No entanto, é normalmente assumido, nestes sistemas, que o cenário é estático, quando na realidade, e numa situação real de trabalho, o cenário é dinâmico.

Relativamente ao modelo de cor usado, o mais relevante é usar um modelo que concentre a intensidade de luz em apenas uma das suas componentes, tornando assim o modelo RGB inadequado para os casos em que não é possível controlar devidamente as fontes luminosas, nem ter os cenários iluminados de forma homogênea e sem sombras. Há uma tendência para se usar os modelos de cor HSV e HSI, através dos quais se consegue extrair informações relativas à tonalidade de cor pela primeira componente (*Hue*) destes modelos.

Quanto às características a analisar para determinar que elementos pertencem à classe dos frutos, verifica-se uma predominância da análise da cor em frutos cuja tonalidade seja diferente da folhagem (verde). Contudo, métodos populares na análise de texturas, noutras áreas de aplicação, também foram avaliados em diversos estudos com enfoque nos frutos. Destes, o melhor desempenho individual é obtido pelo LBP (*Local Binary Pattern*) [34, 35]. Devido às oclusões frequentes dos frutos e a haver ligeiras variações na sua forma, o processamento de informação relativa à forma não é a solução mais comum.

A comparação direta entre as metodologias adotadas pelos diferentes autores nem sempre é possível de realizar, porque usam métricas distintas, ou porque as condições em que foram realizados os ensaios (frutos e condições de luz) não são iguais. No entanto, na Tabela 2.1 é feito um resumo dos resultados alcançados por diferentes autores na deteção e segmentação de diferentes frutos.

Com base na Tabela 2.1 e nas restantes metodologias revistas, os classificadores baseados em métodos de aprendizagem supervisionada atingem melhor resposta do sistema. Quanto ao número de classes de elementos com que é classificado o cenário, este assume um valor superior a dois ( $C > 2$ ) quando é necessário identificar outros elementos que não os frutos para tarefas complementares, tal como, por exemplo, a deteção do pedúnculo da fruta.

Antevê-se que um sistema multiespectral possa permitir obter melhores respostas espectrais do cenário do que uma simples câmara a cores. Contudo, como se assume que o cenário é dinâmico, é impraticável recorrer a um sistema de roda de filtros e à câmara usada em [33-35]; adicionalmente, não é aplicável ao estudo focado nos limões, porque o sensor IR desta câmara não capta a radiação NIR no intervalo de frequências que, previsivelmente, melhor consegue distinguir os limões da folhagem, seguindo os estudos cujas respostas espectrais são apresentadas na Fig. 2.1.

A partir da análise precedente referente ao estado de arte torna-se possível evidenciar quais as metodologias que têm maior probabilidade de sucesso na segmentação e deteção dos limões, a partir de imagens adquiridas num ambiente de iluminação não controlada, tal como será discutido nos capítulos seguintes.

Tabela 2.1. Resultados alcançados por abordagens anteriores

Ref.	TPr	FPr	Sistema de Visão	Modelo de cor	Nº imagens	Fruto
[14]	<sup>†</sup> 90%	4%	Monocular Cores	chromatic idades	110	Citrinos
	<sup>‡</sup> 85%	5%			20	
[10]	<sup>†</sup> 91%	15%	Multiespectral	-	51	Citrinos
[24]	<sup>†</sup> 75.3%	27.3%	Monocular Cores	HSI, YCbCr	96	Citrinos verdes
[31]	<sup>†</sup> 89%	-	Monocular Cores	HSI	100	Maçãs
[12]	<sup>‡</sup> 68.3%	1.1%	Multiespectral	CIELab	6	Uvas
[17]	<sup>‡</sup> 96.6%	0.19%	Multiespectral + sensor de profundidade	RGB	12	Maçãs
	<sup>‡</sup> 83.1%	2.01%			10	Uvas

<sup>†</sup> Taxas relativas à detecção; <sup>‡</sup>Taxas relativas à segmentação.

### 3. MÉTODOS DE APRENDIZAGEM

A aprendizagem constitui um elemento relevante em toda a metodologia desenvolvida nesta dissertação e na resposta do sistema implementado. Em Inteligência Artificial (IA) a aprendizagem consiste em analisar um conjunto de variáveis de entrada e determinar um modelo de decisão que permita classificar corretamente cada sinal de entrada na(s) respectiva(s) classe(s).

Porém, para se obter um modelo de decisão que cumpra uma série de requisitos, como taxa mínima de acerto, taxa máxima de erros e tempo de processamento, requer-se um minucioso processo de preparação de uma base de dados, de atenuação de ruído, de seleção do método de classificação e das características mais significativas para a decisão, bem como a avaliação da resposta em função das parametrizações do método de classificação. Cada uma das etapas do processo de aprendizagem deve ser cuidadosamente planejada, pois, frequentemente, a quantidade de dados a processar implica quantidades de tempo avultadas para se obter uma resposta desejável; porém, a resposta obtida nem sempre é a prevista!

Depois de determinar o modelo de decisão mais adequado aos requisitos impostos, a tarefa de classificação consiste em aplicar o modelo de decisão às variáveis de entrada de modo a obter uma resposta, que poderá ser binária ( $C=2$ ) ou multiclases ( $C>2$ ). Opcionalmente, pode ser necessário pré-processar as variáveis de entrada de modo a obterem-se as características desejáveis, ou para cumprir com imposições no domínio destas, conforme esquematizado na Fig. 3.1.

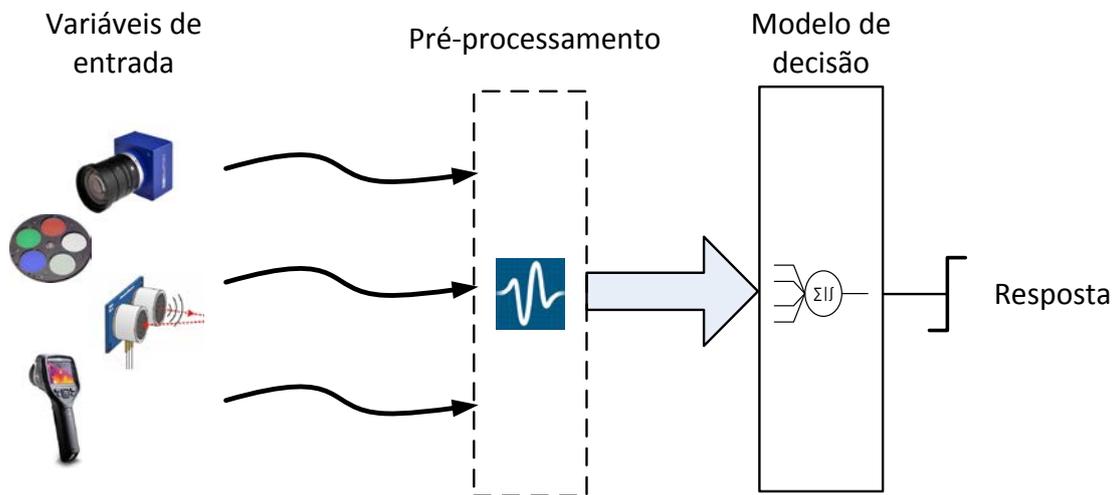


Fig. 3.1. Fluxo de sinal na classificação.

A aprendizagem é realizada considerando determinadas condições no sinal de entrada, pelo que é fundamental que estejam previstas, se não todas as variações, a sua maioria, para que o modelo possa responder continuamente de forma correta. Caso surjam alterações não previstas no sinal

de entrada, a resposta do sistema tanto pode continuar a ser correta como não; estará dependente de como foi realizada a aprendizagem e da própria variação provocada no sinal de entrada. Por exemplo, se no processo de aprendizagem de um classificador binário de limões não for prevista a ocorrência de tonalidades azuis ( $Hue \approx 240^\circ$ ), mas existir uma restrição que ignore todos os *pixels* cuja componente  $Hue \notin [0, 90^\circ]$  será indiferente a sua presença em futuros sinais de entrada, pois a tonalidade azul passa a ser classificada como um elemento não pertencente à classe dos limões. Contrariamente, se o sinal de entrada passar a conter elementos com tonalidades próximas dos limões, mas que não foram previstos durante o processo de aprendizagem, há a possibilidade de serem erradamente classificados como limões.

Para evitar alterações na resposta do modelo de decisão, motivadas por alterações no sinal de entrada, é essencial prever todas as variações expectáveis e incluí-las na base de dados de referência ao processo de aprendizagem, mas sem desvirtuar a proporção de ocorrências, caso contrário estar-se-á a desenvolver um modelo de decisão para os casos excepcionais.

As restantes secções deste capítulo abordam, de forma mais detalhada, as diferenças entre os dois métodos de aprendizagem, bem como sobre a forma de determinar quais as características a usar, de como repartir os dados disponíveis para treinar e avaliar o desempenho de um modelo de decisão e, por fim, sobre os diferentes modelos matemáticos para otimizar o processo de aprendizagem. Finalmente é apresentado o modelo de treino e validação do modelo de decisão.

### **3.1 Métodos de aprendizagem supervisionada e não supervisionada**

A classificação dos elementos de um cenário é um processo que envolve a procura de padrões que caracterizam cada classe nas variações previsíveis de ocorrência nos cenários captados. Enquanto os métodos de aprendizagem supervisionada necessitam de um conjunto de dados previamente classificados para inferir sobre um modelo de decisão que permita classificar corretamente os elementos nas respetivas classes, os métodos de aprendizagem não supervisionada não aceitam essa classificação realizada *a priori*, “limitando-se” a observar cada conjunto de dados e a agrupá-los por diferentes padrões; este processo é designado de *clustering*.

Os métodos de aprendizagem são usados para que o computador infira de forma autónoma sobre um modelo de decisão que, de acordo com esta abordagem, permita distinguir os frutos dos restantes elementos do cenário.

Nos métodos de aprendizagem supervisionada são definidos, inicialmente, as classes existentes durante o treino de um modelo de decisão, enquanto nos métodos de aprendizagem não supervisionada é definido o número de classes (padrões) que o modelo deverá encontrar nos dados a analisar. Quando usados na classificação, os métodos de aprendizagem supervisionada, por exigirem um conjunto de dados para treino do modelo de decisão, implicam o dispêndio de algum tempo na sua implementação.

Nos métodos de aprendizagem não supervisionada, como não é possível ensiná-los a aprender e a memorizar como é constituída cada classe, são forçados a encontrar o número de classes pré-definido em cada conjunto de dados, mesmo que determinada classe não exista. Já os métodos de aprendizagem supervisionada, se não encontrarem nenhum elemento que cumpra os requisitos aprendidos, para determinada classe para a qual foram treinados, não forçam a sua existência para esse conjunto de dados.

Seja num sistema de classificação binária ( $C=2$ ), ou num sistema multiclasse, os métodos de aprendizagem não supervisionada devem ser utilizados quando há a certeza de que o número pré-definido de classes existe sempre nos dados a analisar. Este método é particularmente útil quando existe uma grande quantidade de dados e se pretendem identificar grupos similares, ou seja, grupos que apresentem os mesmos padrões.

Foi identificada na bibliografia a aplicação de métodos de aprendizagem não supervisionada na segmentação de um cenário frutícola. Em [20] é utilizado o método de OTSU e o *Self-Organizing Map* na segmentação de laranjas; porém, os resultados alcançados ficaram aquém do método *Fisher Linear Discriminant* (aprendizagem supervisionada), também avaliado nesse estudo. Mas estes métodos, à exceção do método de OTSU, demoraram mais de sete segundos a classificar uma imagem, não os tornando apelativos para aplicações reais, independentemente das taxas de sucesso alcançadas e alcançáveis.

O desempenho de outro método de aprendizagem não supervisionada, o *K-means*, foi também avaliado na segmentação de um cenário vitivinícola [12]. Neste caso recorreu-se a vários classificadores binários aplicados a imagens adquiridas com diferentes filtros óticos, para determinar os elementos de cinco classes distintas. Apesar dos autores estarem satisfeitos com os resultados, tendo obtido uma taxa de TP = 68.3 % e de FP = 1.1 % na classificação da classe das uvas, um estudo posterior [17] viria também, neste caso, a demonstrar que a aplicação de métodos de aprendizagem supervisionada resultam num melhor desempenho.

Wachs *et al.*, [81], também avaliou o desempenho de uma análise baseada em *K-means* comparando-o com uma abordagem baseada num classificador de características de *Haar* na deteção de maçãs. A abordagem com *K-means* é a que regista pior desempenho, mas ambos os métodos resultam num elevado número de falsas deteções. Em [82] os autores também recorrem a *K-means* para segmentar o cenário em quatro classes ( $C>2$ ) com o intuito de, posteriormente, detetarem citrinos através da transformada de *Watershed*.

Em situações ideais teríamos as classes bem definidas, com as respetivas médias ( $\mu$ ) afastadas e, preferencialmente com desvio padrão ( $\sigma$ ) ou variância de baixo valor ( $\sigma^2$ ), ou seja, o sinal de cada classe encontrar-se concentrado, não existindo sobreposição das classes, configurando uma distribuição bimodal, conforme exemplificado na Fig. 3.2.

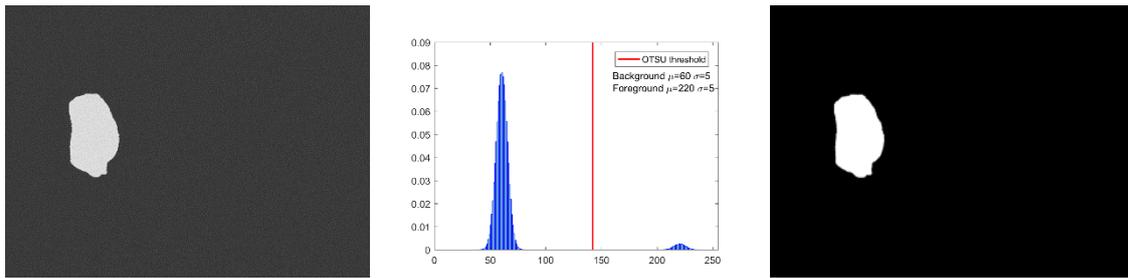


Fig. 3.2. (esquerda) Imagem criada com ruído Gaussiano; (centro) histograma da imagem; (direita) máscara resultante do método de OTSU.

Porém, em condições reais, nem sempre se consegue definir o limiar de cada classe e ambas se podem sobrepor. Na Fig. 3.3 encontram-se representadas imagens adquiridas com o filtro ótico BP635<sup>1</sup> numa plantação real, com os respetivos histogramas e a máscara resultante obtida a partir do limiar determinado pelo método de OTSU. As imagens desta figura foram segmentadas manualmente de forma a obter-se a máscara correspondente à classe dos limões. Assim, a partir dessa máscara foi determinada a média e o desvio padrão da classe dos limões (*foreground*) e do *background* da imagem real, e apresentados conjuntamente com o histograma. Para efeitos desta análise foram ainda desconsiderados os *pixels* saturados, isto é, cujo valor de intensidade é 0 (mínimo) ou 255 (máximo).

Após a análise da Fig. 3.3 verifica-se que em situações reais não existe, ou pode não existir, um vale bem definido entre as duas classes; as classes podem sobrepor-se e, tendencialmente, há um desequilíbrio no número de elementos das duas classes. Todos estes fatores influenciam o desempenho da segmentação através do método de OTSU.

<sup>1</sup> Filtro ótico de 635 nm do fabricante Midwest Optical Systems, Inc.: <http://midopt.com/filters/bp635/>

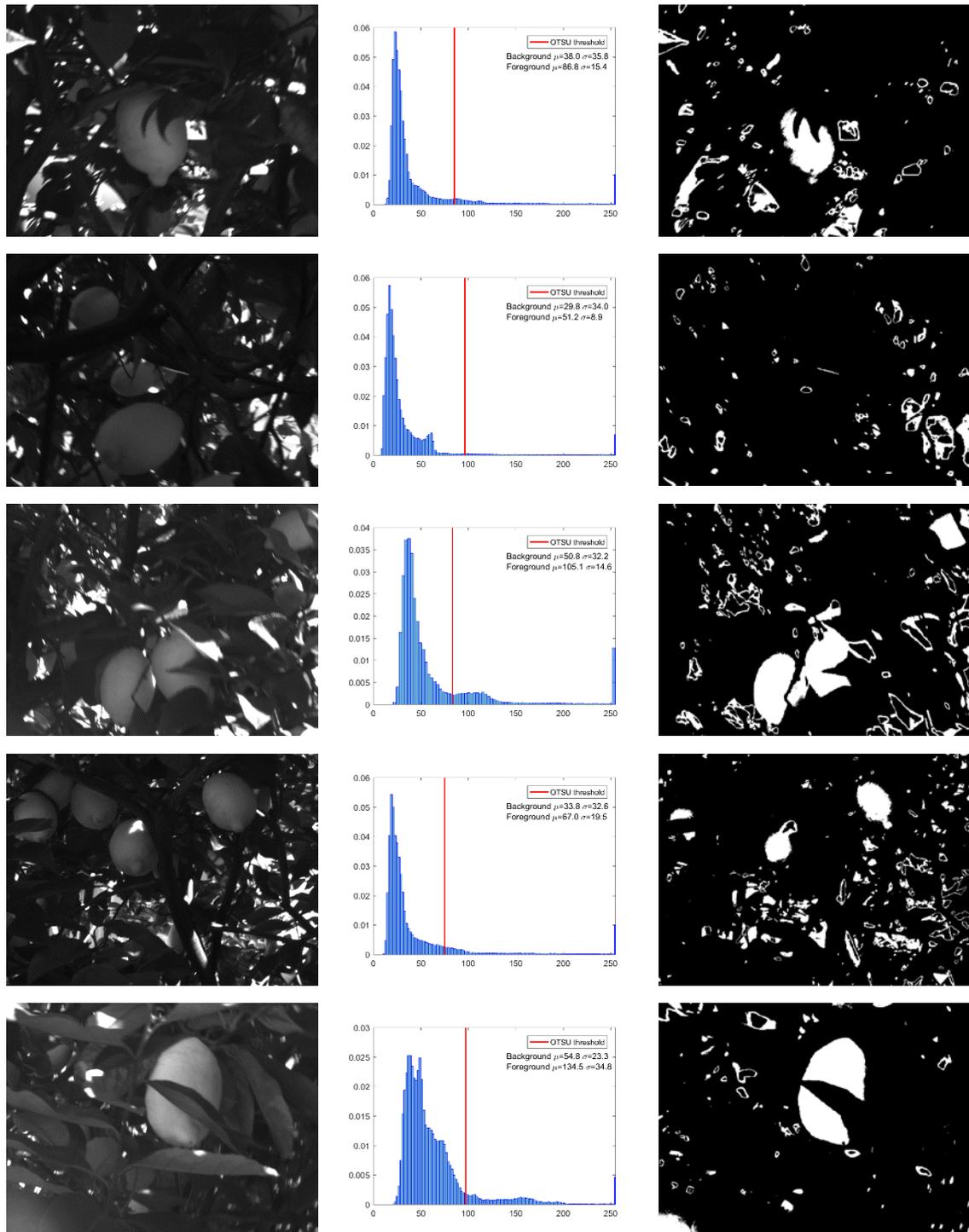


Fig. 3.3. (esquerda) Imagem em condições reais adquirida com filtro ótico 635nm; (centro) Histograma da imagem com estimativa da distribuição Gaussiana; (direita) Máscara resultante do método de OTSU.

O limiar (*threshold*) global determinado pelo método de OTSU consiste em minimizar a variabilidade da intensidade dentro das duas classes (para  $C=2$ ), de acordo com a equação seguinte:

$$\sigma_D^2(t) = \omega_0(t)\sigma_0^2(t) + \omega_1(t)\sigma_1^2(t) \quad (3.1)$$

de forma que,

$t$  representa o valor do limiar que separa as duas classes

$\sigma_0^2$  e  $\sigma_1^2$  representam a variância de cada classe

$\omega_0$  e  $\omega_1$  a probabilidade de um pixel pertencer a cada uma das classes determinadas pelas equações seguintes:

$$\omega_0(t) = \sum_{i=0}^{t-1} p(i) \quad (3.2)$$

$$\omega_1(t) = \sum_{i=t}^{L-1} p(i) \quad (3.3)$$

Assumindo que  $p$  representa o histograma de uma imagem de dimensões  $M \times N$  e com  $L$  níveis de cinzento, dado por:

$$p(i) = \frac{n(i)}{M \times N} \quad (3.4)$$

de tal forma que,

$$p(i) \geq 0 \wedge \sum_{i=0}^{L-1} p(i) = 1 \quad (3.5)$$

Para se obter o limiar ótimo tem que se determinar o respetivo valor para o qual se consegue minimizar a equação (3.1):

$$t^* = \min_{0 \leq t \leq L-1} \sigma_D^2(t) \quad (3.6)$$

Sendo as imagens de resolução constante, a população total da amostra será sempre a mesma; no entanto irá variar a população (número de *pixels*) de cada classe. Nas imagens captadas, havendo uma predominância populacional da classe do *background*, ou seja,  $\omega_0 > \omega_1$ , o limiar determinado pelo método de OTSU, para contrabalançar com esse desequilíbrio populacional, irá procurar que a classe dos limões possua maior variância,  $\sigma_1^2 > \sigma_0^2$ . Contudo é mais fácil interpretar a seleção do limiar ótimo a partir da variabilidade entre classes, conforme demonstrado na equação (3.7):

$$\begin{aligned} \sigma^2 = \sigma_E^2 + \sigma_D^2 &\Leftrightarrow \sigma_E^2 = \sigma^2 - \sigma_D^2 \Leftrightarrow \sigma_E^2 = \omega_0(\mu_0 - \mu)^2 + \omega_1(\mu_1 - \mu)^2 \\ \sigma_E^2 &= \omega_0\omega_1(\mu_0 - \mu_1)^2 \end{aligned} \quad (3.7)$$

em que a média total da amostra é dada por:

$$\mu = \omega_0\mu_0 + \omega_1\mu_1 \quad (3.8)$$

Sendo a variância ou variabilidade total da amostra composta pela soma da variabilidade entre classes ( $\sigma_E^2$ ) e da variabilidade da intensidade dentro das duas classes ( $\sigma_D^2$ ), ao minimizar a  $\sigma_D^2$  significa que terá que se maximizar  $\sigma_E^2$  para se obter o limiar ótimo ( $t^*$ ):

$$t^* = \max_{0 \leq t \leq L-1} \sigma_E^2(t) \quad (3.9)$$

Como a variância de cada classe é determinada a partir da sua média, não é tão intuitivo minimizar a variância da cada classe com menor população, equação (3.1), como é o caso da maximização da diferença das médias das classes, equação (3.7).

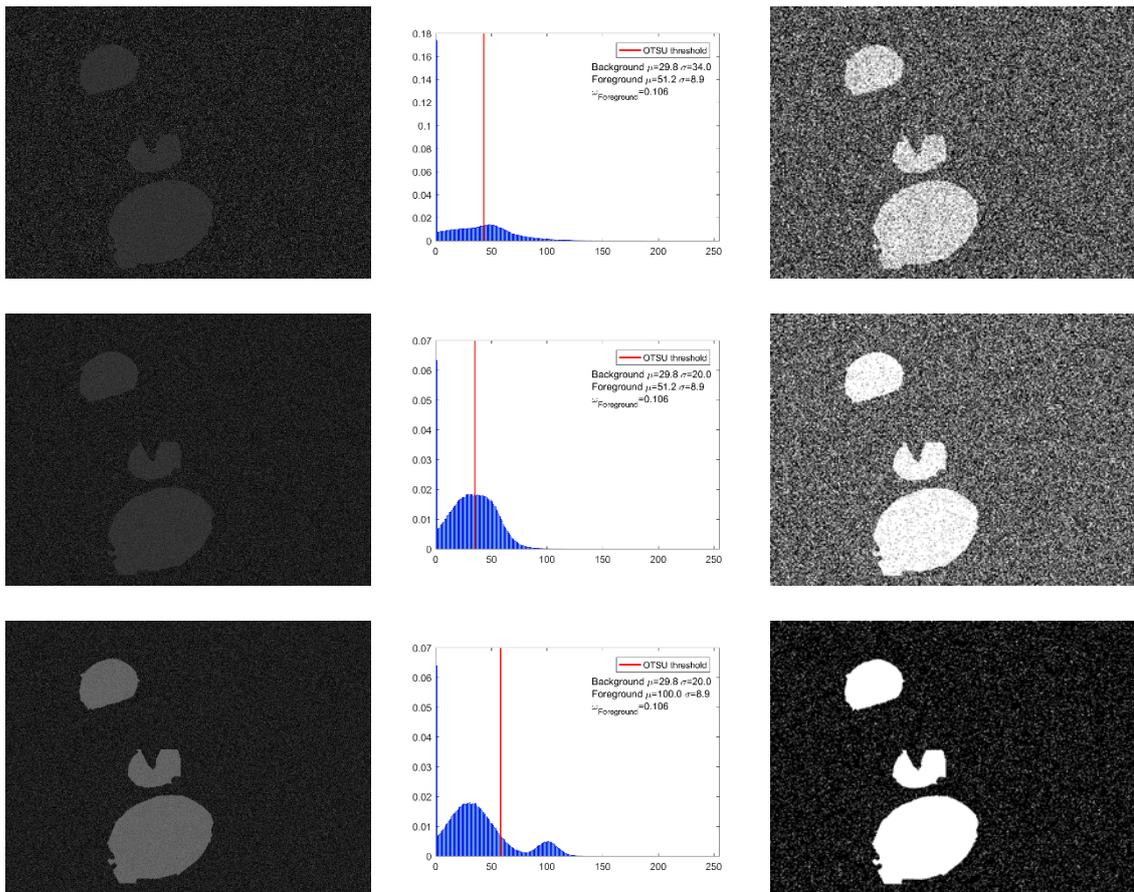


Fig. 3.4. (esquerda) Imagem simulada a partir da imagem real da segunda linha da Fig. 3.3; (centro) Histograma da imagem com parâmetros das distribuições Gaussianas; (direita) Máscara resultante do método de OTSU.

Para demonstrar os efeitos práticos de diferentes parametrizações da média e variância de cada classe, a Fig. 3.4 e a Fig. 3.5 representam imagens simuladas derivadas de imagens reais da Fig. 3.3 e respectivas máscaras determinadas manualmente. Para cada imagem é definida a distribuição normal dos elementos das duas classes; inicialmente é simulada uma imagem com as médias e variâncias estimadas e, nas linhas seguintes de cada figura, é alterado o desvio padrão ( $\sigma_0$ ) da

classe *background* e a média da classe dos limões ( $\mu_1$ ). É ainda indicada a porção de elementos da classe dos limões presente em cada imagem ( $\omega_{\text{Foreground}}$ ).

Na Fig. 3.4 os sinais de ambas as classes estão inicialmente sobrepostos e não existe uma distribuição total bimodal, pelo que é expectável uma resposta insatisfatória do método de OTSU. Ao minimizar a variância da classe *background* (no histograma é apresentado o valor do respetivo desvio padrão  $\sigma_0$ ) sem alterar a médias das classes - representações da segunda linha - origina uma distribuição global aproximada de uma distribuição normal. No entanto, essa distribuição é unimodal e resulta num pior desempenho do método de OTSU relativamente à situação inicial; o limiar ótimo determinado assume um valor inferior. Afastando a média e mantendo os restantes parâmetros constantes dá origem, neste caso, a uma distribuição bimodal e a uma melhor segmentação pela aplicação do método de OTSU; porém há ainda uma elevada ocorrência de FP.

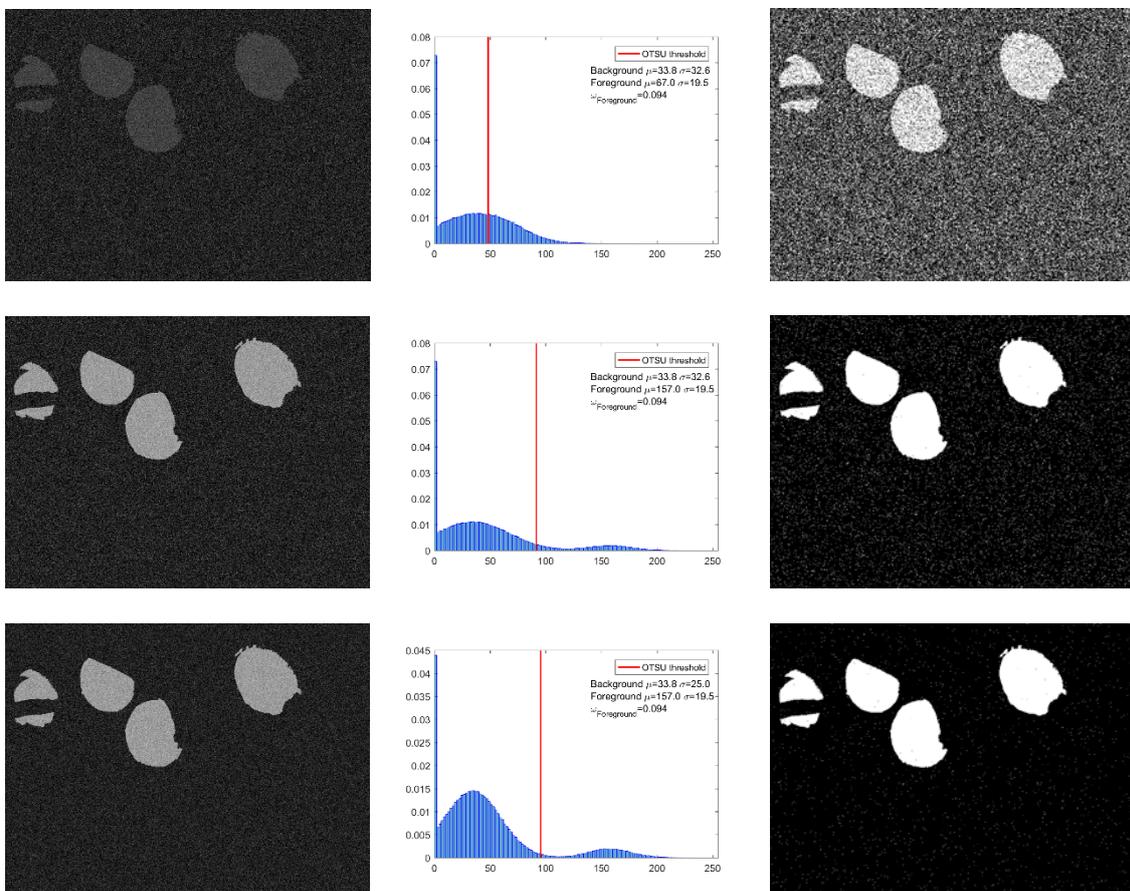


Fig. 3.5. (esquerda) Imagem simulada a partir da imagem real da quarta linha da Fig. 3.3; (centro) Histograma da imagem com parâmetros das distribuições Gaussianas; (direita) Máscara resultante do método de OTSU.

A Fig. 3.5 contém uma simulação análoga, mas para uma imagem diferente, onde primeiro se maximiza a distância entre as médias e depois se minimiza a variância dentro da classe do

*background*. O afastamento da média das distribuições normais das duas classes melhora significativamente a resposta do método de OTSU. Por sua vez, a minimização da variância da classe do *background* tem um impacto menor no desempenho do método de OTSU.

Na prática (Fig. 3.3) há a sobreposição das duas classes (ruído) e a distribuição global pode não assumir o formato de uma distribuição bimodal. Também o ambiente de aplicação em questão não permite garantir condições de luz constantes na própria imagem, nem entre imagens, tornando sensível a segmentação dos cenários captados, recorrendo unicamente a uma imagem, com o método de OTSU no presente contexto. O insucesso do método de OTSU na detecção e segmentação de laranjas também foi experienciado em [14].

A aplicação do método de OTSU terá sucesso em ambientes de iluminação controlada e ou com a utilização de filtros óticos de forma a assegurar que um conjunto de fatores, como o rácio da população de cada classe, as suas variâncias e médias, permanecem estáveis e dentro das condições de sucesso deste método.

Nos casos em que a iluminação não é homogénea em todo o cenário uma abordagem com vários limiares determinados localmente pode resultar num melhor desempenho que o limiar ótimo global, isto é, limiar único para toda a imagem.

Métodos de aprendizagem supervisionada são usados para que o computador aprenda com casos conhecidos e possa determinar um modelo de decisão para inferir sobre a classe de casos futuros. Na literatura revista é identificado o uso de SVM [17, 31, 83, 84], NN [10, 85] e KNN (*k-nearest neighbors*) [86, 87].

Em [25] é apresentado um estudo comparativo de vários métodos onde o KNN, NN e SVM demonstraram ser os de melhor desempenho na detecção de pêssegos em estado de maturação.

Uma comparação entre os métodos *Fischer's linear discriminant*, *Bayesian Classifier* e NN é apresentada em [88] para segmentação das imagens e determinação do tamanho de citrinos. Dos três métodos avaliados o melhor desempenho foi obtido por NN.

Quando se recorre a classificadores binários, no caso do SVM o cenário (ou a população da amostra) é composto por mais do que duas classes, e é pretendido classificar cada elemento na sua respetiva classe, pode recorrer-se a duas estratégias distintas:

- *One-vs-all* (um contra todos, OVA) - durante o treino haverá apenas duas classes de elementos, em que uma é a pretendida e a outra a união das restantes classes. No final é atribuído ao elemento a classe que resultou numa maior confiança;
- *One-vs-one* (um contra um, OVO) - são criados vários classificadores binários com combinações de apenas duas classes, num total de  $C(C - 1)/2$ , em que  $C$  representa o número de classes. A decisão final da classe a que pertence o elemento avaliado é obtida por maioria de voto, isto é, de todos os classificadores a classe que foi atribuída a esse elemento mais vezes é a classe a que pertence.

Um abordagem multiclasse ( $C > 2$ ) para a classificação de cenários com citrinos é realizada em [83], onde os autores classificam os elementos presentes nas imagens em quatro classes distintas seguindo a estratégia OVO. Recorrendo a classificadores SVM com parâmetros fixos, de um total de 592 visualmente identificados em 83 imagens, foram corretamente identificados 92.4 % dos frutos.

Uma abordagem diferente, seguindo a estratégia OVA com uma ligeira variação, é proposta em [17]. Neste caso, conforme os elementos são atribuídos a uma determinada classe são excluídos de análises posteriores, ou seja, cada classificador terá uma população total diferente. Nestes casos é necessária alguma atenção com o possível desequilíbrio provocado entre o número de elementos de cada classe no treino dos classificadores.

Apesar dos resultados apresentarem tradicionalmente pior desempenho com os métodos de aprendizagem não supervisionada, quando não existe quantidade de dados suficiente para treinar devidamente um modelo de decisão, é recomendável procurar um método de aprendizagem não supervisionada que permita distinguir a população nas várias classes.

Os métodos de aprendizagem não supervisionada são também uma escolha viável na presença de características que permitam facilmente distinguir as várias classes e assim permaneçam com variações das condições ambiente, ou sempre que se consiga controlar e manter estáveis as condições de luz nos cenários a captar.

Alternativamente aos métodos apresentados para classificação dos cenários captados, pode-se recorrer a redes neuronais convolucionais (*Convolutional Neural Networks – CNN*) onde, nas primeiras camadas, são tipicamente aplicados filtros sobre a camada anterior (camada convolucional ou *convolutional layer*) e realizada a redução do espaço (*subsampling*). Na(s) última(s) camada(s) existem ligações entre todos os elementos para decidir a que classe pertence a amostra de entrada, Fig. 3.6. O número de camadas destas redes não é fixo.

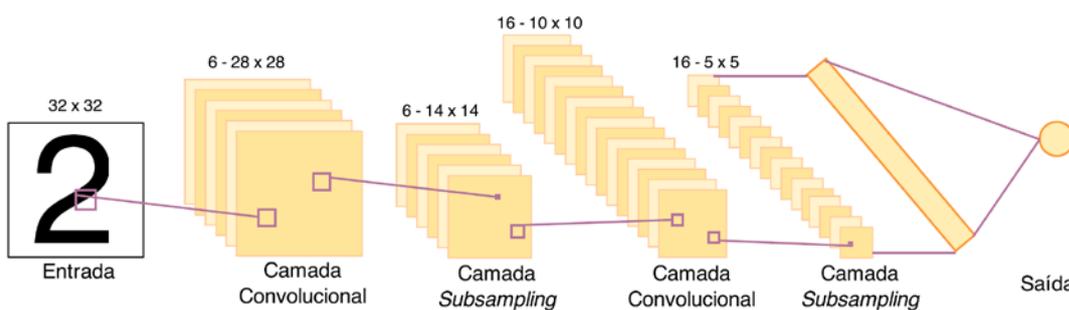


Fig. 3.6. Arquitetura da CNN LeNet-5 [89].

Com CNN pode usar-se uma rede já treinada para um determinado caso e (re)treinar apenas as últimas camadas da rede para que responda de forma correta aos novos casos. Para tal, devido à exigência de cálculo computacional, estes métodos têm que ser treinados em centros de

computação de elevado desempenho ou em computadores regulares com placas gráficas de elevado desempenho, pois exigem grande capacidade de memória disponível. Porém, depois de treinada uma rede CNN, a mesma pode ser utilizada para classificação num computador regular.

As redes CNN funcionam para imagens com dimensões definidas pelos seus autores que têm que ser respeitadas, pois foram desenvolvidas para a classificação de uma imagem completa. Porém, na detecção de objetos estes podem ser afetados de diversos fatores de escala, interessando saber em que região da imagem se encontra o objeto pretendido. Em [90, 91] são apresentados dois modelos com as modificações necessárias para usar o método CNN na detecção de objetos.

Em [90] são apresentadas recomendações para o treino de CNN, de acordo com as observações dos resultados dos testes realizados, entre as quais se destaca que a rede pode ser treinada com amostras de apenas uma determinada dimensão e pode detetar objetos com diversos fatores de escala, onde o treino com amostras de diferentes escalas não teve impacto significativo nos resultados finais.

Na avaliação realizada por Sa *et al.*, [77], na detecção de frutos com dados multiespectrais, há um melhor desempenho com CNN do que com o método CRF. Os autores verificaram que as cores permitem atingir melhores resultados do que através de uma rede CNN para análise de imagens compostas por imagens RGB e IR. No entanto, a resposta é melhorada se cada imagem, RGB e IR, for analisada individualmente e, no final forem combinadas as respostas das duas análises individuais. Relativamente à escala dos alvos, os autores referem que, se houver objetos muito mais pequenos do que os existentes durante o treino da rede, não é possível detetá-los corretamente.

Além disso, para treinar uma rede CNN para detecção de objetos, basta criar uma base de dados com ROI em torno dos alvos ao invés de selecionar manualmente cada *pixel*. Este simples procedimento permite uma criação mais rápida da base de dados de referência.

A partir das metodologias revistas foi escolhido o método SVM, enquanto método de aprendizagem supervisionado para a segmentação dos limões dos restantes elementos do cenário, sem necessidade de classificar os restantes elementos nas suas classes específicas, mas apenas classificá-los como *background*. O método de decisão é inspirado na abordagem sugerida em [17], mas apenas recorrendo a imagens a cores sem a componente IR. Assim será construída uma árvore de decisão com classificadores SVM seguindo uma estratégia OVO incompleta, ou seja, à medida que os *pixels* são classificados como *background* são excluídos de análises posteriores, onde cada classificador é treinado com amostras dos limões contra amostras de subclasses específicas do *background*, pois, no final da segmentação, interessa apenas existir a classe dos limões e a dos restantes elementos, como também é sugerido em [17]. No entanto, são também avaliados os resultados de uma estratégia OVA e comparada com os restantes casos avaliados.

## 3.2 Seleção de características

A seleção das características (*features*) que o SVM irá analisar é um procedimento relevante com impacto direto no desempenho do classificador. Ao usarem-se características irrelevantes na classificação de duas classes irá sobrecarregar-se o processo de treino do classificador, o que poderá originar um maior tempo de processamento e um pior desempenho do classificador [92].

Caso o modelo de decisão seja alimentado com características redundantes haverá também sobrecarga durante o treino; contudo, esta pode ser uma mais-valia quando existe muito ruído nas amostras, ao reforçar as condições de sucesso na separação das classes [93].

Em [94] são apresentados métodos para classificação de variáveis para sistemas de classificação binária, assim como casos particulares em que, através de análises estatísticas, se consideram certas combinações de variáveis irrelevantes quando, na realidade, representam uma boa classificação e vice-versa.

A Fig. 3.7 contém histogramas com três características diferentes, passíveis de serem utilizadas pelo modelo de decisão. Pela análise isolada dos seus histogramas verifica-se que as duas características representadas mais à esquerda (*b* e *indexA*) não representam, individualmente, uma solução a ser analisada pelo SVM, pois há uma sobreposição significativa das amostras das duas classes. Já o *indexY* isolado permitiria obter um classificador com bom desempenho.

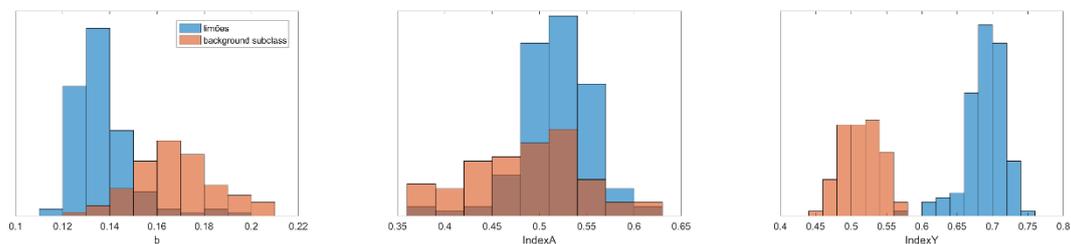


Fig. 3.7. Histograma de características para segmentação de limões.

No entanto, ao combinar especificamente essas duas características, aparentemente irrelevantes, consegue-se separar as duas classes presentes na amostra (Fig. 3.8). Ainda neste caso específico é possível aumentar a distância entre as duas classes ao combinar o *indexA* (característica individualmente irrelevante) com o *indexY*, dando origem a uma classificação mais robusta na presença de ruído.

Nem sempre um maior número de características a analisar é sinónimo de um melhor desempenho do classificador [92]. Tal como já foi referido, características irrelevantes não beneficiam o treino do classificador. Na Fig. 7.5 está representado um caso concreto de ensaios realizados em que um maior número de características origina um pior desempenho.

Na bibliografia revista é identificado um ensaio [10] onde é comparado o desempenho de um classificador baseado em NN, usando como características as imagens obtidas com seis filtros

óticos distintos, com outro classificador que recorre apenas às três características com maior significado determinado por uma análise PCA. Também aqui, o melhor desempenho (taxa TP mais elevada e FP menor) é obtido com as três características mais significativas, quando comparado com a análise de todas as características disponíveis.

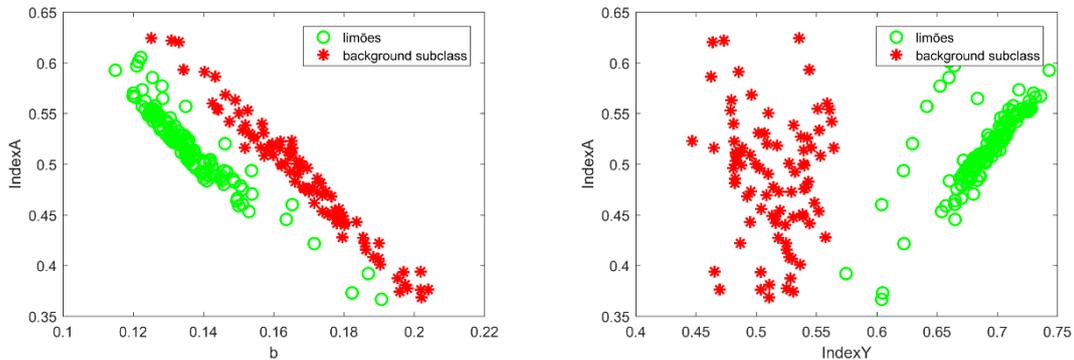


Fig. 3.8. Distribuição da combinação das duas características da Fig. 3.7.

No presente contexto, pretende-se obter um classificador com baixa latência e é expectável, previamente, que nem todas as características sejam úteis na segmentação dos limões da restante envolvente.

Assim, são iniciados ensaios com apenas duas características, sendo analisada a resposta do sistema com diferentes parametrizações do SVM. A partir dos melhores resultados alcançados é iniciada uma nova ronda de testes, adicionando uma característica relativamente à ronda anterior, conforme é descrito na seção 4.1 desta dissertação.

É um processo que é iniciado numa análise do tipo força-bruta, conhecido por não ser eficiente em termos de tempo de computação, mas que garante que não é excluída uma combinação com bom desempenho, composta por duas características isoladamente irrelevantes. Ao longo das várias rondas de treino, poderão ser excluídas algumas características em função do desempenho das combinações onde se inserem.

### 3.3 Treino e validação do modelo de decisão

Os modelos de aprendizagem supervisionada são treinados e validados a partir de uma base de dados de referência. A classificação de elementos presentes em imagens pela análise das condições do *pixel* implica uma quantidade enorme de dados a processar e, na fase do treino de um classificador, representa uma enorme quantidade de dados a analisar para a determinação da solução do problema. Por um lado, é vantajoso ter uma enorme quantidade de dados, pois estarão presentes na amostra de referência diversas condições possíveis para os elementos de várias classes, por outro lado haverá muitos dados redundantes e, havendo sobreposição parcial das

duas classes, forçará o sistema a avaliar o peso das más classificações de cada elemento até determinar uma solução para o classificador.

Devido ao facto da base de dados deste estudo ser proveniente de condições reais, cujas imagens foram adquiridas em ambientes considerados caóticos, onde a iluminação não é controlada nem os elementos são adquiridos em condições uniformes, como é o caso de limões presentes na mesma imagem localizados no interior da copa, sujeitos ao efeito da sombra, e no exterior da copa expostos diretamente à luz solar. Nestas condições é expectável que elementos de classes diferentes apresentem a mesma resposta com determinadas características, nomeadamente a resposta isolada de componentes das cromaticidades *rgb* em condições de baixa luminosidade.

A separação de um conjunto de dados para treino e validação (ou ainda para teste) do modelo de decisão, deve ser realizada garantindo que é mantida a proporção das várias condições em todos os subconjuntos [94]. Contudo, estes métodos não eliminam o ruído existente, apenas separam a população original em vários subconjuntos.

Um método popular para a divisão dos dados de referência é designado por *k-fold*, que consiste em criar subconjuntos a partir de um conjunto de dados destinado ao treino e validação e, com estes subconjuntos é realizada uma análise cruzada, ou seja, são treinados vários classificadores como parte desses subconjuntos e testados nos restantes. Um breve resumo das metodologias e cuidados a ter na preparação dos dados para treino e validação de um classificador são apresentados em [94].

Para diminuir a influência do ruído no treino dos classificadores SVM, e para diminuir a população a processar durante o treino, é proposta uma nova metodologia que consiste em atribuir uma população destinada à validação do classificador e, a partir deste ser determinado a população com que é realizado o treino de cada classificador, onde cada elemento da população destinada ao treino é obtido pela média da população de cada ROI.

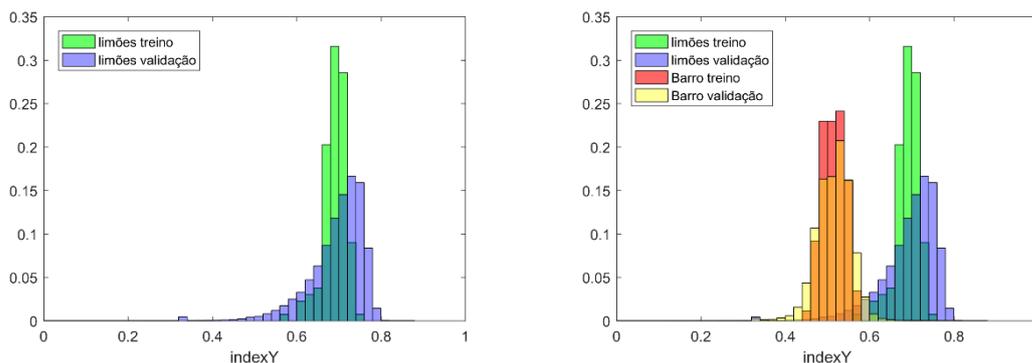


Fig. 3.9. Histograma das frequências relativas: (esquerda) elementos para treino e validação da classe dos limões; (direita) são acrescentados os elementos de uma subclasse do background.

Com esta metodologia, as partes menos expressivas serão excluídas do processo de treino do classificador. Mas também são essas partes que representam as situações dúbias, originadas pela iluminação não uniforme em todo o cenário captado.

A Fig. 3.9 representa a distribuição completa e real da população destinada ao processo de validação, e a população obtida para o treino. Por se assumir que as características de validação adotam uma distribuição normal e, neste caso específico, isto não se verificar, a distribuição da população de treino pode assumir um máximo e um valor médio afetados de um desvio em relação à distribuição real. Juntamente com a restante parametrização do classificador SVM é possível compensar esses desvios pela atribuição de penalizações de má classificação diferenciados para cada classe e, em simultâneo, com o ganho significativo de um menor número de *pixels* a processar.

De acordo com a imagem à direita na Fig. 3.9, verifica-se que distribuições de treino e validação dos elementos pertencentes à classe barro não sofrem desvios tão significativos quanto os limões. Além disso, a solução dada, usando esta metodologia, não é tão afetada, como as distribuições reais onde ocorre sobreposição parcial das classes.

Quanto ao teste do modelo de decisão é apenas avaliado o desempenho da sua resposta final, isto é, se a classe atribuída a cada *pixel* é corretamente atribuída, como pertencente à classe dos limões ou ao *background*.

Uma vez que o modelo de classificação segue uma estrutura em árvore, em cada nó é apenas possível avaliar o desempenho da classificação referente aos dados de teste. Como os *pixels* pertencentes à classe dos limões foram exaustivamente marcados, toda a área remanescente é parte do *background*. Assim, durante o teste do classificador, as regiões relativas aos limões são as mesmas da fase de validação, mas com expansão do número de elementos da classe complementar.

Os desequilíbrios nas amostras podem ter um forte impacto no treino dos classificadores e respetivo desempenho. Um dos primeiros cuidados a ter está relacionado com o equilíbrio entre classes, ou seja, a proporção entre o número de elementos (instâncias) das duas classes deverá ser preferencialmente idêntica. Este fator é designado em inglês por *imbalanced/unblanaced data*.

Por diversos motivos, a amostra inicial pode não promover o equilíbrio desejado entre o número de elementos de ambas as classes a analisar, sendo necessário a adoção de medidas adicionais. Na impossibilidade de obtenção de mais dados para a classe em menor número, pode recorrer-se à criação de instâncias artificiais, e à aplicação de metodologias para a seleção dos elementos para o treino do classificador, ou ainda compensar estes desequilíbrios na parametrização do classificador. Estas são algumas das metodologias referenciadas em [95, 96].

No caso de uma base de dados para tarefas, como a deteção de objetos, podem ainda gerar-se novas amostras pela rotação e aplicação de um fator de escala às amostras existentes.

Os métodos de aprendizagem também são sensíveis ao intervalo de variação dos valores de cada característica. Para uma dada combinação de características, se as amostras de uma dessas características assumir valores entre 0 e 1, e a outra variar entre 0 e 255, durante o treino do classificador, a variável com um intervalo de variação menor será menos significativa do que a outra, devido a operações matemáticas realizadas com o produto interno de vetores [97].

A normalização dos dados pode ser feita de diferentes formas, sendo as mais comuns as seguintes: limitar o domínio de todas as características a um intervalo específico; em função do desvio padrão da população de cada característica. A primeira opção implica que o ruído presente na amostra vá influenciar o fator de escala da normalização dos dados, enquanto que, na segunda opção essa condição é atenuada, não sendo possível definir um domínio fixo para toda a população.

O fator de escala para a normalização das características a analisar é determinado a partir dos dados destinados ao treino do classificador, devendo estes ser guardados para posterior aplicação dos mesmos fatores de escala aos dados de validação, teste e, caso o modelo seja aprovado, a todos os dados a analisar por esse classificador.

Nos ensaios realizados optou-se por normalizar os dados de modo a obter-se um desvio padrão unitário, individualmente em todas as características a analisar.

## 4. CLASSIFICAÇÃO POR SVM

Neste capítulo é apresentado o processo de aplicação do método SVM, método de aprendizagem supervisionada, enquanto classificador dos *pixels* de imagens. Nas seções seguintes são apresentadas as estratégias adotadas para otimização do processo de treino do classificador e apresentado o centro de computação de elevado desempenho (HPC – *High-Performance Computing*) utilizado, e a forma como devem ser elaborados os algoritmos para se obter a máxima vantagem do recurso a centros de computação avançada.

O SVM, enquanto método de classificação, é utilizado para avaliar um conjunto de características e determinar a classe a que pertencem. Apesar do método ter sido desenvolvido para realizar classificações binárias ( $C=2$ ) é possível encontrar aplicações do método estendidas a abordagens multiclasse ( $C>2$ ).

O treino de um classificador SVM, assim como de outros métodos de aprendizagem supervisionada, requer um conjunto de dados de referência ( $\mathbf{T}$ ) para que, a partir do conjunto de características ( $\vec{x}_i$ ) de cada instância, possa ser determinado o modelo que melhor classifica estas nas suas respetivas classes ( $y_i$ ):

$$\mathbf{T} = \{(\vec{x}_1, y_1), \dots, (\vec{x}_n, y_n)\} \quad (4.1)$$

Onde,

$$\vec{x}_i \in \mathbb{R}^m;$$

$$y_i \in \{-1, 1\}.$$

Para o caso da classificação binária, considerando  $n$  instâncias definidas por um conjunto de  $m$  características reais.

A partir de um conjunto de dados de treino ( $\mathbf{T}$ ) existem várias possíveis soluções que permitem classificar esses elementos nas respetivas classes (Fig. 4.1, esquerda). No entanto, a solução determinada pelo SVM é aquela que garante a maior distância (margem) da fronteira (hiperplano) imaginária aos elementos de cada classe; porém, nem todas as instâncias de treino são necessariamente representativas dos vetores de suporte da solução (Fig. 4.1, direita).

Esse hiperplano é definido por:

$$\langle w, x \rangle + b = 0 \quad (4.2)$$

de tal forma que seja possível separar corretamente todas as instâncias  $\mathbf{T}$  nas respetivas classes  $y_i$ , onde,

$$w \in \mathbb{R}^m;$$

$$b \in \mathbb{R} \text{ representa o viés (} \textit{bias} \text{)}.$$

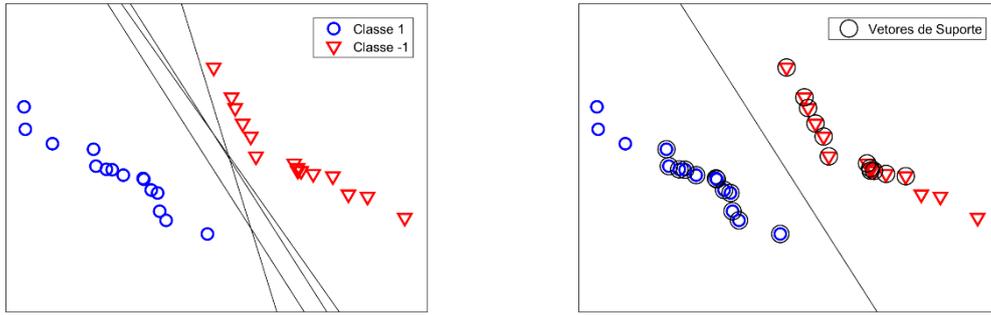


Fig. 4.1. Classificação binário com SVM linear: (esquerda) diferentes hiperplanos possíveis; (direita) hiperplano correspondente à margem máxima entre as duas classes e respectivos vetores de suporte assinalados.

Assim, cada instância  $\vec{x}_i$  é classificada segundo a fórmula:

$$f(\vec{x}) = \text{sgn}(\langle w, x \rangle + b) \quad (4.3)$$

Seguindo as considerações e a notação apresentada em [98], o SVM trata de determinar o hiperplano que maximiza a margem  $\frac{2}{\|w\|}$  entre as duas classes, ou seja, a solução ótima que consiste em minimizar  $\frac{1}{2} \|w\|^2$ , de acordo com a equação (4.4).

$$\min \frac{1}{2} \|w\|^2 + C \sum_i \xi_i, \forall i: y_i(\langle w, x \rangle + b) \geq 1 - \xi_i \quad (4.4)$$

A equação (4.4) corresponde à solução para um problema em que os dados de treino contêm instâncias que tornam o problema não linearmente separável, motivo pelo qual são atribuídas variáveis de desvio  $\xi$  (do inglês, *slack variables*), às quais é associada uma penalização  $C$ . Quanto maior for o valor atribuído ao parâmetro  $C$ , maior o impacto das más classificações na solução a determinar pelo sistema. De referir ainda que, normalmente, valores de  $C$  mais elevados estão relacionados com o treino mais demorado do classificador. De acordo com a nomenclatura adotada pelo *software* utilizado, o parâmetro  $C$  é designado por *box constraint*.

Pelos motivos enunciados em [98], a otimização da margem não é realizada pela equação (4.4) mas sim por:

$$\max \sum_i \alpha_i - \frac{1}{2} \sum_i \sum_j \alpha_i \alpha_j y_i y_j \langle x_i, x_j \rangle \quad (4.5)$$

em que  $\alpha_i$  corresponde aos multiplicador de Lagrange.

A Fig. 4.2 representa soluções de classificadores SVM treinados com o mesmo conjunto de instâncias, mas para diferentes valores de  $C$ . Para o presente caso, qualquer valor atribuído ao parâmetro  $C$  superior a 10000 produzirá o mesmo efeito que  $C = 10000$ , significando que  $\alpha_i$

proveniente da equação (4.5) permanece constante nesses casos. Para os restante casos representados nessa figura,  $\alpha_i$  é limitado pela definição de  $C$  aquando do treino do classificador.

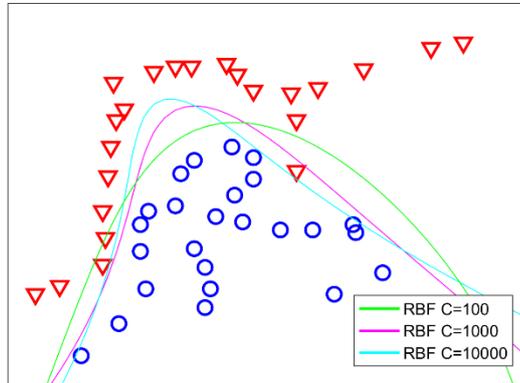


Fig. 4.2. Classificadores SVM para diferentes valores de  $C$ .

Porém, nem todos os problemas são linearmente separáveis e as soluções do hiperplano até agora apresentadas deixam de ser eficazes. Nestes casos, a solução passa por alterar o espaço original pelo mapeamento dos dados de treino para um novo espaço amostral de dimensões superiores (equação (4.6)), de forma a que se tornem linearmente separáveis onde, em vez de se calcular o produto interno de  $\langle x_i, x_j \rangle$ , é determinado o produto interno das instâncias no espaço  $\mathcal{H}$ , por  $\langle \Phi(x_i), \Phi(x_j) \rangle$ .

$$\Phi: \mathbb{R}^m \mapsto \mathcal{H} \quad (4.6)$$

Essas transformações no espaço são então definidas por funções de núcleo (do inglês, *kernel functions*). Na presente dissertação foi selecionada a função de núcleo de base radial, abreviada por RBF, com origem na designação inglesa *Radial Basis Function*, expressa da seguinte forma:

$$k(x_i, x_j) = e^{-\frac{\|x_i - x_j\|^2}{\sigma^2}}, \sigma > 0 \quad (4.7)$$

onde  $\sigma$  corresponde ao parâmetro da função de núcleo RBF, e designado no *software* utilizador por *kernel scale*.

Desta forma, a equação (4.5) tem que ser reescrita da seguinte forma:

$$\max \sum_i \alpha_i - \frac{1}{2} \sum_i \sum_j \alpha_i \alpha_j y_i y_j k(x_i, x_j), \sum_i \alpha_i y_i = 0 \wedge \forall i: 0 \leq \alpha_i \leq C \quad (4.8)$$

Também a equação (4.3), que anteriormente era usada para determinar a classe de cada instância, é reescrita, sendo agora representada por:

$$f(z) = \text{sgn}(\sum_i \alpha_i y_i k(x_i, z) + b) \quad (4.9)$$

Além dos parâmetros  $\sigma$  e  $C$  são ainda controlados, durante o treino dos classificadores SVM, a percentagem de ruído de toda a amostra e o custo relativo das más classificações ( $c$ ), sendo designados, no *software* utilizado, por *outlier fraction* e *cost*, respetivamente.

O peso atribuído a uma má classificação é, inicialmente, definido por  $C$ , mas, em determinadas situações, pode interessar atribuir uma penalização diferente a um elemento da classe dos limões, classificando como *background* relativamente à situação inversa. Esta diferenciação pode ser feita para forçar o modelo de decisão a classificar corretamente uma determinada classe, surgindo também uma maior probabilidade de elementos da outra classe serem incorretamente classificados. Adicionalmente, quando há um desequilíbrio acentuado no número de instâncias de treino das duas classes, o valor de  $c$  serve para maximizar a importância dos elementos da classe com menor população.

A forma como  $c$  influencia a penalização das más classificações está representada na Fig. 4.3. Caso seja atribuído a  $c$  o valor 1, ambas as classes são afetadas pela mesma penalização: se  $c < 1$  é atribuída uma penalização menor aos elementos da classe dos limões classificados como *background*, do que na situação inversa; se  $c > 1$  é dada mais importância à correta classificação dos limões, sendo atribuída uma penalização maior aos elementos da classe dos limões classificados como *background*.

		Classe Prevista	
		1	-1
Classe Real	1	0	$c$
	-1	1	0

Fig. 4.3. Penalização relativa de más classificações pelo parâmetro  $c$ .

A segmentação dos limões dos restantes elementos da imagem pode ser feita pelo treino de um classificador único. Porém, são facilmente identificáveis algumas (sub)classes de elementos nas imagens pelas propriedades idênticas que partilham, como é o caso da folhagem que apresenta uma tonalidade verde ou os ramos com propriedades distintas dos limões e da folhagem.

Sendo fácil identificar estas e outras classes de elementos nas imagens, também poderá haver um melhor desempenho através do treino de classificadores que separem os limões de classes específicas, quando comparado com o uso de um único classificador global. Assim, é proposto abordar a segmentação dos limões nas imagens a partir de uma árvore de decisão, com uma estrutura idêntica à ilustrada na Fig. 4.4, onde, em cada nível, são eliminados elementos com características semelhantes entre si, mas em oposição aos limões: trata-se de decompor uma classificação complexa em subclassificações mais simples.

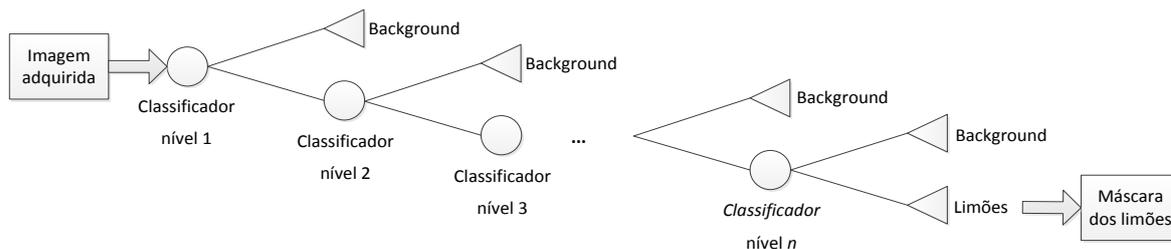


Fig. 4.4. Esquema representativo da estrutura da árvore de decisão.

Para avaliar se um *pixel* pertence, ou não, à classe dos limões, os classificadores SVM são treinados com as componentes dos modelos de cor e com os índices de cor, mencionados na secção 6.2. Com estas características são criadas diferentes combinações de vários elementos a serem usadas pelos classificadores SVM.

Conforme os elementos de imagem são classificados como *background*, num determinado nível da árvore de decisão, são excluídos de análises posteriores. Nesta situação, se houver igualdade de instâncias de treino de duas classes, pretende-se que  $c > 1$ , pois, devido às avaliações consecutivas a que as instâncias dos limões estão sujeitas no processo de decisão em árvore, visa-se evitar a sua má classificação massiva e dar origem a taxas elevadas de FN (Falsos-Negativos).

## 4.1 Métodos de otimização

Durante o treino do classificador há dois conjuntos de dados de entrada que são manipulados pela ação externa do operador, com influência direta na solução determinada, que são: a combinação de características a analisar; e as parametrizações do classificador SVM.

Uma vez que não está definido o número de características a analisar, nem as parametrizações do classificador SVM, é usada a configuração que apresentar a taxa TP e FP mais adequada. Porém, o número de combinações possíveis com diferentes parametrizações do SVM e com combinações de características é elevado, pelo que importa otimizar o processo de treino na busca do classificador que permite obter o melhor desempenho.

Quanto ao número de características a analisar e às suas combinações é assumido o desafio de proceder à segmentação do cenário recorrendo a poucas características das 11 componentes de cor disponíveis para análise. Os testes de *força-bruta* permitem obter a combinação ótima, mas, em contrapartida, demoram mais tempo a ser processados do que os anteriormente referidos.

Com recurso a um HPC com 480 núcleos de processamento, torna-se exequível, no presente contexto, testar exaustivamente certas configurações no treino dos classificadores, mas não todas. Deste modo, é proposta uma avaliação do desempenho dos classificadores por gerações, em que a cada uma destas é considerada uma característica adicional, relativamente à geração anterior.

A computação evolucionária tem como base a evolução biológica dos seres vivos, em que cada nova geração é influenciada pelos seus progenitores, isto é, possui parte das características dos seus ascendentes [99].

Em computação evolucionária, a procura da melhor solução (cromossoma) para um problema, recorrendo a algoritmos genéticos, consiste em criar novas soluções a partir de um conjunto inicial pré-selecionado pela recombinação e mutação dos seus elementos, mantendo o número de elementos constante. A cada geração é avaliado o desempenho de cada nova solução e as melhores soluções servirão de base para a criação de novas soluções na geração seguinte. Este processo evolutivo termina assim que um determinado critério seja atingido, tal como: atingir um número máximo de gerações (iterações), um determinado rácio TP/FP; assim que são esgotadas as hipóteses de criação de uma geração diferente das até então avaliadas.

Uma descrição mais completa destes algoritmos, e exemplos das suas aplicações, são apresentadas em [99, 100].

A abordagem desenvolvida foi inspirada na computação evolucionária para a determinação das melhores combinações de características a fornecer aos classificadores para a segmentação dos limões em relação aos restantes elementos presentes em cada imagem, em que o número de elementos (características) das combinações aumenta em cada geração.

Essa característica adicional é obtida a partir dos elementos que constituem as combinações com melhor desempenho na geração anterior. Para o efeito são hierarquizadas as combinações em função da AUC (*Area Under the Curve*) para um intervalo admissível da taxa de falsos-positivos e para um tempo máximo de execução previamente definido.

Esta metodologia inicia-se com uma abordagem tipo “força-bruta”, ao testar todas as combinações de duas características, mas, a cada nova geração, podem ser desconsideradas certas características, caso não integrem individualmente as melhores combinações selecionadas.

Esta metodologia pretende limitar o número de combinações a avaliar, visando economizar recursos no treino dos modelos de classificação. Contudo, não significa que uma combinação excluída não fosse originar o classificador com melhor desempenho!

Na primeira geração são então avaliadas todas as combinações possíveis com apenas duas características de cor, no total de 55 ( $C_2^{11}$ ) combinações provenientes do conjunto inicial de características  $\mathbb{P}$ . Nas gerações seguintes são mantidas as  $n$  melhores combinações da geração anterior e criado um subconjunto  $\mathbb{C}$  com todas as características únicas presentes nessas combinações ( $\mathcal{v}$ ), de tal modo que  $\mathbb{C} \subseteq \mathbb{P}$ .

As novas combinações são geradas por uma metodologia idêntica à apresentada em [101], isto é, cada nova combinação irá resultar da concatenação de uma combinação da geração anterior com um novo elemento proveniente de  $\mathbb{C}$ . O processo é executado de forma a combinar cada combinação da geração anterior com cada elemento de  $\mathbb{C}$ , conforme ilustram a Fig. 4.5 e a Fig. 4.6. Este processo é repetido em cada geração.

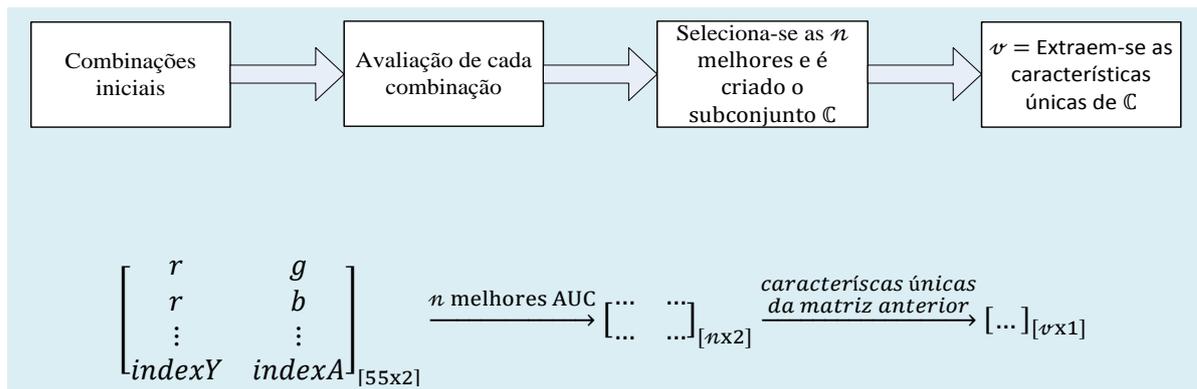


Fig. 4.5. Esquema de geração de novas combinações.

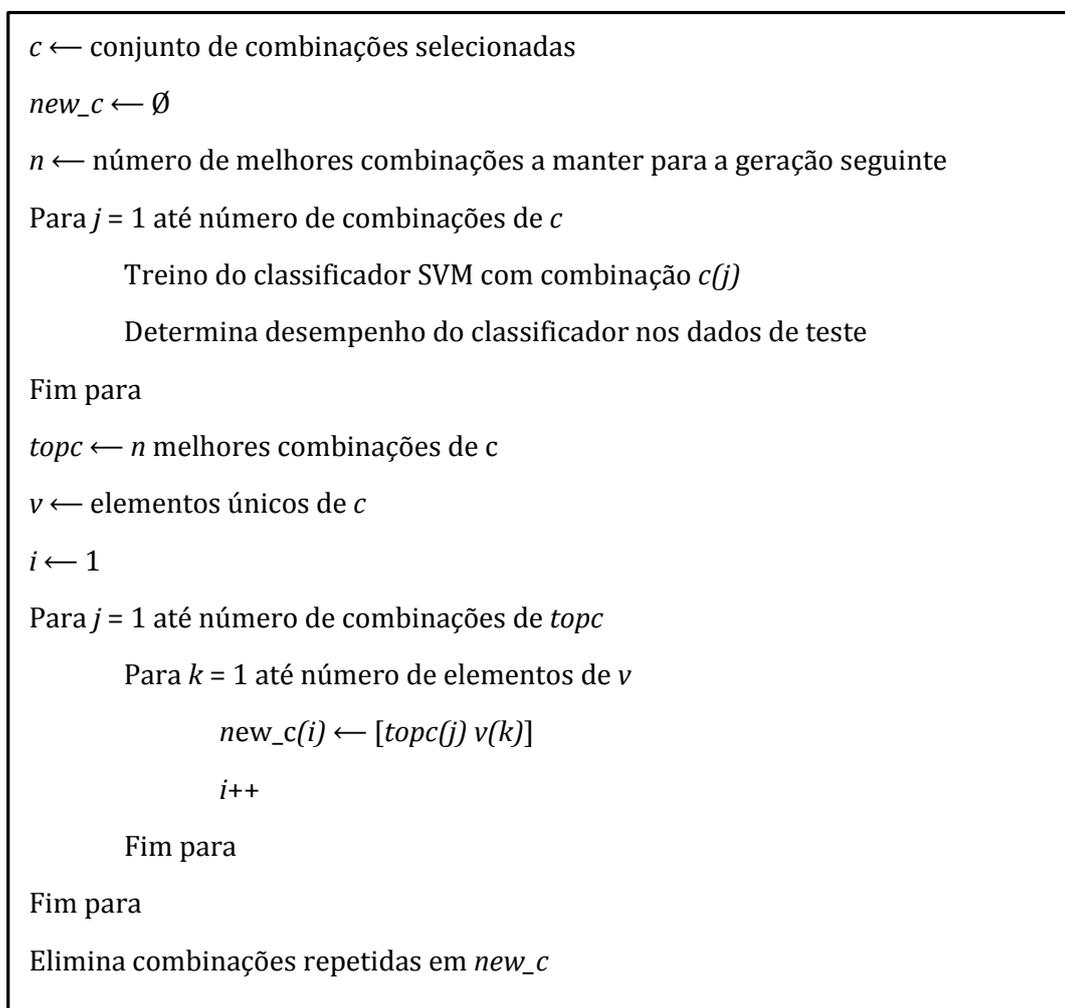


Fig. 4.6. Algoritmo de geração de combinações.

Além do espaço de características, as parametrizações do SVM têm influência no treino do classificador e respetiva resposta nos dados de validação e teste. A avaliação de todas as

combinações de parâmetros pré-estabelecidos através de métodos de força-bruta não representa uma opção viável em termos práticos devido aos recursos computacionais consumidos relativamente à metodologia proposta. Concretamente, a diferença entre os métodos é feita entre um processo iterativo simples, trepa-colinas, e um processo iterativo matricial que por si só exige mais tempo de processamento, podendo o mesmo ser potenciado pela exigência de recursos adicionais de memória.

Assim, recorre-se ao método designado por trepa colinas (do inglês *Hill-climbing*), um método de pesquisa local, que é aplicado de modo a provocar uma variação ascendente ou descendente dos parâmetros a seguir referidos, no sentido de melhorar os resultados atingidos. Os parâmetros a controlar com esta metodologia são:  $\sigma$ ,  $C$  e  $c_r$ . Definida uma configuração inicial, a metodologia começa por determinar qual o melhor valor a atribuir ao parâmetro  $\sigma$ , seguindo-se o parâmetro  $C$  e  $c_r$ , respetivamente.

Por ser um método de pesquisa local, há o risco de se convergir para uma solução ótima local e não global. A forma utilizada para evitar ficar-se preso a uma solução ótima local e não global é através da inicialização desta abordagem em pontos diferentes. Assim, o método irá convergir para soluções ótimas locais e espera-se que uma destas soluções represente a solução ótima global.

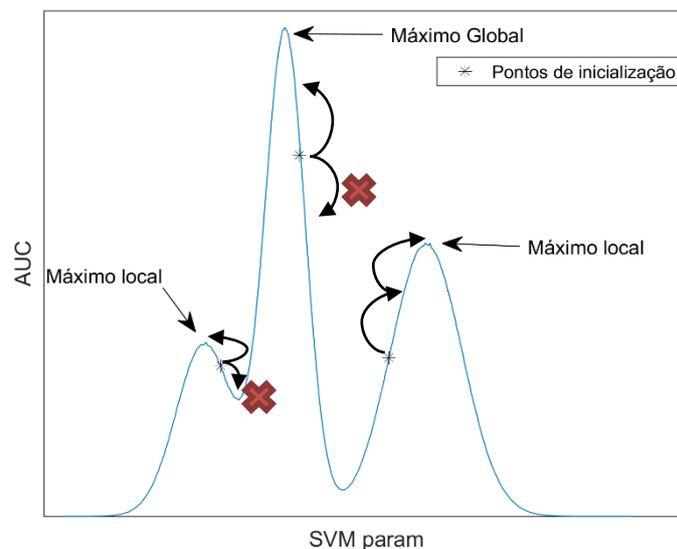


Fig. 4.7. Exemplo de seleção de parâmetros pelo trepa colinas.

A Fig. 4.7 demonstra o modo de evolução do trepa colinas na seleção dos parâmetros do SVM. O processo começa por avaliar a exatidão na configuração inicial (pontos de inicialização) e, por defeito, analisa a configuração seguinte no sentido crescente do parâmetro. Se a exatidão aumentar, a evolução continua no sentido crescente; se diminuir, a evolução é feita no sentido decrescente. Este processo é repetido até que seja atingido o máximo local ou até um número máximo de iterações definido pelo utilizador. A Fig. 4.8 apresenta o fluxograma deste processo.

Os parâmetros  $\sigma$  e  $C$  podem assumir valores muito pequenos ou muito elevados; não é conveniente avaliar o desempenho de um classificador SVM fazendo-os variar de forma linear, por isso a evolução destes parâmetros segue um crescimento exponencial (equação (4.1)), onde é feito variar o seu expoente segundo um crescimento linear.

Porém, quanto maior for o expoente da função, maior será a variação do parâmetro  $\sigma$  ou  $C$  entre iterações sucessivas. Como forma de compensar essas grandes variações e de perder a convergência para o máximo global, a partir do momento em que não são detetadas melhorias na exatidão é mantido o valor do melhor expoente avaliado ( $\beta'_0$ ), sendo reduzida a variação do expoente numa razão decimal pelo incremento do parâmetro  $\beta_{div}$ .

A equação (4.10) representa a evolução do parâmetro  $\sigma$ , sendo aplicado um método análogo ao parâmetro  $C$  do SVM (equação (4.11)).

$$\sigma(i) = 10^{\alpha + \beta_0 \pm \frac{\beta(i)}{10^{\beta_{div}}}} \quad (4.10)$$

$$C(i) = 10^{\alpha + \beta_0 \pm \frac{\beta(i)}{10^{\beta_{div}}}} \quad (4.11)$$

$$\beta'_0 = \beta_0 \pm \frac{\beta(i)}{10^{\beta_{div}}} \quad (4.12)$$

Nas equações precedentes,  $\alpha$  representa o valor de inicialização do expoente,  $\beta_0$  inicia-se com o valor 0 e assume o valor de  $\beta'_0$  sempre que ocorrer o incremento de  $\beta_{div}$  nas condições supra mencionadas (equação (4.12)).  $\beta$  define os valores da variação do expoente e, pela formulação estipulada, é recomendável que lhe sejam atribuídos apenas valores naturais ( $\beta_{div} \in \mathbb{N}^*$ ).

O número máximo de iterações é determinado no algoritmo de treino do classificador em função do valor máximo admissível de  $\beta_{div}$  e dos valores de  $\beta$ .

O outro parâmetro, cuja solução ótima é determinada pelo “trepa colinas”, é o  $c_r$ , sendo esta avaliada da mesma forma que é representada na Fig. 4.7, mas sendo a sua evolução feita de forma linear, contrariamente aos parâmetros anteriormente mencionados.

Os critérios de aceitação dos classificadores são definidos em função do tempo estimado despendido para o processamento de uma imagem completa (resolução 640x480) e para uma taxa máxima de FP. A experiência proveniente dos primeiros ensaios realizados indica que, ao longo da evolução do “trepa colinas” na determinação das parametrizações do SVM, não há uma tendência definida de evolução do tempo de processamento estimado para a classificação de uma imagem completa. Assim, as únicas condições de paragem colocadas na determinação destes parâmetros através do “trepa colinas”, é através do número máximo de iterações ou após ter sido encontrado o máximo local.

Após a resposta convergir para um valor máximo é selecionada a melhor configuração que respeite o tempo máximo de processamento estabelecido, assim como a taxa máxima de FP admissível. Havendo a possibilidade dessa configuração não ter sido explorada por incremento de  $\beta_{div}$ , por exemplo por existir outra solução com melhor exatidão, mas que não cumpra os critérios

do tempo e de FP, essa solução é refinada nos moldes anteriores, com as condições de tempo e de taxa de FP ativas como critério de paragem.

Sabe-se que, tradicionalmente, quanto maior for o valor assumido pelo parâmetro  $C$  mais tempo é exigido no treino do classificador. Contudo, no uso do modelo treinado para a classificação, não foi verificada a mesma tendência.

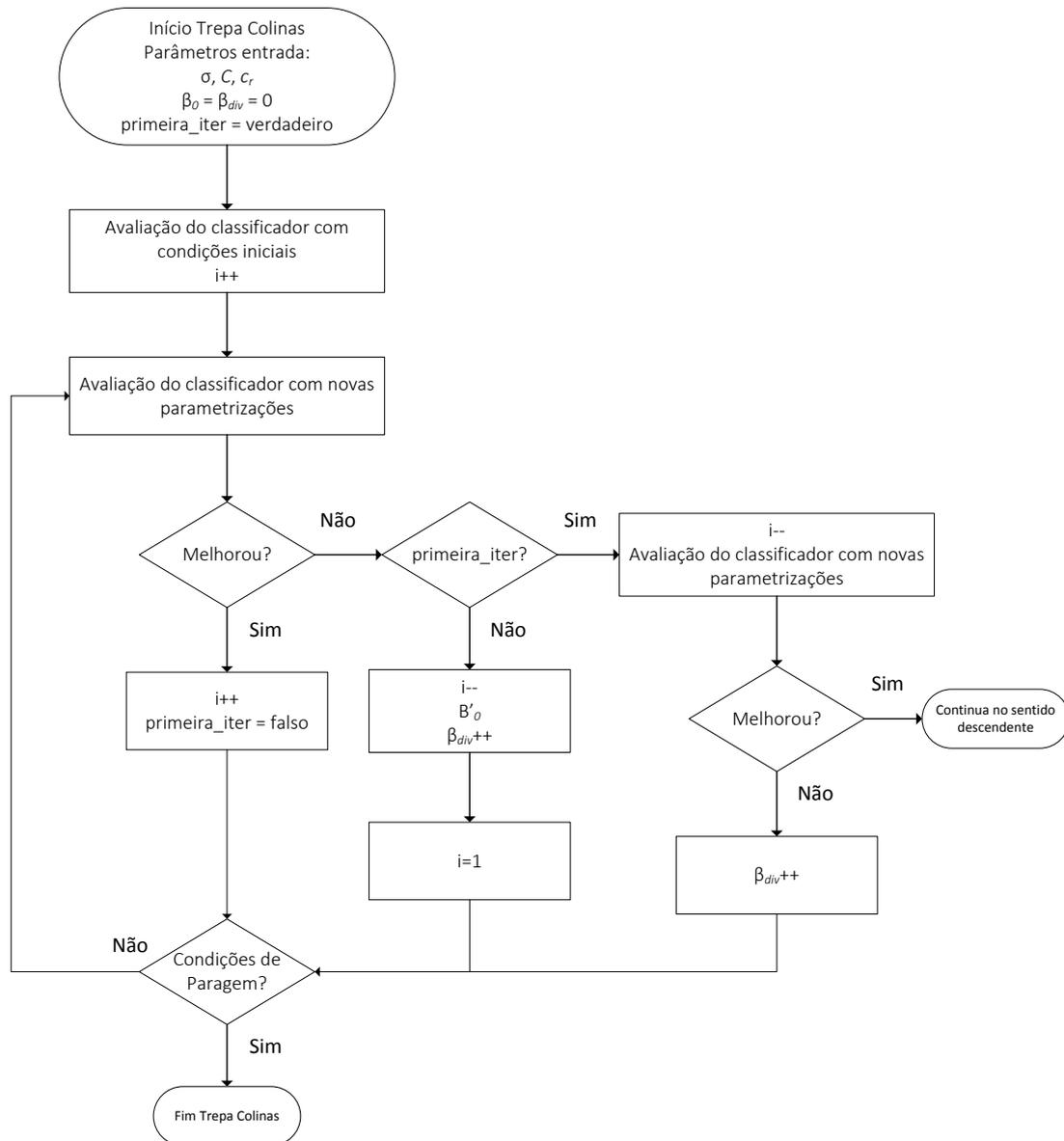


Fig. 4.8. Fluxograma do trepa colinas aplicado à determinação das parametrizações do SVM.

O fluxograma da Fig. 4.8 esquematiza as operações realizadas para o melhor valor de  $\sigma$  e  $C$  no sentido ascendente. Caso se verifique que se obtém uma melhor exatidão no sentido descendente, o procedimento é similar. Depois de determinado o valor ótimo local para  $\sigma$ , é realizado o mesmo procedimento para determinar  $C$  e, após este, a determinação de  $c_r$ . O

procedimento para este último parâmetro não se encontra esquematizado por representar uma situação de menor complexidade. Determinados estes três parâmetros segue-se o processo de refinamento dos resultados.

A metodologia proposta determina os parâmetros do SVM de forma autónoma, para que ocorra um melhor desempenho na classificação, avaliado pela sua exatidão.

## 4.2 Processamento em HPC

Os centros de computação de elevado desempenho (HPC) são supercomputadores usados para acelerar o processamento de dados, os quais, em computadores domésticos demorariam imensamente mais tempo para serem processados. A velocidade da aceleração depende das características do *hardware* das máquinas e da qualidade do *software* desenvolvido para submissão nos programas de cálculo. Por envolver a interligação de várias unidades de processamento, um supercomputador é também designado de *cluster*.

O processamento em HPC é realizado quando há uma exigência de capacidade de cálculo ou de memória que ultrapassam as capacidades de unidades comuns, como são exemplo aplicações de previsões meteorológicas, processamento de grandes quantidades de dados (*big data*) ou o treino de modelos de inteligência artificial.

Como existem muitos processadores disponíveis, a conclusão de um trabalho é acelerada pela divisão de tarefas que possam ser executadas em simultâneo por estes. Nem todos os trabalhos justificam ser executados num *cluster*; o manual do *cluster* utilizado nestes ensaios [102] faz uma abordagem simples e esclarecedora de que tipo de programas justificam o recurso a um *cluster* HPC.

O *cluster* Minerva [102] é um HPC com o sistema operativo CentOS 7.3, com 20 nós de computação, cada um com dois processadores (Intel Xeon E5-2695v2) de 12 *cores*, perfazendo um total de 480 *cores* disponíveis para processamento de dados. Além de outras características, tem, no total, 192 GB de memória RAM e tem instalado o *software* Matlab 2016a usado nas simulações apresentadas nesta dissertação. Relativamente ao seu desempenho, os cálculos realizados anunciam um desempenho teórico máximo de 9.216 TFLOPS e as avaliações realizadas, segundo os padrões internacionais, reportam um desempenho máximo real de 8.041 TFLOPS, o que corresponde a uma eficiência de 87.3 %, [102].



Fig. 4.9. Fotos dos bastidores do cluster Minerva, imagens cedidas pelo LaCED.

Um *cluster* apresenta uma grande capacidade de aceleração, quando se trata de tarefas repetitivas em que são apenas alterados alguns parâmetros entre simulações. Na Fig. 4.10 está representado o tempo despendido na execução de vários processos, onde é admitido que:

- Cada processo demora uma hora a ser executado;
- Um PC i7 consegue executar oito processos em paralelo;
- O *cluster* Minerva executa 480 processos em simultâneo.

Como se pode verificar, para poucos processos não é justificável o recurso a HPC, pois não há ganhos significativos em termos de tempo de execução, e representaria encargos económicos de muito maior dimensão.

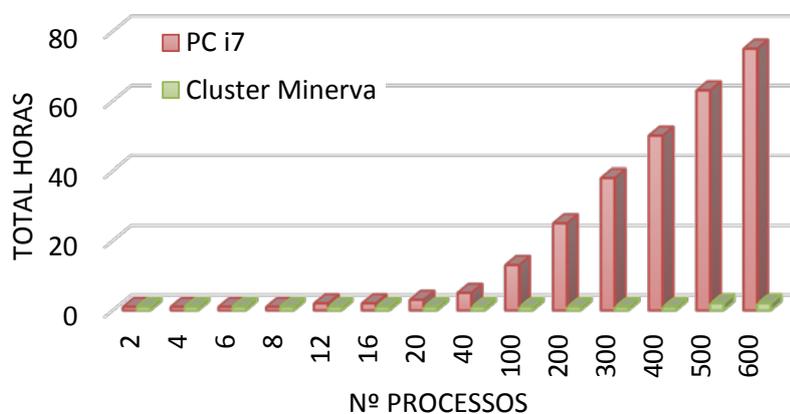


Fig. 4.10. Tempo processamento PC vs Cluster.

No entanto, os processos a serem submetidos a um *cluster* têm que ser: devidamente planejados e elaborados, considerando que não há intervenção humana durante o processamento; que são definidos os parâmetros de entrada, e apenas se tem acesso aos parâmetros de saída; qualquer outra variável ou ficheiro adicional, que seja necessário analisar, deve ser indicada a sua localização para que o processo seja concluído com sucesso.

Outra condição imposta nas simulações realizadas é a seguinte: os processos têm que ser independentes, ou seja, quando uma tarefa com vários processos é submetida, estes não podem depender de resultados entre eles. Assim, os algoritmos devem ser preparados para convergirem autonomamente!

Além disso, para uma maior rentabilidade, os *scripts* devem ser preparados para serem genéricos, isto é, para que possam ser frequentemente reutilizáveis apenas pela modificação dos parâmetros de entrada. Contudo, o tempo gasto em preparar um *script* genérico, que não receba intervenção humana a meio do processo, e que esteja preparado para receber e interpretar os dados de outros processos então terminados, é superior à elaboração de um *script* que possa ser intervencionado durante a sua execução.

Considerando que um *script* para ser executado num computador pessoal i7 demora cerca de oito horas a ser concebido, e que para o processamento num *cluster* o *script* demora cerca de três vezes e meia mais tempo a ser elaborado, a análise de retorno de investimento não pode ser feita com base apenas na Fig. 4.10. Considerando o tempo de elaboração dos *scripts*, Fig. 4.11, só com cerca de 200 processos é que se justifica, em termos de tempo, recorrer a um *cluster* HPC.

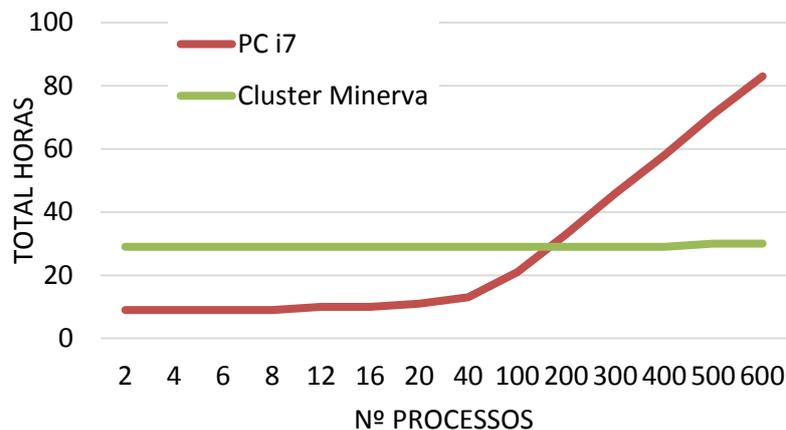


Fig. 4.11. Análise comparativa do tempo gasto na elaboração dos *scripts* e execução dos processos: PC i7 vs Cluster Minerva.

Os *scripts* desenvolvidos em Matlab são compilados e, posteriormente submetidos ao *cluster*, onde um gestor de tarefas (SLURM - *Simple Linux Utility for Resource Management* [103]) é

responsável por gerir todos os pedidos e respetivas prioridades e distribuir as tarefas pelos diferentes nós de computação.

O fator de aceleração do processamento num *cluster* também está relacionado com a arte de subdividir um programa em pequenos processos passíveis de serem executados em paralelo.

## 5. DETEÇÃO

A detecção dos frutos pode ser facilitada por um pré-tratamento da imagem, através do processo de segmentação proposto nos capítulos anteriores, com o propósito de eliminar significativamente o ruído existente, restando apenas analisar as regiões com tonalidades semelhantes às dos limões.

Quando se trata de detetar laranjas ou maçãs é comum serem empregues metodologias de detecção de circunferências na imagem como método de detecção de frutos [23, 31]. Porém, em condições reais nestes cenários os frutos são, frequentemente, captados em condições de ocultação parcial, sendo o grau de semelhança do fruto ao seu formato ideal negativamente afetado.

Além disso, os frutos não são exatamente iguais, contrariamente ao que ocorre na detecção de peças mecânicas, onde se pode realizar a detecção das mesmas usando como referência o respetivo modelo 3D [104, 105]. Por consequência há a necessidade de adotar metodologias alternativas.

Não obstante os resultados promissores na detecção de frutos com recurso a redes neuronais convolucionais (CNN) [77, 80], não é considerada a aplicação desta metodologia na presente dissertação por impossibilidade de avaliação do seu desempenho no cenário avaliado, devido à indisponibilidade de recursos computacionais.

No entanto, em [14] é apresentada uma metodologia baseada na forma dos frutos para proceder à sua detecção. Em vez de comparar, na imagem segmentada, as semelhanças das regiões resultantes com um formato ideal, os autores assumem que as regiões resultantes são compostas por círculos e sugerem percorrer o perímetro dessas regiões para estimar a localização do centro das circunferências; seguidamente, em função da concentração dos hipotéticos centros, é determinada a existência de um fruto. Devido à semelhança do formato dos limões com uma elipse, essa metodologia não corresponde à opção eficiente para a detecção dos limões na árvore.

Em [10], a detecção dos frutos é realizada pela aplicação da transformada de *Watershed* após a segmentação. Para ultrapassar o problema da subdivisão excessiva das regiões iniciais em pequenas regiões são aplicadas operações morfológicas de dilatação e erosão. Assim, é mencionado que a metodologia proposta permite detetar corretamente 91 % dos frutos nas imagens avaliadas, para 15 % de ocorrência de falsos-positivos; de salientar que a fórmula de cálculo da taxa de falsos-positivos é diferente da apresentada nesta dissertação para os dados relativos à segmentação.

Wachs *et al.*, [28], aborda a detecção de maçãs pelo método de *Haar* usando imagens RGB e termográficas. O método de *Haar* é bastante popular na detecção de faces e rápido, sendo o

processamento realizado com base numa imagem integral e os classificadores treinados e, posteriormente, aplicados seguindo a metodologia de aprendizagem *AdaBoost* [29, 106].

Os fundamentos do método *AdaBoost* consistem em aplicar uma sucessão de classificadores fracos (designados em inglês por *weak classifiers*) que não prejudicam a taxa de acerto dos elementos da classe positiva, mas que, individualmente apresentam uma taxa elevada de falsos-positivos ou de falsas-deteções. Como existem vários estágios de classificação, com classificadores “fracos”, a taxa de falsos-alarmes (FA) global tenderá a baixar com o aumento do número de estágios (equação (5.1)), com o objetivo de se obter um classificador (global) robusto ( $F(x)$ ), conforme representado na equação (5.2).

$$FA_{global} = FA_{estágio}^{\text{número de estágios}} \quad (5.1)$$

$$F(x) = \sum_{i=1}^n \alpha_i f_i(x) \quad (5.2)$$

Onde,

$f_i(x)$  representa cada classificador “fraco”

$\alpha_i$  representa o peso atribuído a cada classificador “fraco”

$n$  é o número total de classificadores “fracos”

$x$  é um vetor que contém as características a analisar

Nos resultados apresentados em [28] verifica-se uma elevada ocorrência de falsos-alarmes. No entanto, os autores argumentam que a solução proposta é viável, pois, como a câmara é montada num braço robótico, haverá a possibilidade de confirmar se uma determinada região representa efetivamente um fruto, ou não, pela análise de várias imagens da mesma região. Usando os resultados apresentados nessa publicação, e aplicando as métricas de avaliação do desempenho das metodologias de deteção dos frutos nesta dissertação, apresentadas na seção 6.4, a metodologia proposta apresenta uma *Precision* = 74.37 % e um *Recall* = 66.37 %.

Em [34, 35] é proposto realizar a deteção de pimentos através da segmentação de imagens por três metodologias distintas: SAE; LBP; e HOG. O resultado da segmentação por cada um dos métodos é um mapa de probabilidade que é sujeito a uma análise de deteção de objetos (*blobs*) pela metodologia *Laplacian of Gaussian* (LoG). Da avaliação individual das metodologias de segmentação, a que apresenta melhor desempenho é o método LBP. Contudo, combinando a informação proveniente de todas as metodologias avaliadas, permite obter ainda melhores resultados. Os autores reportam que, nos dados de validação são obtidos resultados de *Precision* = 75.8 % e, em teste, uma *Precision* = 69.2 %. Quanto aos falsos-alarmes, os dados são expostos usando uma métrica diferente da proposta nesta dissertação, sendo verificado, em média, 1.2 FA nas 32 imagens de validação, subindo este valor para 2.1 nas 33 imagens usadas para teste.

A detecção de frutos, no cenário em análise nesta dissertação, é avaliada pela parametrização de soluções pré-existentes no Matlab 2016a, nomeadamente na classe denominada *Object Detectors*, parte integrante de *Computer Vision System Toolbox*.

Relativamente à construção de classificadores treinados em cascata, as metodologias pré-configuradas são: HOG, LBP e *Haar*. Pelos resultados de outros ensaios [28, 34, 35] espera-se que os melhores resultados sejam alcançados pelo método LBP. Contudo, como a exigência temporal para o treino destes classificadores é inferior à do treino dos modelos SVM, segundo a metodologia adotada para a segmentação das imagens, é feita a avaliação destes três métodos usando o *cluster* HPC Minerva.

No treino dos classificadores é necessário providenciar amostras da classe positiva e negativa, para que o método aprenda a classificar corretamente os limões nas imagens. O número de amostras usadas é relatado na secção 6.3.

Além das amostras, também é definido o método a usar, o número de estágios da cascata, o valor admissível de FA por estágio e a imagem monocromática a analisar. Por imagem monocromática entende-se a componente de um dos modelos de cor usados.

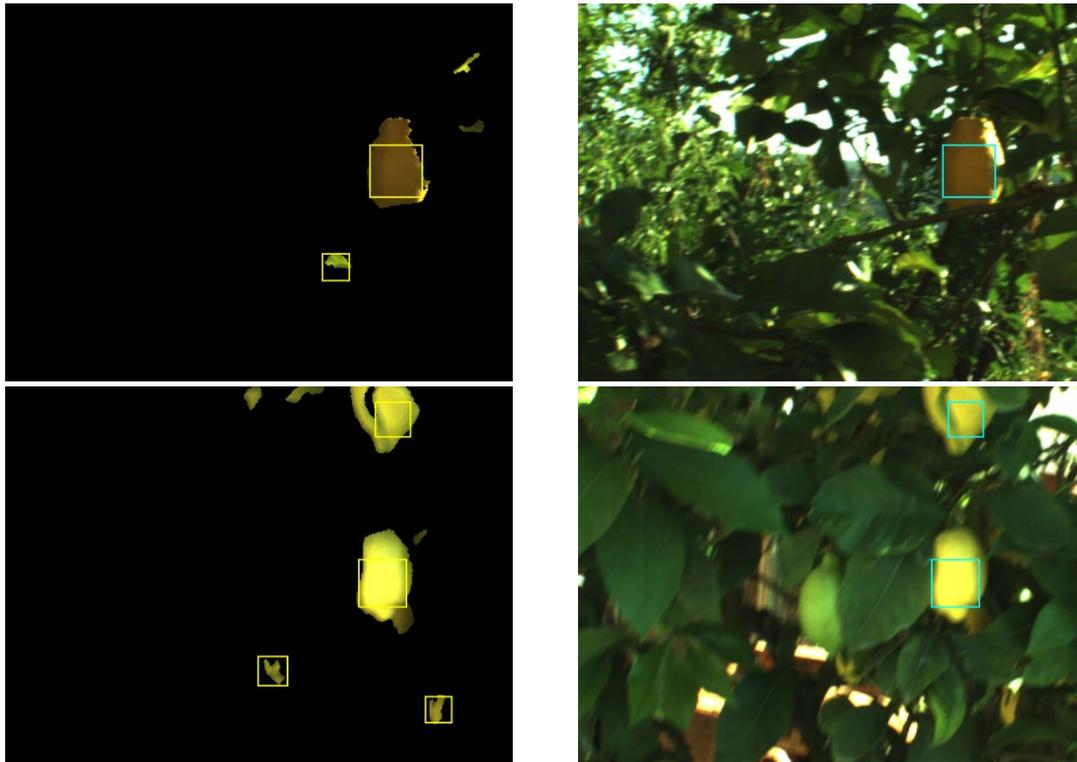
Estas soluções pré-configuradas do Matlab recorrem a funções do OpenCV [107], uma biblioteca de visão artificial, na qual é fornecida uma imagem e o modelo de classificação, sendo devolvida uma lista com a(s) ROI encontrada(s), na fase de aplicação do classificador. Neste processo não há acesso ao grau de confiança de semelhança da ROI reportada com as condições ideais, nem ao mapa de probabilidade referido em [35].

Estes métodos podem ainda ser aplicados diretamente na imagem original sem se realizar a etapa de segmentação proposta nesta dissertação. Porém, essa opção daria origem a uma análise mais complexa e à ocorrência de um maior número de falsos-alarmes. Com a metodologia de segmentação proposta, através de classificadores SVM, é eliminada a maior parte dos elementos pertencentes a outra classe distinta dos limões e, simultaneamente, mantendo a maior parte dos *pixels* dos limões ativa para se poder realizar a detecção dos frutos.

Dos resultados provenientes dos classificadores treinados, verificou-se: a existência de ROI parcialmente sobrepostas e relativas ao mesmo fruto; ROI de grandes dimensões, a englobar vários frutos separados; e a detecção em regiões pequenas, que efetivamente representam frutos, mas em condições em que apenas uma pequena mancha é visível entre os elementos do *background*, não representando assim, desse ponto de vista, um fruto a colher. A fim de solucionar estas situações foram adotadas as medidas a seguir apresentadas.

Relativamente às regiões de pequenas dimensões, foi adotada uma estratégia em duas etapas - a primeira etapa consiste na eliminação das regiões com área inferior 80 *pixels*, após aplicação da operação morfológica de erosão (aqui apresentada no final do processo de segmentação por SVM); assim, procura-se eliminar todas as regiões individuais de pequenas dimensões.

Seguidamente, das ROI sugeridas pelo classificador, são excluídas todas as que apresentarem uma área inferior a 1500 *pixels*. Esta proposta deverá estar relacionada com a distância estimada dos frutos à câmara, pois quanto maior for essa distância menor será a área ocupada pelo fruto na imagem e, naturalmente, menor será o retângulo (ROI) referente a esse fruto. A Fig. 5.1 ilustra duas situações em que são sugeridas ROI de pequenas dimensões (coluna à esquerda) e que são filtradas com o método proposto (coluna da direita). Note-se que na imagem da última linha há sugestões de ROI onde realmente existem limões, mas, por representarem regiões tão pequenas, são excluídas do restante processo de análise.



*Fig. 5.1. Exemplo da remoção de pequenas áreas: (esquerda) ROIs proposta pelo classificador; (direito) resultado da filtragem pela área.*

Como as imagens usadas no treino e validação foram selecionadas procurando que a aparência dos frutos nas imagens fosse aproximadamente constante, este limiar aplicado às ROI propostas pelo modelo de classificação permite eliminar os falsos-alarmes originados por pequenas manchas.

Futuramente, ao alterar a distância da câmara aos frutos, este limiar pode ser, facilmente atualizado para a obtenção de uma melhor resposta do sistema. Também, com a aquisição sucessiva de imagens e ou utilização de sensores adicionais, pode-se inferir a distância da câmara a cada região da imagem e usar um limiar global mais adequado, ou até agir localmente, ajustando esse limiar em função da distância estimada da câmara aos elementos de uma determinada região da imagem.

Quanto às ROI sugeridas de grandes dimensões, e a envolver vários frutos em regiões separadas da imagem, é analisada a percentagem de área ativa de cada uma dessas ROI. A área ativa refere-se àquela que não foi eliminada no processo de segmentação por SVM. Desta forma, todas as ROI que apresentem uma percentagem de área ativa inferior a 40 % (limiar que pode ser ajustado) são excluídas como região onde exista um fruto.

A Fig. 5.2 ilustra, na primeira linha, o caso de uma ROI de grandes dimensões onde a maioria da área ocupada é eliminada na segmentação. Na segunda linha, a ROI na posição mais superior da imagem também é eliminada por apresentar uma pequena percentagem de área ativa.

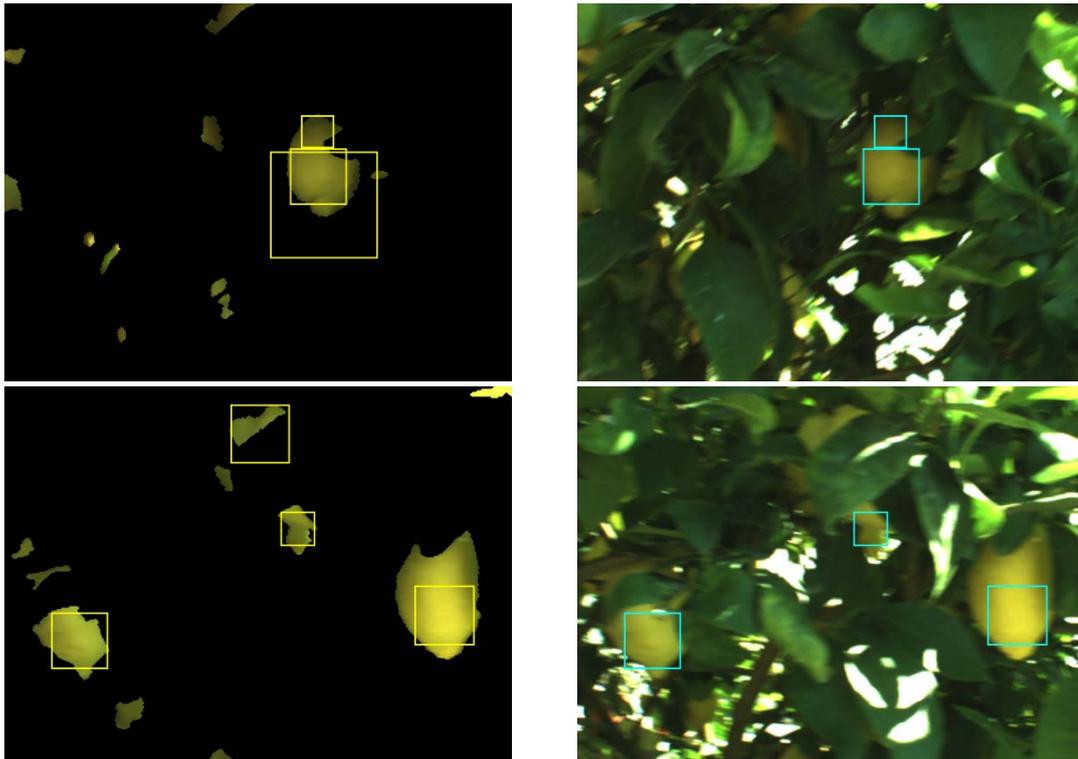


Fig. 5.2. Exemplo da remoção ROIs com pouca área ativa: (esquerda) ROIs proposta pelo classificador; (direito) resultado da filtragem pela área ativa.

As ROI parcialmente sobrepostas são avaliadas em relação à área comum às ROI com uma área da ROI de menores dimensões. Se essa relação for superior ao limiar definido de 50 %, essas duas ROI originarão uma única ROI. Os limites desta nova ROI são determinados de forma a abranger os limites individuais de cada uma das duas ROI originais, conforme esquematizado na Fig. 5.3. Em casos extremos, quando uma ROI está completamente contida noutra, a ROI resultante é a mesma da ROI original de maior dimensão. Este processo é repetido até que não existam ROI cuja relação de área sobreposta seja inferior à parametrizada no limiar.

A Fig. 5.4 demonstra dois casos em que o classificador indica ROI sobrepostas (à esquerda) e o resultado da criação de novas ROI, conforme é descrito no parágrafo anterior.

Todos os limiares referidos foram determinados empiricamente e são mantidos fixos para todas as simulações de detecção realizadas. No futuro poderão ser ajustados em função das condições de cada caso, podendo, inclusivamente, ser parâmetros de entrada da função responsável pela detecção dos frutos.

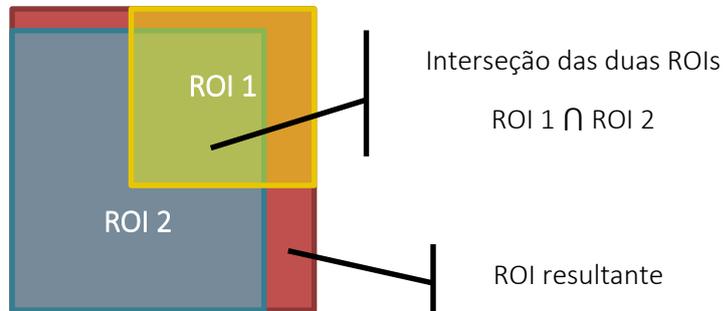


Fig. 5.3. Esquema da ROI resultante e da área a abranger.

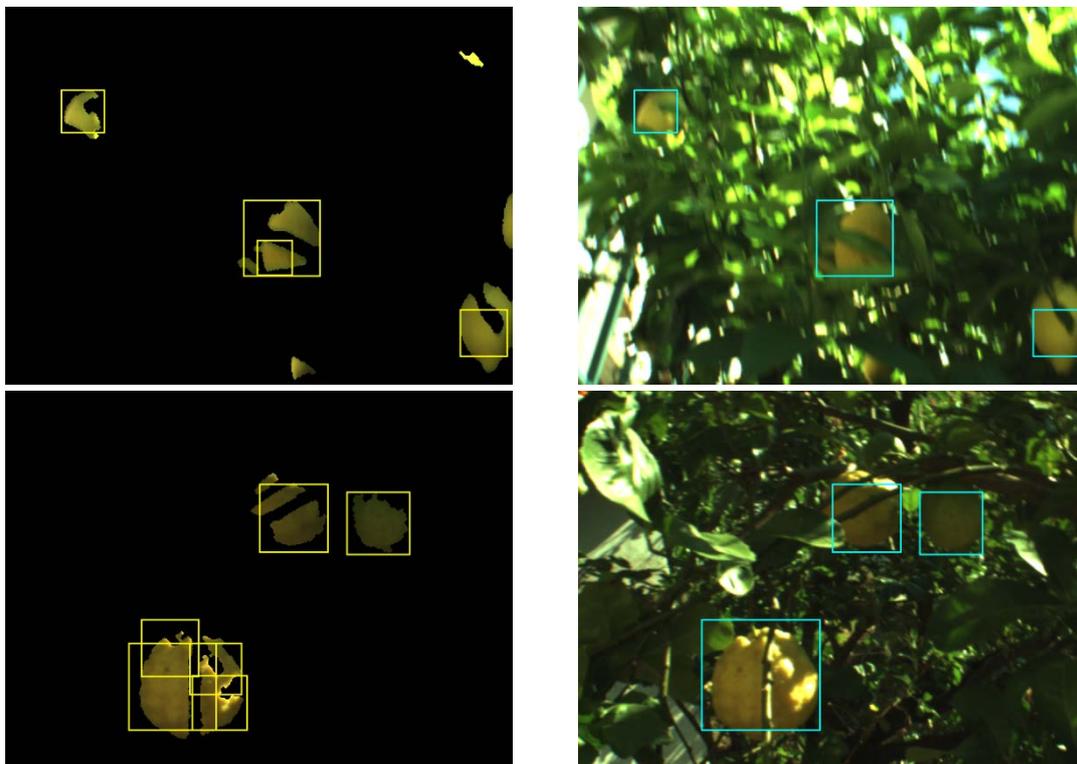


Fig. 5.4. Exemplo da união de ROIs que se interceptam: (esquerda) ROI proposta pelo classificador; (direito) resultado da união de ROI.

## 6. DADOS EXPERIMENTAIS

Este capítulo apresenta o sistema de visão utilizado na captura de imagens, os modelos de cor usados no processamento de imagem de modo a atenuar os efeitos das variações das condições de luz, a quantidade de dados usados no treino, validação e teste dos classificadores e, por fim, as métricas usadas para avaliar a resposta das metodologias implementadas.

### 6.1 Sistema de visão e base de dados

A avaliação de metodologias de segmentação e detecção de objetos, neste caso de limões, é realizada sobre uma base de dados que contém cada caso devidamente classificado.

Apesar de haver bases de dados com imagens devidamente classificadas, não foi identificada nenhuma disponível para o caso específico da segmentação e detecção de limões nas árvores.

Dada a lacuna, foi criado um sistema físico para aquisição de imagens no terreno, composto por: três câmaras *GigE Vision*; um gerador de sinais; um *Ethernet switch*; e um computador para armazenamento dos dados e baterias para alimentação elétrica das câmaras e do *switch* (Fig. 6.1). O gerador de sinais é um elemento essencial para assegurar uma taxa de aquisição de imagens constante e sincronizada pelas três câmaras do sistema.

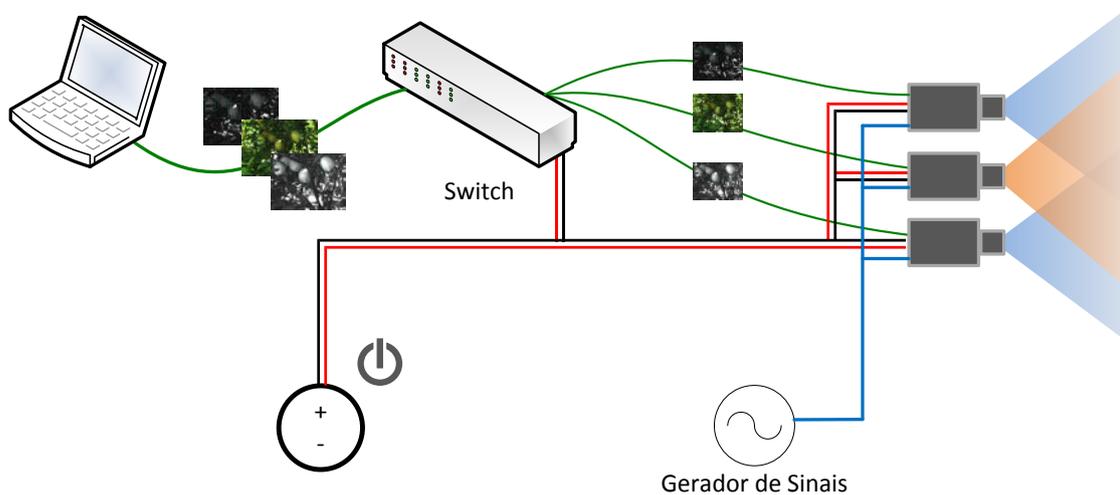


Fig. 6.1. Esquema do sistema de aquisição de imagem.

O sistema de visão é constituído por uma câmara a cores JAI CB-140-GE numa posição central com uma lente de 6 mm acoplada e duas câmaras monocromáticas Manta G146B equipadas com lentes de 8 mm, tal como ilustrado na Fig. 6.2. A literatura [10] identifica as regiões do espectro

eletromagnético no intervalo entre os 580 nm e os 700 nm como de maior contraste entre os citrinos e a folhagem (Fig. 2.1); foram ainda montados filtros óticos passa-banda nas câmaras monocromáticas. Numa câmara foi montado o filtro BP525 e, na outra o filtro BP635, ambos do fabricante *Midwest Optical Systems*, para restringir a captura de imagens em regiões específicas do espectro. A escolha do filtro BP525 está fora do intervalo de comprimentos de onda recomendados, pelo facto das análises hiperespectrais terem sido realizadas para outra classe de citrinos que não a dos limões, prevendo-se um ligeiro desvio do sinal em relação ao estudo original.



Fig. 6.2. Sistema de visão experimental.

Durante a aquisição de imagens, o sistema foi manipulado por um operador simulando hipotéticos movimentos de um braço robótico num processo de apanha de fruta. Além disso, procurou-se que a distância entre as câmaras e o fruto mais próximo fosse de 40 cm a 120 cm com uma taxa de aquisição constante de 20 fps. De modo a abranger o número máximo de cenários possíveis, as imagens foram adquiridas em duas plantações diferentes sem recurso a iluminação artificial. Além disso, registaram-se imagens à volta da copa da árvore, incluindo zonas onde o sol incidia diretamente e zonas de sombra.

O sistema de aquisição de imagens apresentado é comparável com a proposta de Li *et al.*, [19], a qual permite adquirir imagens em vários intervalos do espectro eletromagnético. Contrasta com os sistemas multiespectrais convencionais [12, 17] por ter mais do que uma câmara e fazer a aquisição simultânea das várias assinaturas espectrais, sem prejuízo de haver mudanças no cenário captado entre imagens. Por outro lado, como o centro ótico das câmaras é diferente no sistema proposto, a correlação de *pixels* de uma mesma região entre imagens, a ser feita, requer maior processamento de sinal.

Das imagens registadas selecionaram-se 133 imagens a cores, com resolução de 640x480 *pixels*, representativas de diferentes cenários captados. Como o objetivo é a segmentação de limões em relação aos restantes elementos, segmentaram-se manualmente todos os *pixels* correspondentes

aos limões (Fig. 6.3). Depois de marcados todos os elementos da classe dos limões, considerou-se que a região complementar corresponde ao fundo (*background*).

Porém, os restantes elementos da imagem possuem características muito distintas e, como um todo, não expressam uma assinatura visual muito fidedigna.

Esta subdivisão faz parte de uma estratégia de procurar criar classificadores que determinem a assinatura visual (características mais relevantes) de cada subclasse do *background* para uma melhor segmentação dos limões dos restantes elementos. Ao recorrer a um classificador universal há o risco de menosprezar as subclasses com menor expressão e excluir um conjunto de características por não terem significado expressivo num classificador geral. No entanto, essas mesmas características podem fazer parte de um bom, ou até o melhor, classificador de uma subclasse face aos limões.

Assim, subdividiu-se a classe do *background* em cinco subclasses:

- Folhagem (verde);
- Ramos;
- Elementos cinza;
- Elementos laranja;
- Folhagem seca ou velha.

A (sub)classe da folhagem verde é aquela com maior representação e a visualmente mais próxima de ervas que possam aparecer noutras imagens. Em função das condições de luz, estes elementos podem assemelhar-se aos limões.

A classe dos ramos é dominada por tons castanhos, pelo que é expectável que as características usadas para construir um classificador para esta classe sejam aplicáveis também a outros elementos castanhos, como a terra. Em condições de pouca luz, como no interior da copa das árvores, os limões ficam semelhantes, em termos de cor, aos ramos expostos diretamente à luz solar. Nestas condições, e à vista desarmada, não é fácil ao utilizador reconhecer rapidamente os limões.

As classes dos elementos cinza e laranja podem não ter significado noutros locais, mas, nas imagens captadas, existentes na base de dados criada, estão presentes e, por isso, importam segmentar.

A classe da folhagem seca foi criada por ser um elemento presente na base de dados para avaliar a resposta dos classificadores. Contudo, sabe-se de antemão que, apenas pela cor, não será fácil distinguir os limões da folhagem seca.

A caracterização destas subclasses foi realizada por amostragem, ou seja, não foram sujeitas ao processo de marcação minucioso, como os limões; foram marcadas algumas regiões nas imagens relativas a cada uma das subclasses.

Outros elementos, como o azul do céu, não são considerados neste estudo por serem irrelevantes na base de dados e por representarem uma classe bem definida e distinta do alvo em estudo, os limões.

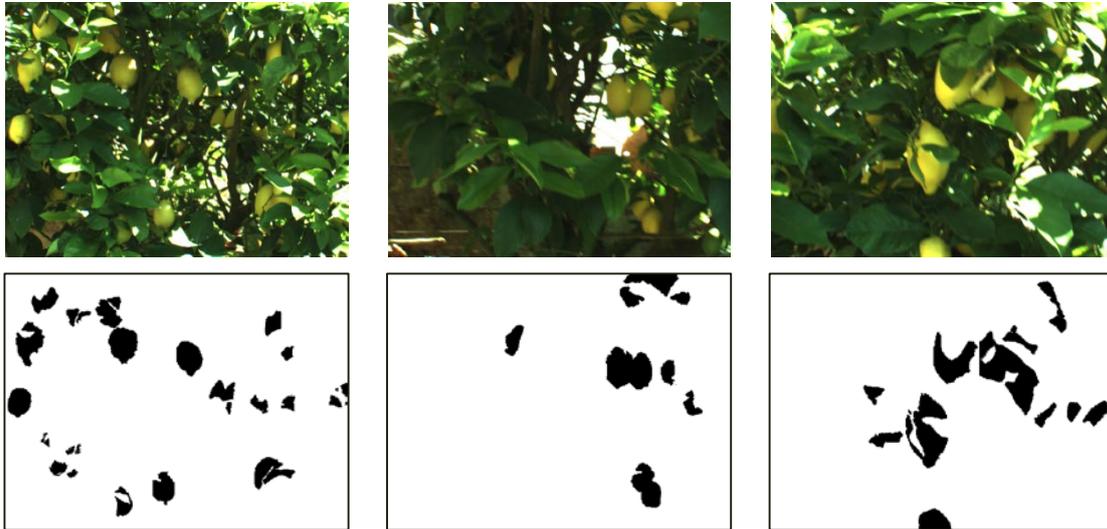


Fig. 6.3. Exemplo de imagens captadas e respetiva máscara dos pixels relativo à classe dos limões.

Elementos sujeitos a condições de luz adversas, condições de sub ou sobre exposição, são inicialmente classificados como *background*.

## 6.2 Normalização do modelo de cor

O presente estudo centra-se em imagens captadas no exterior, em condições de luz natural e sem recurso a iluminação artificial. Nestas condições é de extrema importância adotar metodologias que minimizem a influência das alterações da intensidade da luz no resultado final e que, em simultâneo, representem uma situação viável de aplicação em contexto real de trabalho.

No modelo de cor RGB, a proximidade de uma componente de cor ao nível da saturação é indicativa das condições de luz com que uma imagem foi adquirida. Significa, portanto, que um mesmo objeto originará respostas diferentes no modelo RGB quando sujeito a diferentes condições de luz. Por exemplo, na mesma imagem podem ser captados limões na copa exterior da árvore sujeitos a incidência direta dos raios de sol e outros no interior da copa afetados por sombras, apesar de, em ambos os casos, os limões terem a mesma cor, a resposta no modelo RGB será diferente.

Uma vez que a intensidade luminosa não é modelada nem constante é imperioso avaliar modelos de cor distintos do RGB, que permitam separar a intensidade de luz da cor.

As características a avaliar para distinguir os limões dos restantes elementos de uma imagem são as componentes individuais dos modelos de cor HSV e YCbCr. O modelo HSV é constituído pelas componentes de: cor (*Hue*); saturação (*Saturation* ou abreviado para *Sat*); e valor (*Value*, ou

abreviado por *Val*). Enquanto o modelo YCbCr possui uma componente afeta à luminância (Y), as outras duas componentes relativas às cromaticidades (Cb e Cr) derivam diretamente do modelo RGB [108].

Além dos modelos referidos são também consideradas as cromaticidades (*r*, *g* e *b*). Estas têm como desvantagem o fato de serem sensíveis a condições de fraca luminosidade, como se verifica pelas suas fórmulas:

$$r = \frac{R}{R+G+B}; g = \frac{G}{R+G+B}; b = \frac{B}{R+G+B} = 1 - (r + g) \quad (6.1)$$

Em condições ideais, os limões são representados pela cor amarela pura, isto é, quando a componente vermelha (R) e verde (G) atingem o seu valor máximo, e a azul (B) é nula; no modelo de cor RGB esta situação é representada por:

$$R = 1; G = 1; B = 0 \quad (6.2)$$

Quando a mesma cor é representada pelas cromaticidades, obtém-se:

$$r = 0,5; g = 0,5; b = 0 \quad (6.3)$$

Porém, em condições reais, a iluminação varia e pode haver alguma sujidade acumulada na fruta ainda na árvore; no entanto, os limões continuam a ser amarelos. Neste caso, deve-se assumir que os limões são representados no modelo de cor RGB por valores da componente vermelha, semelhantes à cor verde, e os valores da componente azul muito inferiores às outras duas componentes, i.e., valores próximos de zero. Assim, a cor amarela pode ser representada das seguintes formas:

$$R \cong G; B \cong 0 \vee B \ll R \quad (6.4)$$

$$r + g \cong 1 \quad (6.5)$$

A proposta de segmentação dos limões apenas considera a cor como característica diferenciadora, pelo que importa utilizar componentes de cor que maximizem o contraste da classe dos limões para os restantes elementos. Neste sentido, são criados dois índices baseados nas cromaticidades, com o objetivo de maximizar a diferença da classe dos limões para as restantes.

Com base nos pressupostos enunciados pelas equações (6.4) e (6.5), é criado o *indexA*, de forma a atingir valores mais elevados na presença dos limões, isto é, com baixa cromaticidade azul (*b*) e, cumulativamente, com elevada cromaticidade vermelha (*r*), conforme é representado pela equação (6.6).

Adicionalmente, foi criado o *indexY* que, considerando a componente verde, assume o valor máximo na presença da cor amarela pura, de acordo com a equação (6.7).

$$indexA = \frac{r-b}{r+b} \quad (6.6)$$

$$indexY = r + g - |r - g| - b \quad (6.7)$$

## 6.3 Quantidade de Dados

As imagens utilizadas no treino dos classificadores foram adquiridas em duas plantações diferentes localizadas no concelho de Coimbra, durante o período da tarde. Para representação de condições reais de trabalho, foram adquiridas imagens de limões localizados no exterior da copa e de outros limões localizados no interior da copa das árvores sem incidência direta da luz solar. Foram também adquiridas imagens ao redor das árvores de modo a incluir na base de dados a parte da copa com incidência direta dos raios solares e a parte que estava sob o efeito de sombra.

Do total de imagens adquiridas foram selecionadas 133 imagens em condições diferentes, as quais foram sujeitas a uma ação de segmentação manual de todos os elementos identificados como limões.

Independentemente da abordagem ser por um classificador global ou através de uma estrutura em árvore com diversos classificadores para várias classes, foram extraídas amostras dos vários elementos do *background*. É um processo mais rápido do que a marcação dos limões, por não exigir uma marcação detalhada e rigorosa de todos os seus *pixels*.

Tal como descrito na secção 3.3, as instâncias utilizadas para o treino dos classificadores são obtidas pela média das instâncias de validação. Como nem todos os elementos do *background* estão presentes em todas as imagens, as instâncias de treino destas classes são obtidas pela média das suas componentes em cada região assinalada, enquanto no caso dos limões é assumida a média de todas as regiões de cada imagem. Esta diferenciação visa evitar um desequilíbrio ainda maior nas instâncias de treino das duas classes.

Tabela 6.1. Quantidade de instâncias para treino e validação dos classificadores

	Treino		Validação	
	nº instâncias	% relativa	nº instâncias	% relativa
Limões	133	16,79 %	3202285	59,70 %
Folhagem (verde)	251	31,69 %	1213631	22,62 %
Ramos	187	23,61 %	355911	6,63 %
Elementos cinza	102	12,88 %	431730	8,05 %
Elementos laranja	87	10,98 %	120297	2,24 %
Folhagem seca ou velha	32	4,04 %	40492	0,75 %
Total	792	100 %	5364346	100 %

A Tabela 6.1 apresenta o número de instâncias e o valor relativo de cada classe para o treino e validação dos classificadores. As classes da folhagem e dos ramos apresentam um maior número de elementos por serem as mais abundantes nas imagens adquiridas. No outro extremo, a

folhagem seca é a menos expressiva, por terem sido identificados poucos elementos desta classe nas imagens capturadas. Além disso, devido à semelhança da cor dos limões com a folhagem seca é difícil distingui-los apenas pela cor, como será discutido no capítulo seguinte relativo aos resultados experimentais.

Havendo 133 imagens de resolução 640x480, significa que cerca de 13.3 % dos dados de teste são usados no processo de validação. Relativamente aos limões, por terem sido exaustivamente marcados, são usadas as mesmas instâncias na validação e teste dos modelos.

Comparando com outros ensaios identificados no estado da arte (Tabela 2.1), a quantidade de imagens e instâncias disponíveis para o treino e validação dos classificadores credibiliza os resultados alcançados nos ensaios realizados.

Em determinadas situações, as bases de dados de referência são ampliadas com dados artificiais, isto é, são geradas novas instâncias tendo em consideração as condições ótimas de cada classe, podendo ser afetadas por uma penalização de má classificação superior às restantes instâncias, por, naturalmente, representarem as condições inequívocas que definem uma determinada classe com o intuito de forçar o classificador a aprender a classificá-las corretamente.

Quando se trata de uma análise de padrões ou detecção de objetos, a ampliação da base de dados de referência pode ser realizada pela rotação, sobreposição de ROI ou aplicação de fator de escala às instâncias existentes.

A separação de instâncias por classes pode ser feita através de métodos de aprendizagem não supervisionada, em que são agrupadas (*clustering*) as instâncias que partilham determinadas características, para depois servirem para o treino de classificadores, mas não generalizada a eficiência deste métodos, conforme é reportado em [38].

No que respeita aos dados relativos à detecção, foram usados 58 limões completamente visíveis. Através das 58 ROI originais dos limões foi expandida a base de dados para o treino dos classificadores pela aplicação de uma rotação de 90º em sentido anti-horário, relativamente à base de dados original. Assim, cada classificador pôde ser treinado com 116 limões em posições diferentes. Como exemplos da classe negativa, para o treino dos classificadores, foram usadas as 133 imagens utilizadas na segmentação, após suprimir todos os elementos relativos à classe dos limões.

Para avaliar o desempenho dos modelos de detecção, na fase de validação, são avaliadas 44 imagens (distintas das 133 imagens usadas no processo de segmentação) onde são identificados 92 limões, e as respetivas ROI.

À semelhança do sucedido na criação de uma base de dados de referência para a fase da segmentação, a base de dados para o processo de detecção é criada através de marcação manual, por um operador humano, das regiões da imagem onde se encontram limões. Por apenas bastar criar uma ROI retangular à volta dos limões, este processo é significativamente mais rápido do que na criação da base de dados para a segmentação.

## 6.4 Métricas

Os resultados alcançados com os classificadores são avaliados segundo as métricas apresentadas nesta secção. É de extrema importância clarificar os modelos de avaliação de desempenho para que os resultados alcançados possam ser comparáveis com os realizados noutros ensaios.

Há autores que avaliam o desempenho dos seus métodos através de rácios de precisão (*precision*) e de taxa de TP (*recall*) (equação (6.9)). Outros apresentam o desempenho alcançado por curvas ROC através da relação entre a taxa de TP com a taxa de FP (Falsos-Positivos). Apesar da informação em ambos os casos estar interligada, sem o conhecimento do número de instâncias, bem e mal classificadas, não é possível determinar as outras métricas para uma análise direta de dois resultados apresentados de forma diferente.

Todas estas métricas são determinadas a partir da matriz confusão, uma matriz onde é registado o número de casos de sucesso e de insucesso na classificação, conforme é representado na Fig. 6.4.

		Classe Prevista	
		1	-1
Classe Real	1	Verdadeiros-Positivos (TP)	Falsos-Negativos (FN)
	-1	Falsos-Positivos (FP)	Verdadeiros-Negativos (TN)

Fig. 6.4. Matriz confusão genérica.

Seguem-se as definições das métricas usadas para determinar o desempenho dos classificadores:

- A taxa de TP, ou *TPr*, define a taxa de instâncias bem classificadas da classe positiva, que corresponde à classe dos frutos. Esta métrica é, por vezes, apelidada de *recall*, ou sensibilidade:

$$TPr = \frac{TP}{TP+FN} \quad (6.8)$$

- A taxa de FP, ou *FPr*, representa a razão entre instâncias da classe negativa incorretamente classificadas como pertencentes à classe positiva:

$$FPr = \frac{FP}{FP+TN} \quad (6.9)$$

- A exatidão, designada em inglês por *accuracy* e aqui abreviada por *Acc*, corresponde à percentagem global de instâncias bem classificadas nas suas classes, calculada através da seguinte fórmula:

$$Acc = \frac{TP+TN}{TP+FP+TN+FN} \quad (6.10)$$

- A *F1-score* é uma métrica que relaciona a precisão com a *TPr*, de acordo com a seguinte fórmula:

$$F1 - score = 2 \frac{TPr \times Precision}{TPr + Precision} = \frac{2 \times TP}{2 \times TP + FP + FN} \quad (6.11)$$

- A área sob a curva (*AUC*), do inglês *Area Under the Curve*, está associada à representação gráfica das curvas ROC e representa, em termos percentuais, a área ocupada debaixo da curva. Quanto maior o seu valor, melhor o desempenho da combinação de características de cor.

$$AUC = \sum_{i=1}^n \frac{TPr_i}{n} \quad (6.12)$$

- O tempo médio de processamento por imagem,  $\Delta t_{rel}$ , corresponde à estimativa do tempo necessário para processar uma imagem completa (resolução 640x480) com base no tempo gasto na classificação das instâncias de validação, através da fórmula:

$$\Delta t_{rel} = \frac{640 \times 480 \times \text{tempo de classificação}}{N^{\circ} \text{ total instância classificadas}} \quad (6.13)$$

Com exceção do tempo médio de processamento por imagem, medido em valor absoluto, as restantes métricas são medidas em valor relativo.

Em [109] é apresentada uma análise crítica relativamente ao uso das curvas ROC na avaliação do desempenho dos métodos de decisão.

Relativamente às métricas utilizadas para avaliar o desempenho das metodologias usadas na deteção dos frutos, e de forma a evitar ambiguidades geradas pela aplicação de métricas com a mesma designação no processo de segmentação e deteção, são usados exclusivamente os termos *Precision* (precisão) e *Recall* (sensibilidade) para os resultados referentes à deteção de frutos. Exceção feita aos resultados apresentados no capítulo relativo ao estado da arte, onde são transcritos os resultados publicados pelos respetivos autores, quer para a segmentação quer para a deteção.

Assim, na avaliação dos resultados de deteção poderão ocorrer três acontecimentos:

- Frutos bem detetados (*true-detections*, TD);
- Falha na deteção de um fruto (*missed-detections*, MD);
- Falsas deteções, em que a metodologia sugere a presença de um fruto onde o mesmo não existe (*false-alarms*, FA).

Um fruto é considerado bem detetado (TD) quando a relação da área comum à ROI marcada manualmente e a ROI sugerida pelo classificador for superior a 80 % da área da ROI de menores

dimensões. Para um mesmo fruto, se forem sugeridas pelo classificador duas ROI separadas, e que ambas cumpram o critério anterior, apenas uma é considerada TD e a(s) restante(s) FA, conforme é apresentado na Fig. 6.5.

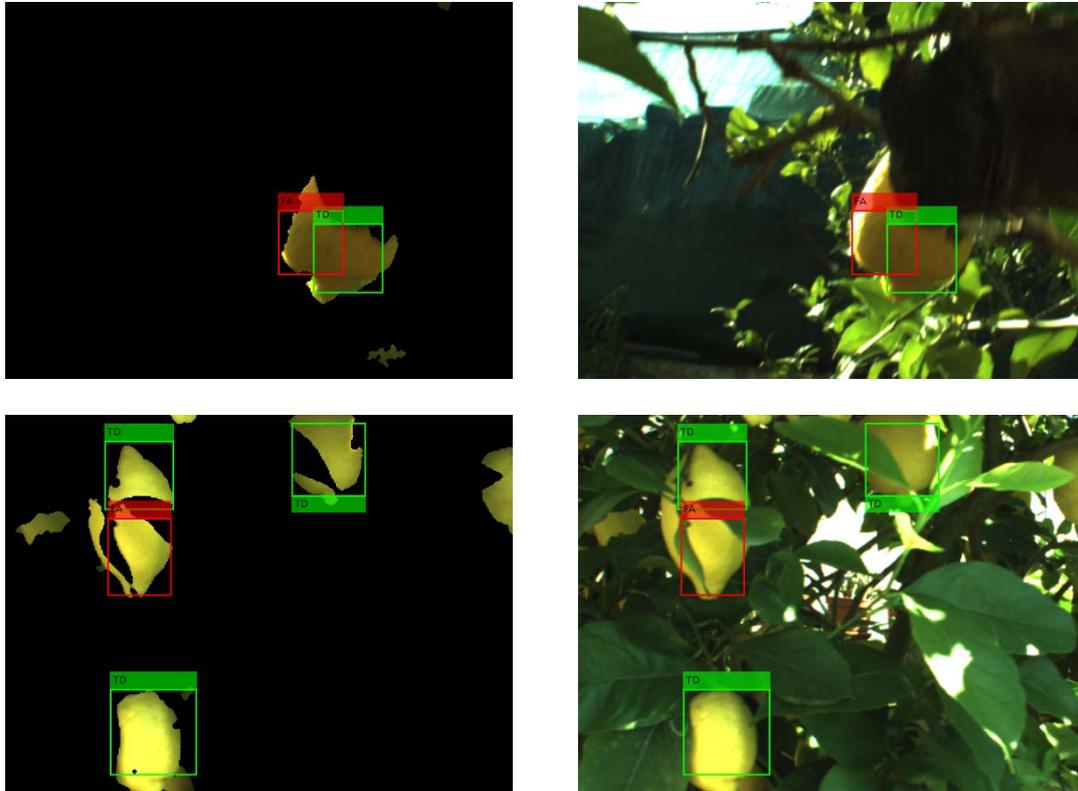


Fig. 6.5. Múltiplas detecções para o mesmo fruto: (esquerda) imagem segmentada; (direita) imagem RGB original.

Na bibliografia revista, também não existe consenso nas métricas a aplicar para avaliar o desempenho das metodologias na detecção dos frutos [10, 35]. Por se considerar haver falta de coerência em aplicar uma taxa de FP, por não existirem elementos TN (teoricamente é zero), são adotadas as métricas apresentadas em [38, 110], definidas pelas equações seguintes:

$$Precision = \frac{TD}{TD+FA} \quad (6.14)$$

$$Recall = \frac{TD}{TD+MD} \quad (6.15)$$

Enquanto a métrica *Recall* permite inferir sobre a capacidade de um método classificar corretamente os frutos identificados na base de dados de referência, a métrica *Precision* indica a robustez desse método relativamente à ocorrência de falsas detecções (FA).

## 7. RESULTADOS EXPERIMENTAIS

Neste capítulo são apresentados os resultados alcançados pelos diferentes métodos de segmentação dos limões, em relação aos restantes elementos presentes nas imagens; na última secção são apresentados os dados relativos à deteção realizada com base nos métodos de *Haar*, HOG e LBP.

Quanto à segmentação inicial, recorreu-se a uma estrutura em árvore na qual a decisão nos nós iniciais é realizada por condições fixas, detalhadas seguidamente, e nos nós posteriores por classificadores SVM.

São apresentados os resultados de diferentes abordagens ao problema de segmentação dos limões: a primeira consiste em avaliar um único classificador SVM global, pela estratégia “um contra todos”, para a segmentação dos limões contra todos os restantes elementos; a segunda consiste em treinar isoladamente vários classificadores SVM, um contra um, com o objetivo de separar os limões de cada classe de elementos do *background*; a terceira abordagem é semelhante à segunda, mas os classificadores SVM são treinados num processo em cadeia e não de forma isolada.

À partida, as vantagens do segundo método avaliado consistem:

- Na obtenção de um modelo de classificação mais robusto e generalista para as cores em questão;
- Na troca da ordem dos classificadores treinados sem alterar a resposta do sistema;
- Na introdução de novos classificadores; ou ainda aplicar um classificador já treinado numa outra aplicação em que seja requerido separar elementos com cores semelhantes às do treino do modelo.

Em contrapartida, quando os modelos são treinados isoladamente pode ser necessário elaborar um modelo mais complexo e, conseqüentemente moroso na classificação, ao tentar classificar corretamente determinadas instâncias que podiam ter sido eliminadas nos nós anteriores.

Tendo em consideração que o espaço temporal destinado a todo o processamento de imagem deve ser inferior a 1 s (um segundo) para que o sistema se torne economicamente viável [3, 4], optou-se também por avaliar o desempenho do sistema em que o treino dos classificadores SVM é realizado num processo em cadeia. Nesta metodologia, os classificadores continuam a ser treinados numa estratégia de “um contra um”, mas apenas com as instâncias que não sejam eliminadas nos nós de classificação anteriores. Pretende-se com esta estratégia obter classificadores mais simples e rápidos, mas corre-se o risco de provocar grandes desequilíbrios no número de instâncias de treino das duas classes nos últimos nós da árvore de decisão, conforme esquematizado na Fig. 4.4.

O número de nós da árvore de decisão será diferente entre a primeira abordagem e as restantes, por se tratar de avaliar um classificador SVM segundo a estratégia de “um contra todos”. Porém, os dois primeiros nós consistem em condições fixas para conseguir fazer uma eliminação grosseira dos *pixels* que não se assemelham à cor amarela dos limões.

Assim, o primeiro nó consiste na condição de classificar imediatamente como *background* todos os *pixels* que apresentem valores na componente Hue inferior a  $15^\circ$  ou superior a  $105^\circ$ ; esta medida foi adotada a partir do valor da cor amarela em condições ideais na componente Hue, correspondente a  $60^\circ$ , com uma margem de erro de  $45^\circ$  por defeito e excesso.

O segundo nó contém uma condição aplicada sobre o *indexY*, onde são consideradas como *background* todas as respostas desta componente inferior ou igual a  $1/3$ .

Com estas duas condições simples, são eliminados todos os *pixels* que apresentem características que não estejam próximas das condições que caracterizam os limões. Os *pixels* saturados (de cor branca ou preta) são também filtrados nestes dois nós iniciais.

Esta abordagem é escolhida com o objetivo de minimizar o tempo de processamento na segmentação. Quanto mais elementos forem classificados como não limões nos primeiros níveis da árvore, por classificadores simples e de rápida execução, mais rápida é expectável a execução global, pois evita-se gastar recursos com estes elementos em avaliações de maior complexidade nos níveis seguintes.

Depois dos nós de classificação, com base em valores fixos pré-estabelecidos (primeiros dois níveis da árvore de decisão), segue-se uma avaliação com classificadores SVM. Nestes níveis, a decisão de classificar os *pixels* como pertencentes à classe dos limões ou noutra é feita através de modelos pré-treinados, conforme é apresentado nas próximas secções deste capítulo.

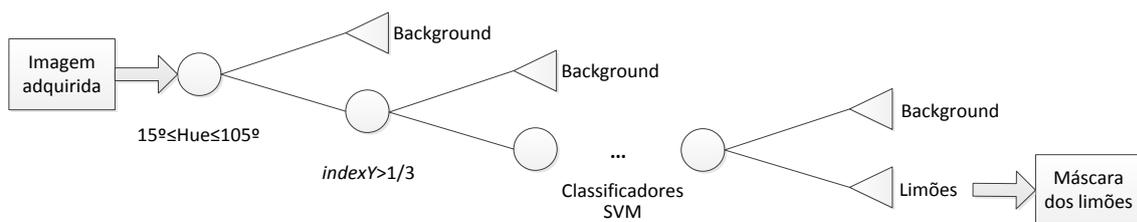


Fig. 7.1. Estrutura base da árvore de decisão.

Assim, a estrutura base da árvore de decisão para as abordagens avaliadas é a representada na Fig. 7.1, sendo os resultados da sua aplicação apresentados na Tabela 7.1. Nesta tabela, os atributos FP referem-se ao número total de instâncias do *background* que não são imediatamente eliminadas nos nós e que seguem para análise pelos classificadores SVM. Os resultados relativos ao treino apenas têm influência no referente aos classificadores em cadeia (secção 7.3).

A ordem dos classificadores é estabelecida em função das classes predominantes, ou seja, são prioritários os modelos que eliminam uma maior quantidade de elementos. Desta forma evita-se gastar recursos no processamento consecutivo de um elevado número de *pixels* para serem apenas eliminados no último nó da árvore de decisão.

Tabela 7.1. Resultados da segmentação dos primeiros níveis da árvore de decisão.

	Treino				Validação			
	TP	<i>TPr</i>	FP	<i>FPr</i>	TP	<i>TPr</i>	FP	<i>FPr</i>
Hue & IndexY	133	100%	553	83,92%	3185631	99,48%	1621013	74,98%

Em todos os ensaios realizados foi avaliada a resposta dos classificadores obtidos considerando a existência de três valores de ruído (*outliers*) no total das instâncias de treino: 0 %; 2 %; e 3.5%. Esta abordagem é possível pelo fato de todas as simulações se realizarem num *cluster* HPC e dada a disponibilidade de tempo de computação, caso contrário seria recomendável definir um único valor para a percentagem de ruído existente na amostra para todas as simulações.

São ainda definidos empiricamente os valores máximos admissíveis de  $\Delta t_{rel}$  e *FPr* para o treino de cada subclasse do *background*, para evitar que os métodos de otimização convirjam para um máximo local cuja resposta do modelo de decisão não seja admissível em termos práticos.

Depois de concluído o treino dos classificadores, podem ser avaliadas as respostas para  $\Delta t_{rel}$  e *FPr* inferiores ou iguais às estabelecidas no momento do treino, em função das necessidades específicas de cada caso.

Os resultados apresentados neste capítulo são referentes a um valor limite de  $\Delta t_{rel}$  e *FPr* a serem explicitados em cada caso. Mas, reforça-se que com a presente metodologia outras respostas podem ser obtidas, dependendo dos requisitos definidos para cada situação, ou seja, são aqui apresentados os melhores resultados obtidos relativos às parametrizações do SVM, às combinações de características avaliadas e aos critérios de  $\Delta t_{rel}$  e *FPr* definidos. Pelos motivos enunciados, não se pode afirmar que a melhor resposta obtida represente a melhor solução absoluta, mas sim relativamente às condições impostas.

A produção das curvas ROC é executada apenas com as configurações com melhor desempenho para determinado *FPr*, ou seja, apenas é mostrada a melhor taxa de TP para cada *FPr* de acordo com o algoritmo representado na Fig. 7.2. Sempre que não seja explicitado na legenda das figuras associadas às curvas ROC, ou no texto que descreve essas figuras a que conjunto de dados é relativo, deve-se assumir que, por defeito, é relativo aos dados de validação do classificador. Quando forem apresentadas curvas ROC relativas aos dados de teste, deve haver uma referência explícita à origem dos dados.

```

Parâmetros de Entrada: Matriz (results_eval) com os resultados a analisar
Parâmetros de Saída: Matriz (results_resume) com a configuração de melhor TPr
para cada FPr

Ordena linhas da matriz results_eval por ordem crescente da coluna relativa ao
ruído admissível na amostra (outliers)

Ordena (novamente) linhas da matriz results_eval por ordem crescente da coluna
relativa FPr

% As alterações na ordenação da linha da matriz results_eval obriga a que sejam
% analisadas primeiro as configurações com FPr mais baixo e treinados com
% menor percentagem de outliers

results_resume ← ∅
j ← 1
tmp_tpr ← 0
Enquanto j ≤ número de linhas de results_eval
    tmp_idx ← determina índice das linhas de results_eval cujo FPr é igual ao
    FPr de índice j de results_eval
    local_max ← determina o valor máximo de TPr nos índices tmp_idx da
    matriz results_eval
    Se local_max > tmp_tpr
        max_idx ← determina o índice de results_eval correspondente ao
        valor máximo de TPr nos índices tmp_idx da matriz results_eval
        Acrescenta à matriz results_resume a configuração de índice
        max_idx da matriz results_eval
        tmp_tpr ← local_max
    Fim de se
    j ← Soma 1 ao último índice de tmp_idx
Fim de enquanto

```

Fig. 7.2. Algoritmo de seleção das melhores combinações para elaboração de curvas ROC.

Este algoritmo é executado por duas vezes: uma primeira para determinar as melhores configurações por cada combinação de características avaliadas; e depois, com os resultados de todas as combinações, são determinadas as melhores respostas para um determinado número de características analisadas.

Desta forma, também é facilitada a leitura das curvas ROC, ao serem representadas as configurações com melhor rácio  $TPR/FPr$ , descartando as configurações cuja taxa de TP é inferior às anteriormente (para  $FPr$  mais baixas) obtidas.

No Anexo B são apresentadas as combinações de características de cor com AUC mais elevada.

## 7.1 Classificador SVM global

Devido à incerteza existente em torno da melhor forma de segmentar os limões dos restantes elementos das imagens, recorrendo apenas a características de cor, em primeiro lugar é avaliado o desempenho de um classificador global treinado segundo a estratégia “um contra todos”. Isto significa que são treinados e avaliados classificadores assumindo como classe positiva a dos limões e o conjunto de todas as outras classes a formar a classe negativa.

Para compreender a evolução da resposta do sistema às configurações do modelo de decisão foi inicialmente avaliado o desempenho de um classificador global, avaliando todas as combinações de cinco características (*features*) distintas nas seguintes condições:

- $\Delta t_{rel} \leq 0.75$  s (750 ms)
- $FPr \leq 6\%$

Dos métodos de otimização enunciados na seção 4.1, apenas não é aplicado, neste caso, o processo iterativo de geração de novas combinações.

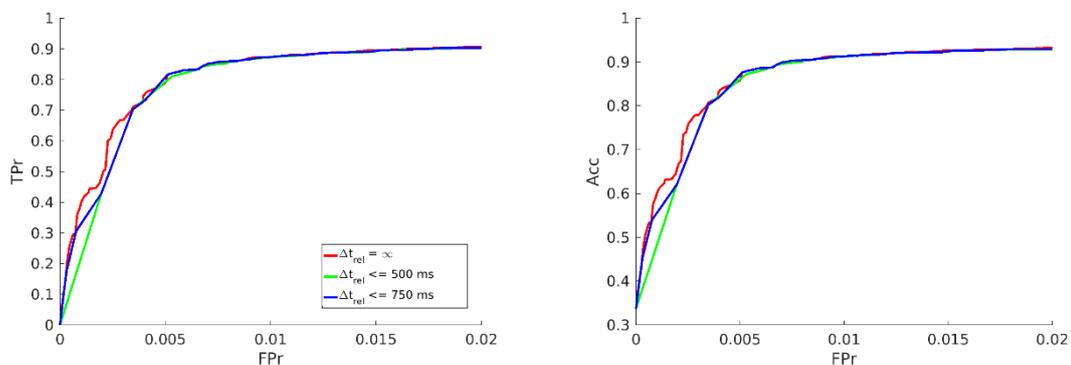


Fig. 7.3. Desempenho nos dados de validação de classificadores SVM global treinados com cinco características: (esquerda) curva ROC; (direita) Acc correspondente.

A Fig. 7.3 contém os resultados alcançados para diversos limites de tempo definidos após o treino dos classificadores. A representação a vermelho é referente à situação em que não existe qualquer restrição temporal. Apesar de ser definido um tempo máximo de processamento por imagem, depois de um classificador ser treinado e o requisito temporal não ser cumprido, são mantidos os seus resultados na base de dados. Para taxas de FP superiores a 0.5 %, as três

respostas representadas são idênticas, sendo preferível optar pela de menor tempo médio de processamento por imagem.

Para  $\Delta t_{rel} \leq 500$  ms, uma das configurações possíveis permite atingir uma taxa de TP de 88.1 % para  $FPr = 1.2$  %. Apesar de não ser possível fazer uma comparação direta dos resultados aqui alcançados com os de outros autores, pelo fato dos frutos e das metodologias aplicadas serem diferentes, contudo, comparando com os resultados resumidos na tabela Tabela 2.1, é atingida uma taxa de TP superior à referida em [14], mantendo uma  $FPr$  mais baixa. Este resultado preliminar na segmentação dos limões, com um sistema de aquisição de imagens simples, supera os alcançados com um sistema multiespectral usando sistemas de aprendizagem não supervisionados na classificação de uvas [12], ou até os reportados em [17], que recorre a classificadores SVM para a classificação de um cenário vitivinícola.

Mantendo a estratégia “um contra todos”, considerando a metodologia de seleção de características proposta,  $\Delta t_{rel} \leq 500$  ms e  $FPr \leq 6$  %, foram alcançados os resultados apresentados na Fig. 7.4.

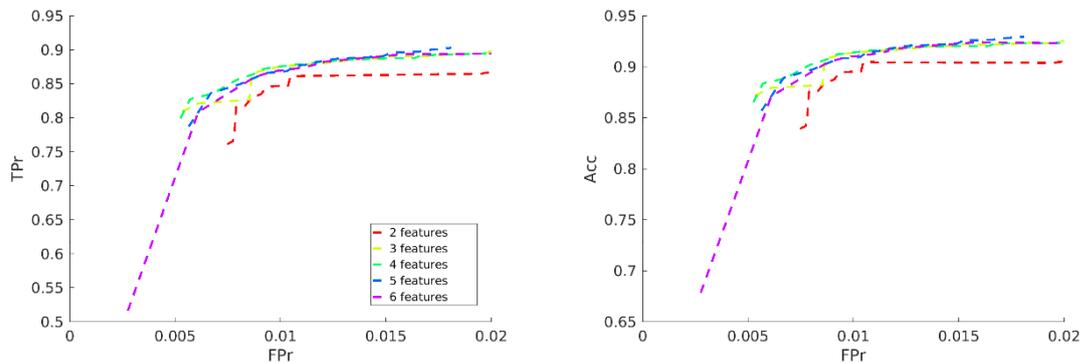


Fig. 7.4. Desempenho nos dados de validação de classificadores SVM global para diferentes números de características e  $\Delta t_{rel} \leq 500$  ms: (esquerda) curva ROC; (direita) Acc correspondente.

Pela análise da Fig. 7.4 verifica-se um crescimento generalizado no rácio  $TPr/FPr$  ao aumentar o número de características analisadas, de duas até quatro, para uma  $FPr < 1.5$  %. A partir desse ponto, a  $TPr$  apresenta um crescimento diminuto.

Tal como já fora mencionado nesta dissertação, um maior número de características analisadas não é sinónimo de um maior desempenho por parte dos modelos de decisão.

Alisando com detalhe a curva ROC, ilustrada na Fig. 7.5, a combinação de cinco características apresenta a maior taxa de TP registada neste modelo de avaliação.

A Fig. 7.6 compara a resposta obtida pelos classificadores treinados com cinco características fixas com os classificadores das melhores combinações de quatro características determinados pelo método iterativo proposto. A curva ROC mostra um desempenho idêntico nas duas abordagens, e o tempo médio de processamento por imagem varia muito em função da  $FPr$  pretendida.

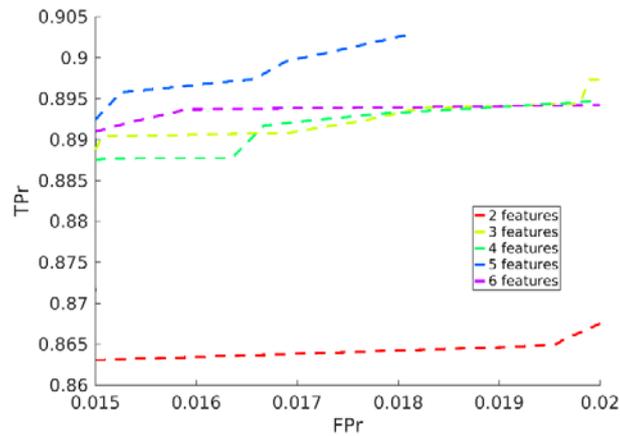


Fig. 7.5. Detalhe da curva ROC da Fig. 7.4.

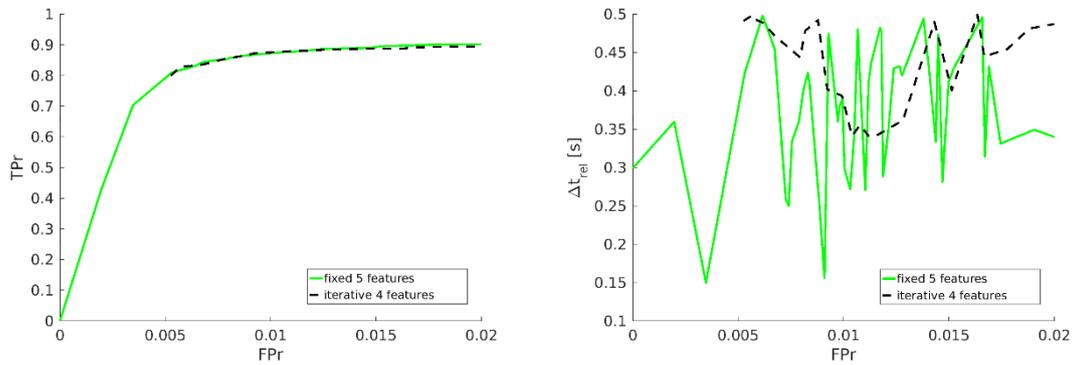


Fig. 7.6. Comparação do desempenhos de classificadores SVM global ( $\Delta t_{rel} \leq 500$  m) com cinco características fixas vs quatro características determinadas pelo método iterativo: (esquerda) curva ROC; (direita)  $\Delta t_{rel}$  correspondente.

Não se pode afirmar que uma determinada configuração seja genericamente melhor do que outra. A seleção deve ser feita em função das várias métricas e daquela que melhor preencher os requisitos do projeto. Para averiguar com mais detalhe o desempenho destas duas abordagens é necessário averiguar a sua resposta nos dados de teste.

### 7.1.1 Resultados dos Testes

Os resultados obtidos durante a validação dos modelos são indicativos do desempenho do método de classificação em condições reais, isto é, no processamento de imagens completas. Assim, foram avaliadas duas configurações distintas do classificador SVM apresentadas na Tabela 7.2.

Tabela 7.2. Comparação dos resultados de validação e teste com diferentes configurações do classificador SVM global.

		Características	$TPr$	$FPr$	$\Delta_{rel}$
SVM Global 1	Validação	$g b Y Sat$	88.77 %	1.51 %	400.6 ms
	Teste		83.18 %	3.57 %	89.5 ms
SVM Global 2	Validação	$g b Y Sat indexA$	90.28 %	1.81 %	390.3 ma
	Teste		90.94 %	6.02 %	124.6 ms

Analisando a Tabela 7.2 identifica-se uma discrepância nos valores da métrica  $TPr$  na etapa de validação e teste. Uma vez que o número de instâncias da classe dos limões é o mesmo nestas duas etapas, a diferença deve-se ao facto de, durante a validação, as métricas são determinadas em função do conjunto total de dados e, durante o teste é avaliada cada métrica por imagem, sendo os valores relativos ao teste apresentados na Tabela 7.2, com a média dos valores obtidos nas 133 imagens avaliadas. Claro que caso fossem apresentados os valores absolutos para a etapa de teste, os resultados obtidos relativos à classe dos limões (neste caso a  $TPr$ ) seriam iguais aos obtidos durante a validação.

Esta discrepância é refletida nos dados de teste apresentados nas restantes metodologias avaliadas e apresentadas nesta secção. Pelos motivos enunciados e para providenciar um maior detalhe sobre os resultados alcançados, são apresentados diagramas de extremos e quartis relativamente a uma configuração selecionada.

Das configurações apresentadas na Tabela 7.2 selecionou-se aquela que permite obter uma taxa de TP mais elevada (SVM Global 2). Porém, a máscara resultante do processo de classificação contém ruído, como ilustrado na Fig. 7.34 para a metodologia de classificadores SVM em cadeia, que pode ser atenuado pelas operações morfológicas de erosão e dilatação.

Usando um elemento estruturante com forma de losango (apelidado de *diamond* no Matlab) e cujo raio foi feito variar entre 3 e 8 (*pixels*) para a erosão e, seguidamente, aplicando-se a operação de dilatação com o raio a variar entre 3 e 9 (*pixels*), obtiveram-se os resultados médios finais ilustrados na Fig. 7.7.

A configuração correspondente a  $TPr = 93.14\%$  e  $FPr = 2.34\%$  é obtida pela aplicação de um processo de erosão com um raio de 4 *pixels* sobre a máscara proveniente do classificador SVM, seguindo-se uma dilatação com um raio de 7 *pixels*. A Fig. 7.8 apresenta o diagrama de extremos e quartis para esta configuração, proporcionando um maior detalhe sobre os resultados reais desta metodologia. As restantes parametrizações do classificador SVM selecionado são ainda descritas na Tabela 7.3.

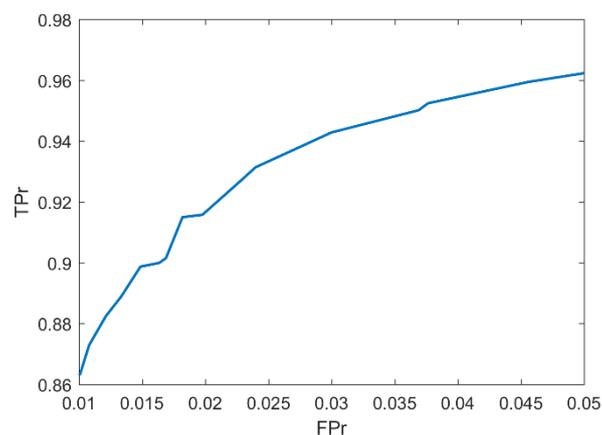


Fig. 7.7. Curva ROC das operações morfológicas avaliadas após o classificador SVM global nas imagens de teste.

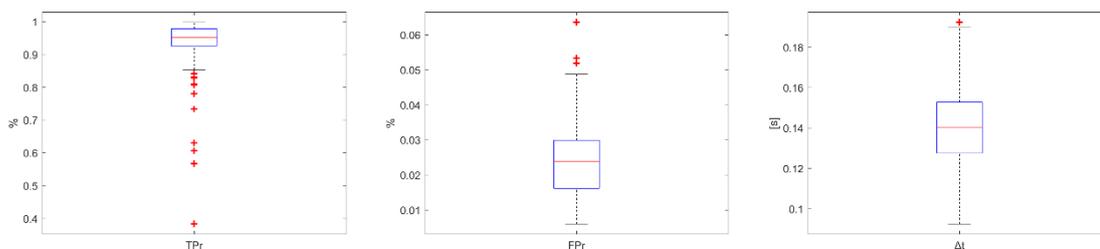


Fig. 7.8. Diagrama de extremos e quartis relativo à configuração selecionada para o classificador SVM global nos dados de teste: (esquerda) TPr; (centro) FPr; (direita)  $\Delta t$ .

Tabela 7.3. Configuração selecionada para o classificador SVM global.

	SVM Global 2
$\sigma$	3.2961
Características	<i>g b Y Sat indexA</i>
$C$	2.5119
$c$	0.3000

## 7.2 Classificadores SVM independentes

Como o *background* possui elementos de características muito distintas, pretende-se avaliar se um modelo de classificação otimizado para distinguir classes específicas apresenta um melhor desempenho global do que o classificador SVM global analisado na secção anterior.

Como os classificadores são treinados individualmente, pode-se trocar a sua ordem de aplicação mantendo o mesmo rácio  $TPr/FPr$ . Contudo, obtêm-se globalmente tempos médios de processamento por imagem mais reduzidos se os classificadores dos primeiros níveis considerarem uma maior quantidade de *pixels* como *background*, evitando assim que estes sejam sucessivamente processados pelos classificadores dos níveis seguintes. Este raciocínio é válido para uma estratégia de processamento sequencial; numa abordagem com computação paralela poderá ser empregue outra estratégia.

As classes predominantes nas imagens são a da folhagem verde e a dos ramos, pelo que importa que sejam os classificadores treinados para essas classes a ocuparem os primeiros níveis da árvore de decisão referentes a classificadores SVM. Assim, a estrutura da árvore utilizada é a representada na Fig. 7.9.

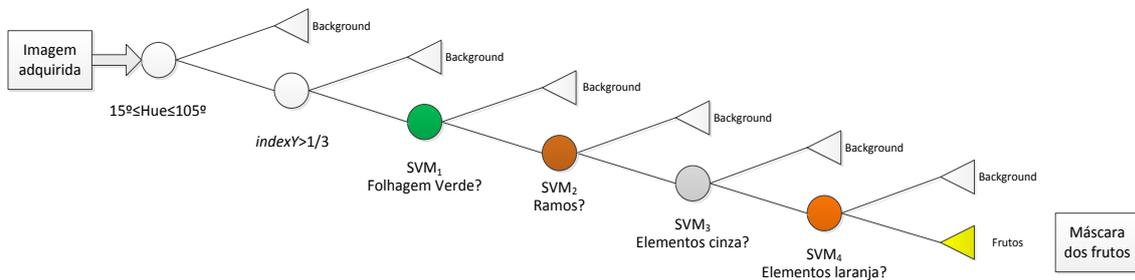


Fig. 7.9. Estrutura da árvore de decisão usada na estratégia “um contra um”.

Como os modelos de decisão são treinados individualmente, os gráficos de desempenho relativos aos dados de validação apresentados na seção seguinte não consideram os efeitos provocados pelos níveis de classificação anteriores, ou seja, são avaliadas todas as instâncias dos limões e de cada subclasse, independentemente de serem classificadas como *background* nos níveis anteriores da árvore de decisão.

## 7.2.1 Limões vs folhagem verde

A folhagem (verde) é a subclasse do *background* com maior presença nas imagens, pelo que importa classificar corretamente os TN sem prejudicar em demasia a  $TPr$ , pois os elementos dos limões serão novamente avaliados pelos classificadores dos níveis posteriores, onde também haverá ocorrência de FN.

Importa também precaver um tempo médio excessivo de processamento por imagem, pois este será o primeiro classificador SVM e terá que analisar uma maior quantidade de instâncias do que os classificadores posteriores. Assim, o presente classificador foi treinado nas seguintes condições:

- $\Delta t_{rel} \leq 0.75$  s (750 ms)
- $FPr \leq 3$  %

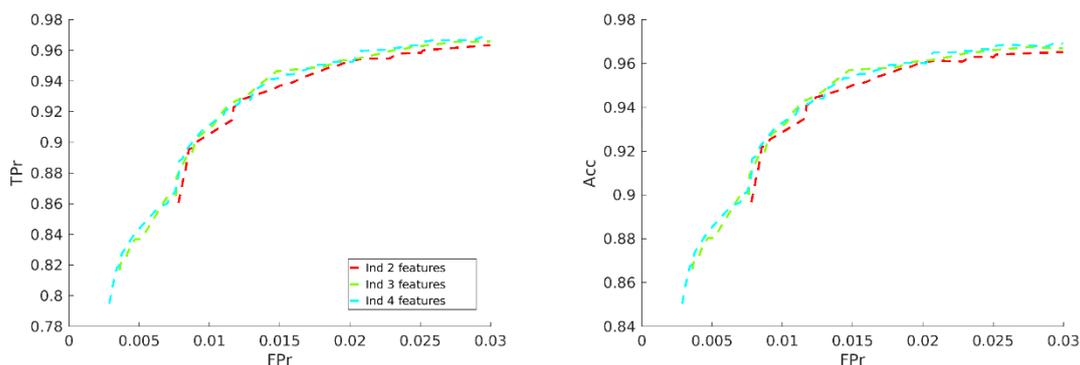


Fig. 7.10. Desempenho de classificadores SVM independentes para folhagem até quatro características e  $\Delta t_{rel} \leq 750$  ms: (esquerda) curva ROC; (direita) Acc correspondente.

A Fig. 7.10 reporta o melhor rácio  $TPR/FPr$  obtido para combinações de diferentes números de características. Não foram testadas combinações com mais de quatro características, por se verificar que não existem melhorias significativas ao passar de três para quatro características analisadas e pela experiência proveniente do treino do classificador global (Fig. 7.4), em que o aumento do número de características originou classificadores com pior desempenho.

A escolha da melhor configuração é subjetiva para cada aplicação; para este classificador importa:

- Manter a  $TPR$  elevada ( $>95\%$ ), pois será negativamente afetada nos classificadores dos níveis seguintes da árvore de decisão;
- $FPr$  baixa por ser a subclasse predominante do *background*;
- Assegurar um  $\Delta t_{rel}$  baixo.

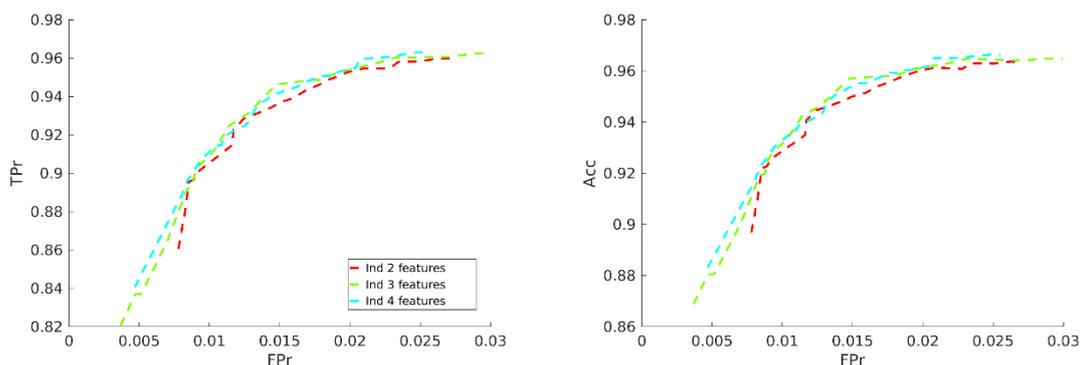


Fig. 7.11. Desempenho de classificadores SVM independentes para folhagem até quatro características e  $\Delta t_{rel} \leq 500$  ms: (esquerda) curva ROC; (direita) Acc correspondente.

O melhor rácio  $TPR/FPr$  pode ser determinado pelas métricas  $Acc$  (6.10) ou pelo  $F1$ -score (6.11); mas, relacionar os seus resultados com a variável tempo, é relativo no que concerne à relevância

dada a cada atributo. Por ser o primeiro classificador SVM é ainda avaliado o seu desempenho, admitindo que  $\Delta t_{rel} \leq 500$  ms, com o objetivo de averiguar o impacto desta alteração, cujos resultados são apresentados na Fig. 7.11, onde as taxas de TP superiores a 95 % são atingidas a partir dos 2 % de *FPr*.

Comparando os resultados da Fig. 7.11 com os da Fig. 7.10, para *FPr* > 2% há uma diferença média de 1 % relativa à *TPr*, com os melhores rácios a serem alcançados para a configuração de  $\Delta t_{rel} \leq 750$  ms. Adicionalmente, o  $\Delta t_{rel}$  é também um parâmetro relevante na seleção da combinação a integrar a árvore de decisão, sendo apresentados na Fig. 7.12 os seus valores para cada configuração.

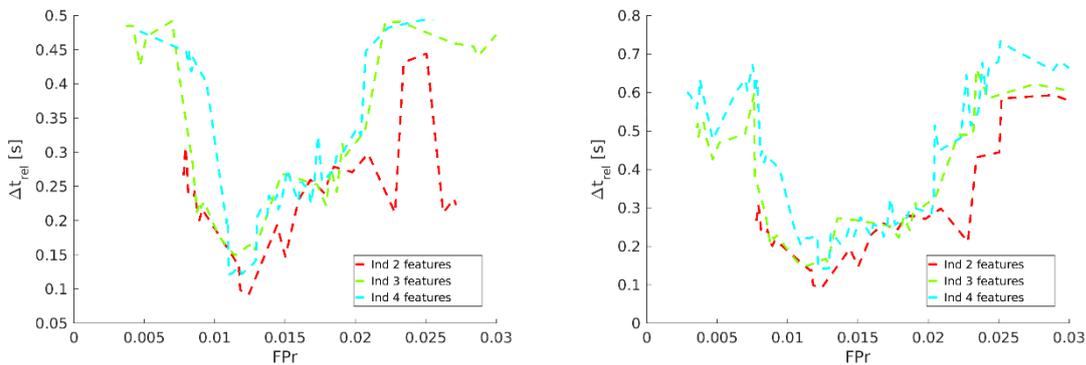


Fig. 7.12. Tempo médio de processamento de cada configuração de classificadores SVM independentes para a folhagem: (esquerda)  $\Delta t_{rel} \leq 500$  ms; (direita)  $\Delta t_{rel} \leq 750$  ms.

Na Fig. 7.12 é possível identificar que, para *FPr* > 2% existem configurações de apenas duas características cujo  $\Delta t_{rel}$  é muito inferior à generalidade dos restantes casos. A opção por uma destas configurações dependerá dos restantes classificadores SVM da árvore de decisão e ou do seu desempenho nos dados de teste.

## 7.2.2 Limões vs ramos

O treino do classificador independente para a subclasse dos ramos foi realizado para:

- $\Delta t_{rel} \leq 0.5$  s (500 ms);
- $FPr \leq 4\%$ .

Para este classificador foram avaliadas combinações de, no máximo, três características, devido a haver melhor rácio *TPr*/*FPr* na combinação com apenas duas características: para *FPr* < 4%, conforme ilustrado na Fig. 7.13. Além disso consegue-se um  $\Delta t_{rel}$  de, sensivelmente, 155 ms para *TPr* > 96 %, tal como ilustrado na Fig. 7.14.

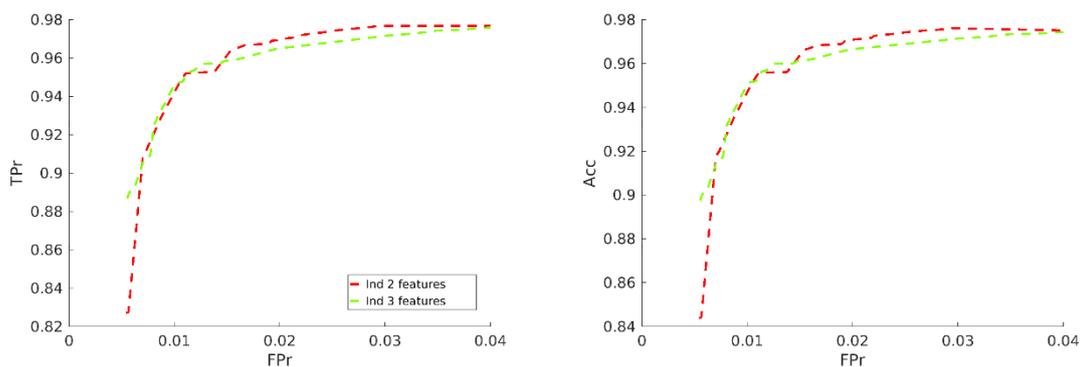


Fig. 7.13. Desempenho de classificadores SVM independentes para ramos até três características e  $\Delta t_{rel} \leq 500$  ms: (esquerda) curva ROC; (direita) Acc correspondente

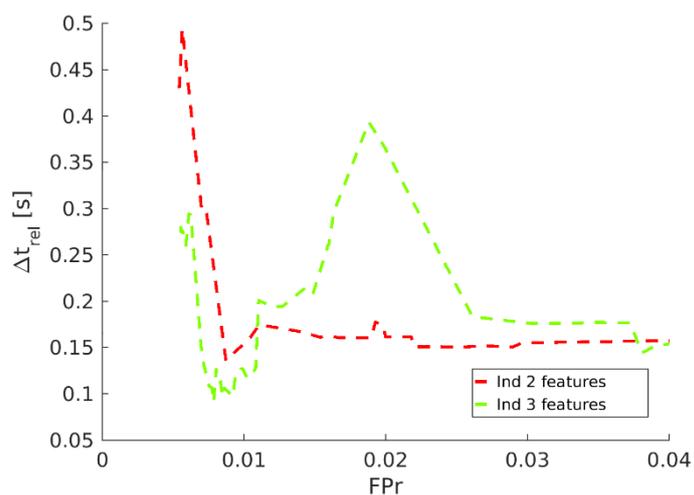


Fig. 7.14.  $\Delta t_{rel}$  dos classificadores SVM independentes para ramos de até três características

### 7.2.3 Limões vs elementos cinza

A subclasse dos elementos cinza representa uma classe particular do conjunto de imagens existentes para o treino do modelo de decisão, relativa a objetos em cimento presentes nas imagens. O treino do classificador independente para a subclasse dos elementos cinza foi realizado para:

- $\Delta t_{rel} \leq 0.75$  s (750 ms);
- $FPr \leq 8$  %.

Por esta subclasse ser um caso particular, não haverá uma percentagem significativa destes elementos nas imagens, pelo que é admissível uma  $FPr$  e  $\Delta t_{rel}$  mais elevada do que nos classificadores anteriores.

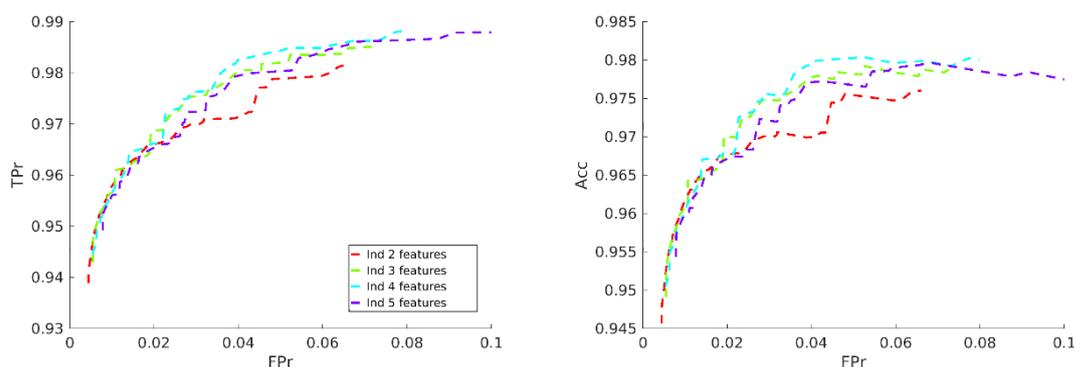


Fig. 7.15. Desempenho de classificadores SVM independentes para elementos cinza até cinco características e  $\Delta t_{rel} \leq 750$  ms: (esquerda) curva ROC; (direita) Acc correspondente.

Para o presente classificador, para uma taxa de TP superior a 98 % (Fig. 7.15), o melhor desempenho é atingido com combinações de quatro características e, mais uma vez, verifica-se que o uso de mais características pode provocar um efeito negativo no desempenho da classificação.

Relativamente ao tempo médio de processamento por imagem, ilustrado na Fig. 7.16, há a tendência generalizada das combinações de duas características permitirem uma resposta com menor latência e, para as restantes combinações verificarem-se picos de comportamento. Para este classificador é prevista uma menor influência do tempo médio de processamento na resposta global do método de classificação, por ser um dos últimos classificadores onde é esperado haver poucas instâncias a processar, devido a serem filtradas nos níveis anteriores.

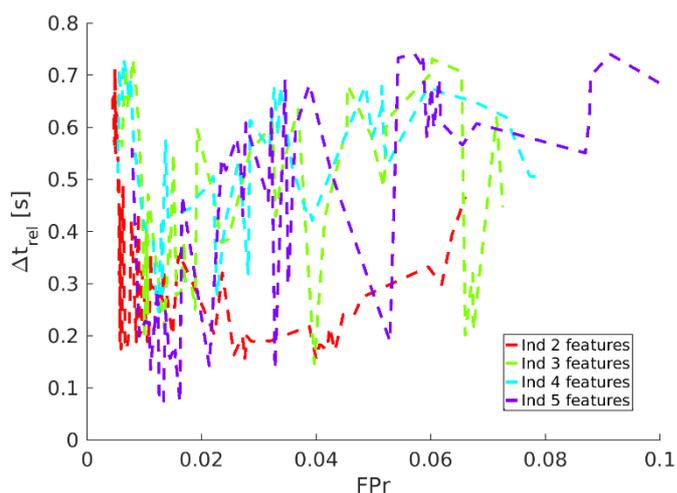


Fig. 7.16.  $\Delta t_{rel}$  dos classificadores SVM independentes para elementos cinza de até cinco características.

## 7.2.4 Limões vs elementos laranja

Os elementos laranja também fazem parte de uma subclasse particular do conjunto de imagens existentes, visando obter uma taxa global elevada de TP, sendo admitida uma  $FPr$  no treino dos classificadores superior à definida no treino das classes da folhagem e dos ramos. Pois, só avaliando o rácio  $TPr/FPr$  nos dados de validação, é possível selecionar a configuração que, aparentemente, é considerada ideal, mas só confirmada pelos dados de teste. Assim, este classificador é treinado nas condições seguintes:

- 1)  $\Delta t_{rel} \leq 0.5$  s (500 ms);
- 2)  $FPr \leq 6$  %.

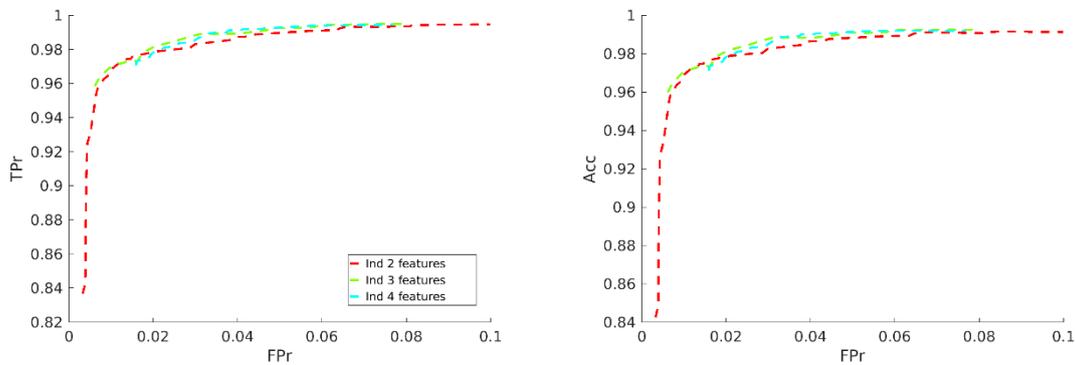


Fig. 7.17. Desempenho de classificadores SVM independentes para elementos laranja até quatro características e  $\Delta t_{rel} \leq 500$  ms: (esquerda) curva ROC; (direita) Acc correspondente.

A Fig. 7.17 ilustra o desempenho deste classificador, onde é perceptível uma melhoria nos resultados ao usar mais do que duas características. Para facilitar a interpretação dos resultados alcançados pelas combinações de três e quatro características, a Fig. 7.18 mostra com maior detalhe a curva ROC e o  $\Delta t_{rel}$  para taxas de FP entre 4 % e 6 %. Para esta configuração, o melhor rácio  $TPr/FPr$  é obtido para as combinações de quatro características, pelo que só se houver limites temporais de execução muito restritivos é que deve ser selecionada outra configuração entre as soluções apresentadas nestas figuras e em função dos tempos apresentados na Fig. 7.20.

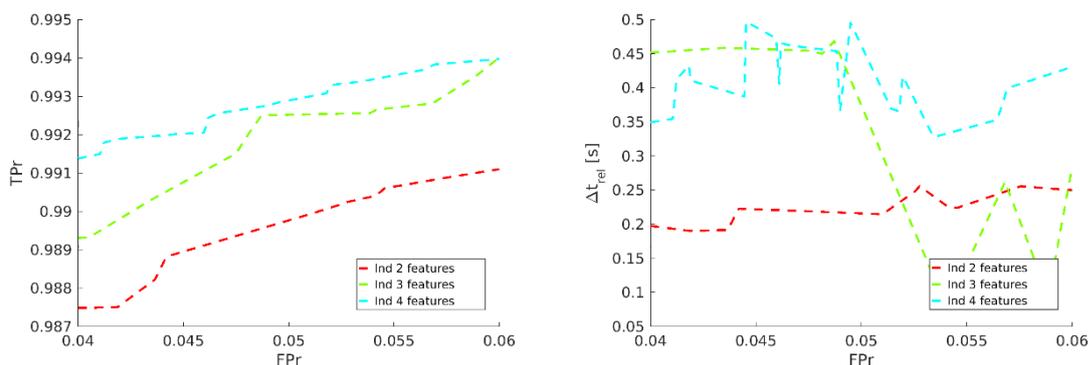


Fig. 7.18. Desempenho detalhado dos classificadores SVM independentes para elementos laranja para  $4\% \leq FPr \leq 6\%$ : (esquerda) curva ROC; (direita)  $\Delta t_{rel}$  correspondente.

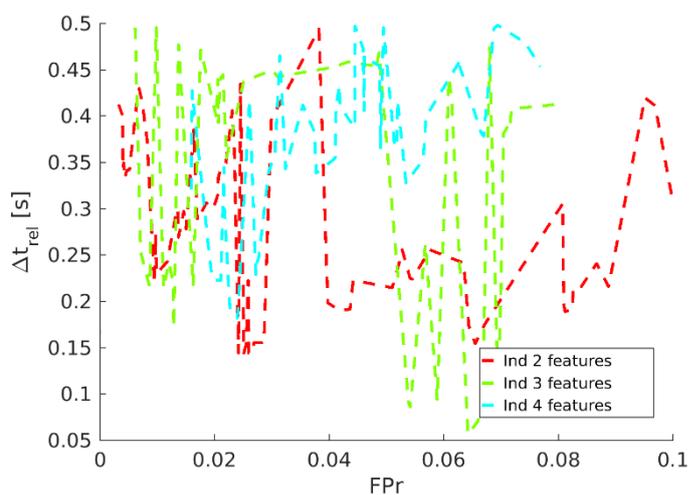


Fig. 7.19.  $\Delta t_{rel}$  dos classificadores SVM independentes para elementos laranja de até quatro características.

## 7.2.5 Limões vs folhagem seca

A folhagem seca representa a subclasse do *background* com maior semelhança de cor com os limões, sendo a sua presença nas imagens adquiridas muito baixa. Com base nestes pressupostos, e por ser o último classificador SVM, foram analisados classificadores SVM treinados da seguinte forma:

- $\Delta t_{rel} \leq 0.8$  s (800 ms);
- $FPr \leq 100\%$ .

Conforme representado na Fig. 7.20, para não afetar a taxa de TP, os classificadores treinados resultam numa *FPr* elevada. Motivado pela baixa latência exigida no processamento das imagens,

optou-se por desconsiderar esta subclasse, pois, devido à sua presença remota nas imagens, ao rácio  $TPR/FPr$  desadequado, e à sobrecarga de tempo gasto em processamento, não justificam, nestas condições, a aplicação deste classificador.

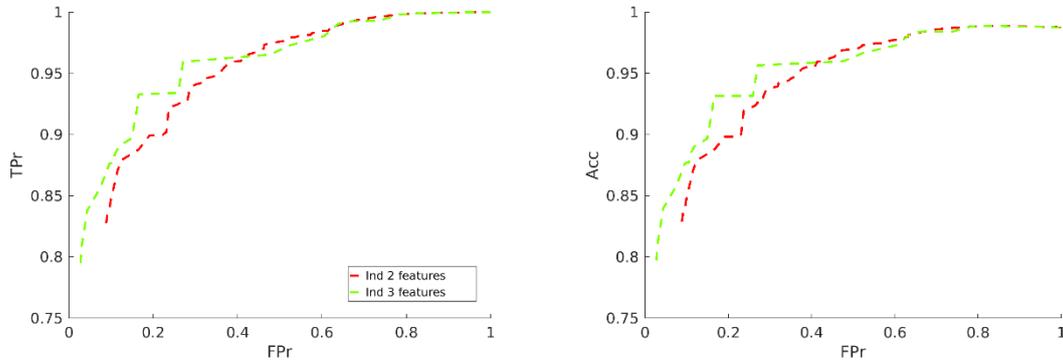


Fig. 7.20. Desempenho de classificadores SVM independentes para folhagem seca até três características e  $\Delta t_{rel} \leq 800$  ms: (esquerda) curva ROC; (direita) Acc correspondente.

## 7.2.6 Resultados de teste

Para avaliar o desempenho global desta metodologia de classificação nas imagens de teste foram pré-selecionadas duas configurações possíveis para os classificadores SVM, das quatro classes avaliadas, apresentadas na Tabela 7.4. Avaliando todas as combinações possíveis das configurações pré-selecionadas, estas perfazem um total de 16 combinações distintas.

Este tipo de análise é realizado devido a, até então, haver unicamente informações relativas ao desempenho do classificador atuando isoladamente e não integrado na árvore de decisão.

A Fig. 7.21 apresenta a curva ROC das diferentes combinações nos dados de teste. Para facilitar a leitura, também neste caso são omitidas as combinações que apresentam taxas de FP superiores aos resultados anteriores e, cumulativamente, uma pior  $TPR$ .

Nos resultados alcançados verificou-se ainda que, independentemente da configuração selecionada para o classificador dos elementos cinza, a resposta global mantinha-se inalterada. É possível que as instâncias que uma e outra configuração classifiquem de forma diferente e sejam filtradas nos níveis anteriores da árvore de decisão, ou até que este classificador tenha um impacto muito reduzido na resposta global. Na avaliação seguinte, das operações morfológicas a aplicar, serão avaliadas combinações com classificadores das quatro classes do *background* e as mesmas combinações com a exclusão do classificador dos elementos cinza, com o objetivo de dissipar as dúvidas existentes quanto ao impacto deste classificador no desempenho global da metodologia proposta.

Tabela 7.4. Configurações pré-selecionadas dos classificadores SVM independentes.

Cód. Configuração	Características	TPr	FPr	$\Delta t_{rel}$ [s]
Folhagem C1A	$g Y Hue Sat$	95.95 %	2.07 %	0.448
Folhagem C1B	$g Y index Y index A$	96.30 %	2.47 %	0.494
Ramos C2A	$b Y$	97.19 %	2.20 %	0.151
Ramos C2B	$b Y$	97.67 %	2.96 %	0.155
Elem. Cinza C3A	$g b Cb Sat$	98.48 %	5.18 %	0.571
Elem. Cinza C3B	$g b Cb Sat$	97.95 %	3.42 %	0.521
Elem. Laranja C4A	$g Y index Y$	99.40 %	5.54 %	0.482
Elem. Laranja C4B	$r b Val$	99.01 %	3.17 %	0.482

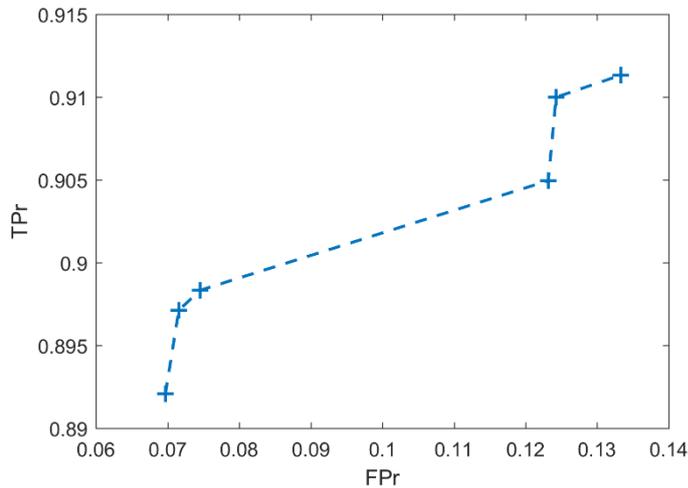


Fig. 7.21. Curva ROC das configurações pré-selecionadas dos classificadores SVM independentes nos dados de teste.

Para avaliar o impacto das operações morfológicas, aplicadas à máscara resultante do método de segmentação dos limões, foram selecionadas duas combinações, de acordo com os resultados ilustrados na Fig. 7.21, e criadas as respectivas variações sem o classificador dos elementos cinza descritas na Tabela 7.5.

Para uniformizar a avaliação do impacto das operações morfológicas, posteriormente à metodologia de segmentação proposta nesta secção, também é aplicado um elemento estruturante, com forma de losango, com um raio a variar entre 3 e 8 (*pixels*) para a operação de erosão, aplicando-se, seguidamente, a operação de dilatação, com um raio entre 3 e 9 (*pixels*).

Tabela 7.5. Configurações dos classificadores SVM independentes usadas na avaliação do impacto das operações morfológicas.

TPr	FPr	Configuração dos Classificadores Independentes				Cód.
		Folhagem	Ramos	Elem. Cinza	Elem. Laranja	
89.83 %	7.46 %	C1A	C2B	C3A	C4B	Config Ind A1
-	-	C1A	C2B	-	C4B	Config Ind A2
91.00 %	12.43 %	C1A	C2A	C3A	C4A	Config Ind B1
-	-	C1A	C2A	-	C4A	Config Ind B2

Analisando a curva ROC resultante das operações morfológicas, Fig. 7.22, conclui-se que é atingido um melhor desempenho ao recorrer quer à “Config Ind A1” quer à “Config Ind A2”. Além disso, entre estas duas configurações, o classificador dos elementos cinza não produz nenhuma alteração no rácio  $TPr/FPr$ ; apenas irá aumentar a latência do sistema ao fazer com que os *pixels* sejam analisados por um classificador adicional.

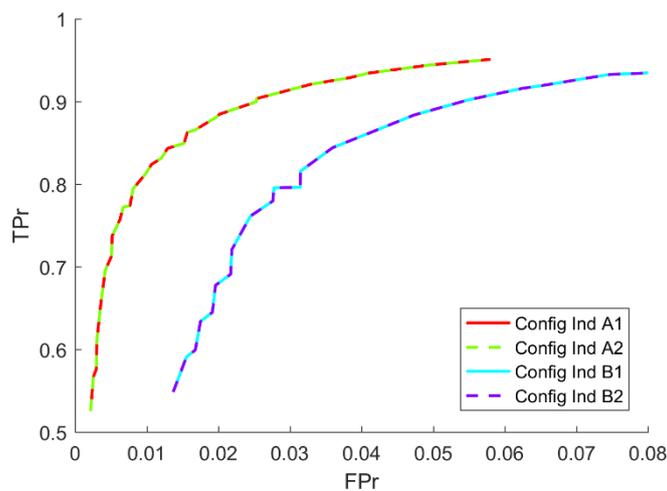


Fig. 7.22. Curva ROC das operações morfológicas avaliadas após os classificadores SVM independentes nas imagens de teste.

Para analisar e discutir os reais efeitos da utilização do classificador dos elementos cinza, seria necessário colocar este classificador no último nível de decisão e avaliar o rácio  $TPr/FPr$  em cada nível da árvore de decisão. Porém, acontece que o rácio  $TPr/FPr$ , para o classificador SVM global, é mais satisfatório do que o atingido com a metodologia dos classificadores independentes, onde é atingida uma taxa de TP de 93.44 % para uma FPr correspondente de 4.10 %. Desta forma, não é encontrada motivação suficiente, no presente contexto, que justifique explorar este acontecimento.

Contudo, para que seja possível comparar os resultados alcançados pelas diferentes abordagens, para uma taxa de TP aproximada, são apresentados na Fig. 7.23 os diagramas de extremos e os quartis para a configuração detalhada na Tabela 7.6.

Como o classificador dos elementos cinza não produz efeitos nos resultados finais da árvore de decisão, opta-se por mostrar os resultados da “Config Ind A2”, ou seja, sem aplicar um classificador SVM para a classe dos elementos cinza. A configuração selecionada permite alcançar uma taxa TP de 93.44 % e FPr de 4.10 %, quando aplicada a operação de erosão com um raio de três *pixels* seguida da dilatação com um raio de sete *pixels*.

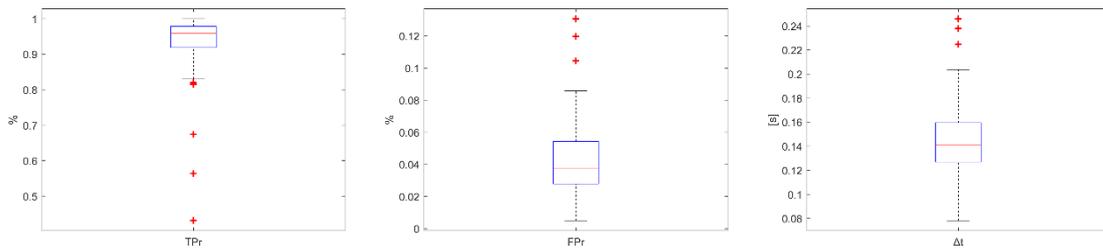


Fig. 7.23. Diagrama de extremos e quartis relativo à aplicação dos classificadores em independentes nos dados de teste: (esquerda) TPr; (centro) FPr; (direita)  $\Delta t$ .

Tabela 7.6. Configurações selecionada para os classificadores SVM independentes e resultados de teste.

	Config Ind A2		
	Folhagem	Ramos	Elementos Laranja
$\sigma$	10.000	1.000	4.266
Características	<i>g Y Hue Sat</i>	<i>b Y</i>	<i>r b Val</i>
<i>C</i>	4	1000	2.512
<i>c</i>	0.200	1.300	0.500
<i>outliers</i>	3.5 %	3.5 %	0 %
<i>TPr</i>	93.44 %		
<i>FPr</i>	4.10 %		
<i>Acc</i>	95.86 %		
$\Delta t$ [s]	0.145		

## 7.3 Classificadores SVM em cadeia

Não obstante os resultados obtidos pelos classificadores SVM independentes, nesta secção apresenta-se a avaliação do desempenho global alcançado por classificadores SVM, onde cada classificador, em cada nível, é treinado apenas com as instâncias não classificadas como *background* nos níveis anteriores.

Ao treinar os classificadores com menos instâncias, principalmente da classe negativa, haverá predominância das instâncias da classe positiva; assim, prevê-se que o parâmetro  $c$  do SVM tenha maior influência na resposta do que nos casos anteriormente descritos.

Por ser um processo de treino em cadeia, importa analisar o desempenho dos classificadores pelos dados de validação, considerando o número total de instâncias de validação de cada classe, conforme apresentado na Tabela 6.1.

Deve-se ter em conta que o desempenho real de cada classificador seja diferente dos apresentados devido a parte das instâncias iniciais serem “filtradas” nos níveis anteriores da árvore de decisão. O desempenho individual destes classificadores pode ser consultado no Anexo C, onde apenas são consideradas as instâncias que não foram retidas nos níveis anteriores da árvore de decisão. A única variável que não é alterada é a referente ao tempo médio de processamento por imagem.

### 7.3.1 Limões vs folhagem verde

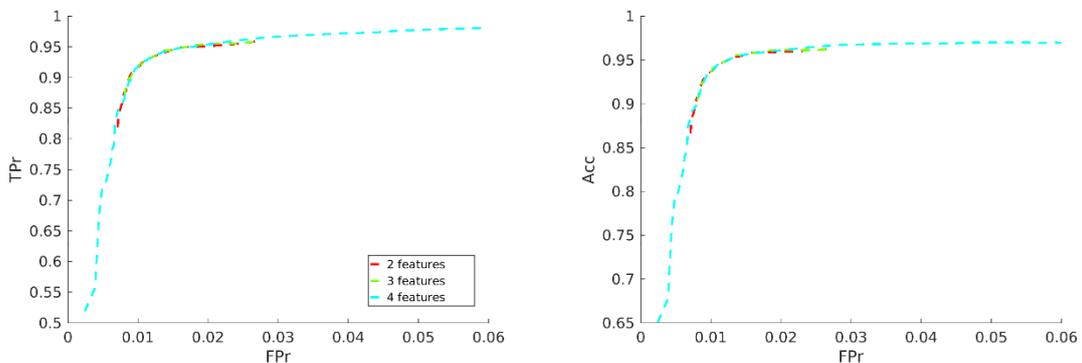


Fig. 7.24. Desempenho de classificadores SVM em cadeia para folhagem até quatro características e  $\Delta t_{rel} \leq 500$  ms: (esquerda) curva ROC; (direita) Acc correspondente.

Na Fig. 7.24 é apresentado o desempenho dos classificadores para a separação dos limões da folhagem. Neste classificador são apenas processadas as instâncias que não foram classificadas como *background* nos dois níveis anteriores, de acordo com a Fig. 7.9. De acordo com a Tabela 7.1, a  $TPR$  máxima possível de alcançar seria de 99.48 %, uma vez que as restantes instâncias são

previamente classificadas como *background*. Do total de instâncias de treino da classe da folhagem, 43.43 % são eliminadas nos níveis anteriores.

A Fig. 7.25 apresenta com maior detalhe a curva ROC e o tempo médio de processamento por imagem, onde se verifica que, para taxas mais elevadas de FP, a combinação de quatro características permite obter melhores resultados.

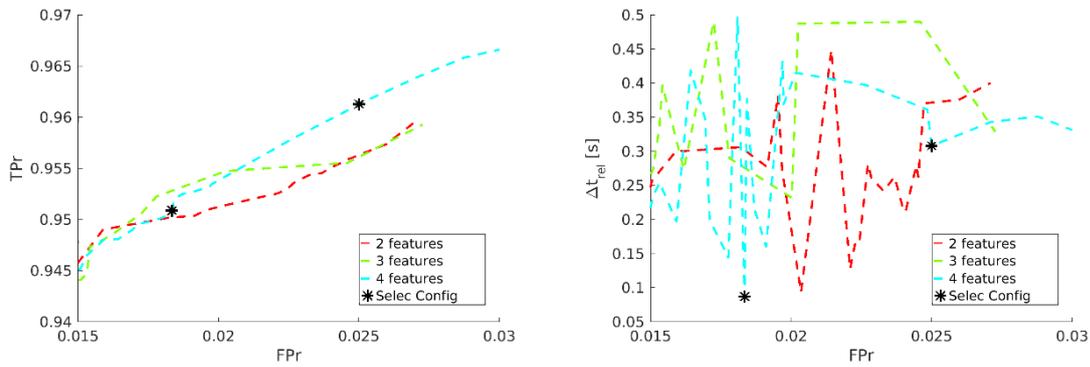


Fig. 7.25. Detalhe do desempenho de classificadores SVM em cadeia para folhagem até quatro características: (esquerda) curva ROC; (direita)  $\Delta t_{rel}$  correspondente.

Uma vez que é necessário definir a configuração a usar antes de se avançar para o treino do classificador do nível seguinte, optou-se por avaliar duas configurações diferentes assinaladas na Fig. 7.25, estando os resultados e as diversas configurações detalhados na Tabela 7.7. O ponto denominado “Config A” foi selecionado por apresentar um  $\Delta t_{rel}$  baixo relativamente aos restantes, e a “Config B” por apresentar uma melhor TPr sem ultrapassar a barreira dos 3 % de FPr.

Tabela 7.7. Configurações selecionadas de limões vs folhagem.

	Config A	Config B
$\sigma$	6.7608	35.4813
Características	<i>g Sat indexY indexA</i>	<i>g Hue Sat indexY</i>
<i>C</i>	1.0233	1.2882
<i>c</i>	0.3000	1
<i>outliers</i>	0%	3.50%
<i>TPr</i>	95.09%	96.13%
<i>FPr</i>	1.83%	2.50%
<i>Acc</i>	95.93%	96.51%
<i>F1-score</i>	97.14%	97.55%
$\Delta t_{rel}$ [s]	0.087	0.308

### 7.3.2 Limões vs ramos

Os resultados desta classe são influenciados pelo número de instâncias que não são “filtradas” nos níveis anteriores. A Tabela 7.8 apresenta a supressão de instâncias da classe dos limões e ramos resultante dos níveis de decisão anteriores.

Tabela 7.8. Supressão de instâncias da classe ramos.

	Supressão de instâncias			
	Classe limões		Classe ramos	
	Treino	Validação	Treino	Validação
SVM folhagem: Config A	0.00 %	4.91 %	75.40 %	74.75 %
SVM folhagem: Config B	0.00 %	3.87 %	57.22 %	59.61 %

Aplicando a “Config A” no nó de decisão anterior obtêm-se os resultados representados na Fig. 7.26. Verifica-se ainda que a adição de uma terceira característica na combinação a ser analisada pelo classificador não melhora os resultados alcançados, à semelhança do que acontecia no treino dos classificadores por um processo independente (seção 7.2.2).

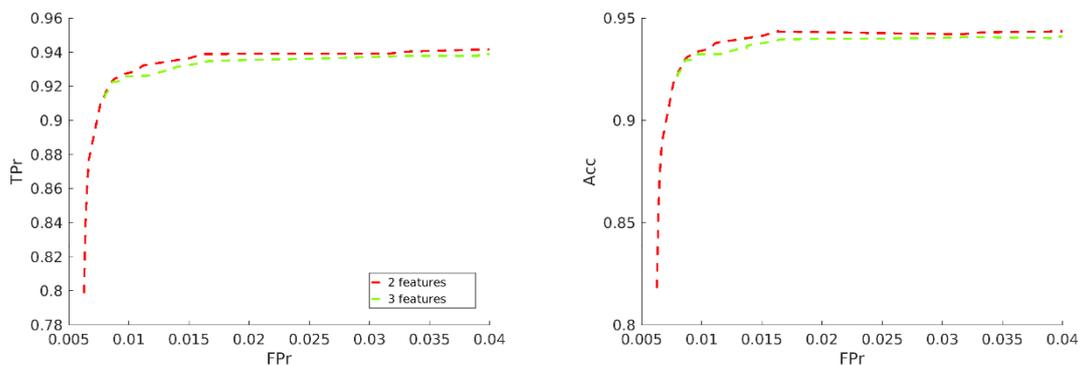


Fig. 7.26. Desempenho de classificadores SVM em cadeia para ramos até três características, para a Config A e  $\Delta t_{rel} \leq 500$  ms: (esquerda) curva ROC; (direita) Acc correspondente.

Usando a “Config B”, não se verificam discrepâncias tão significativas no desempenho, quando comparadas com as combinações de duas ou três características, como na “Config A”, conforme é representado na Fig. 7.27. A Fig. 7.28 mostra com maior detalhe a curva ROC e o  $\Delta t_{rel}$  desta configuração.

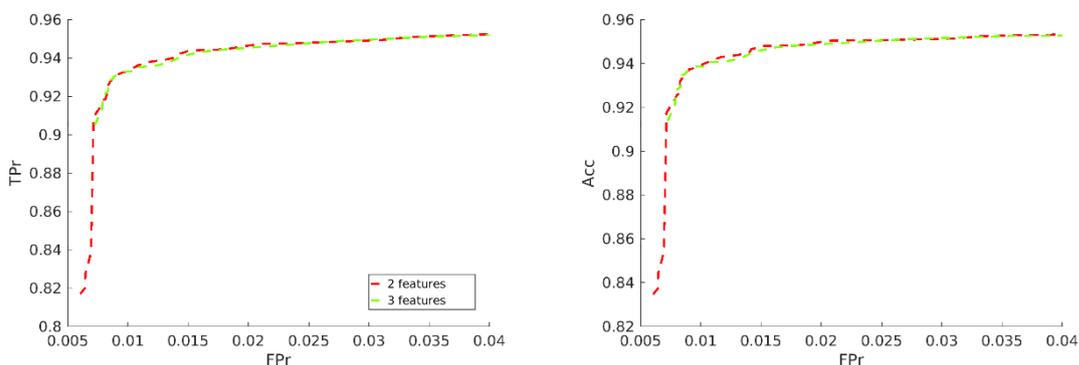


Fig. 7.27. Desempenho de classificadores SVM em cadeia para ramos até três características, para a Config B e  $\Delta t_{rel} \leq 500$  ms: (esquerda) curva ROC; (direita) Acc correspondente.

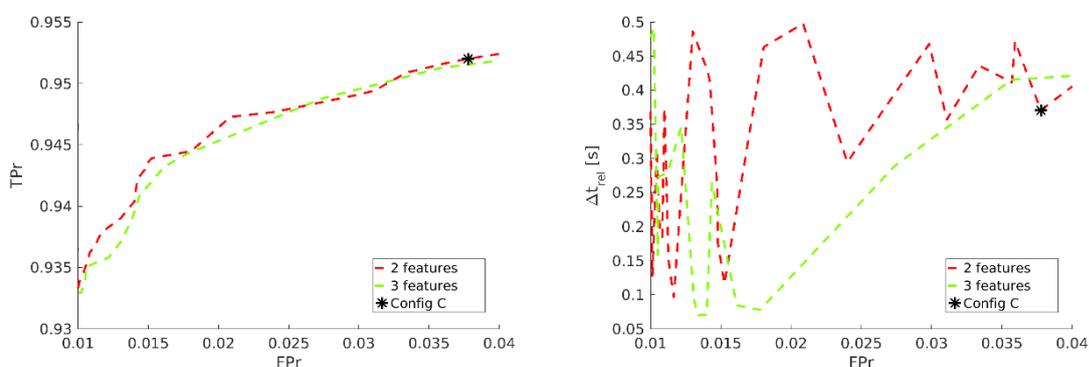


Fig. 7.28. Detalhe do desempenho de classificadores SVM em cadeia para ramos até três características, para a Config B: (esquerda) curva ROC; (direita)  $\Delta t_{rel}$  correspondente.

Para facilitar a comparação do desempenho dos classificadores, quando é usada a “Config A” ou a “Config B” - à semelhança da operação realizada para traçar o melhor desempenho para combinações com um determinado número de características - são apresentadas na figura 7.21 apenas duas curvas, uma correspondente à aplicação da “Config A” e a outra à “Config B”.

Conforme ilustra a Fig. 7.29, e com maior detalhe na Fig. 7.30, há um melhor desempenho com a “Config B” para a segmentação da folhagem do que com a “Config A”. Em ambas as configurações, o  $\Delta t_{rel}$  cumpre com o limite máximo imposto de 500 ms, pelo que é descartada a hipótese de usar a “Config A”.

Como há maior supressão de instâncias de treino para a classe dos limões e dos ramos, ao usar a “Config A”: são prematuramente eliminadas instâncias que podem ser relevantes para o treino destes classificadores; o processo inicia-se com uma menor  $TP_r$ . Assim se justifica a discrepância de resultados apresentada na Fig. 7.29, aquando da comparação das duas configurações.

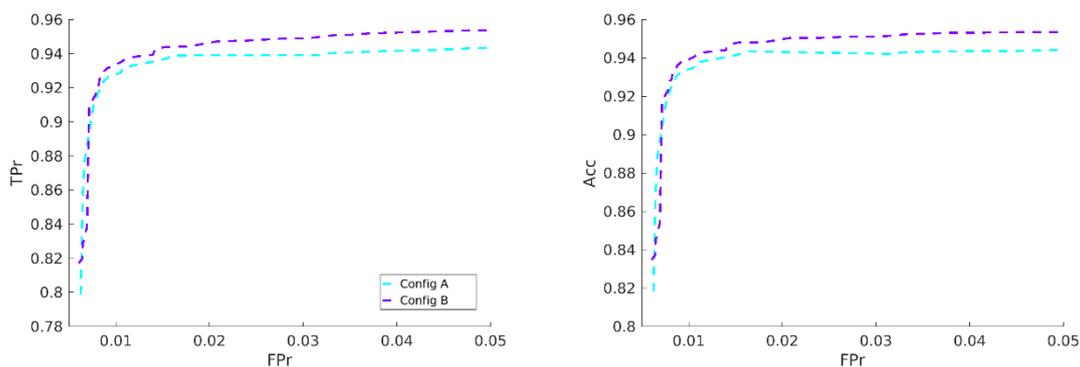


Fig. 7.29. Desempenho de classificadores SVM em cadeia para ramos e  $\Delta t_{rel} \leq 500$  ms em função da configuração do nível anterior: (esquerda) curva ROC; (direita) Acc correspondente.

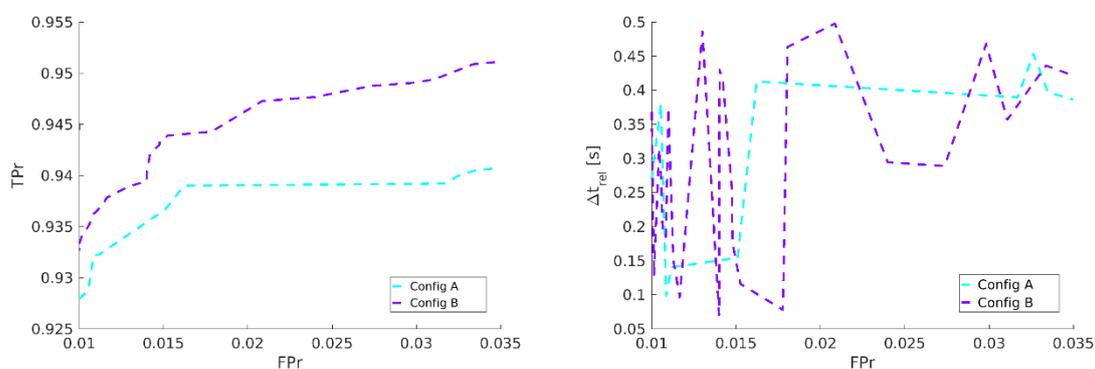


Fig. 7.30. Detalhe do desempenho de classificadores SVM em cadeia para ramos com  $\Delta t_{rel} \leq 500$  ms em função da configuração do nível anterior: (esquerda) curva ROC; (direita)  $\Delta t_{rel}$  correspondente.

A configuração do classificador SVM para esta classe, denominada “Config C”, é apresentada na Tabela 7.9. Para este classificador procurou-se estabelecer um compromisso  $TPr/FPr$  de modo a não prejudicar em demasia a  $TPr$  neste e nos níveis seguintes da árvore de decisão, e por ser uma classe também significativa, mantendo a  $FPr$  inferior a 4 %.

Tabela 7.9. Configuração selecionada de limões vs ramos.

	Config C
$\sigma$	0.1820
Características	$b Y$
$C$	0.6310
$c$	1
<i>outliers</i>	0 %
<i>TPr</i>	95.20 %
<i>FPr</i>	3.78 %
<i>Acc</i>	95.30 %
<i>F1-score</i>	97.33 %
$\Delta t_{rel}$ [s]	0.371

### 7.3.3 Limões vs elementos cinza e laranja

Seguindo a estrutura apresentada na Fig. 7.9, segue-se o treino dos classificadores para duas (sub)classes particulares do banco de imagens existente: os elementos cinza; e laranja. Porém, após analisar os efeitos dos níveis de decisão anteriores nas instâncias da classe dos elementos cinza (Tabela 7.10), conclui-se que há uma supressão significativa (> 65 %) das instâncias de treino.

Extrapolando a experiência obtida no treino do classificador da secção anterior (secção 7.3.2), onde a “Config A” originava também uma supressão significativa das instâncias de treino da classe dos ramos, o desequilíbrio de instâncias de treino das duas classes, limões e elementos cinza, pode originar classificadores que promovem a degradação mais acentuada da taxa de TP, como aconteceu para os ramos, tal como ilustrado em detalhe na Fig. 7.30.

Tabela 7.10. Supressão de instâncias da classe elementos cinza.

	Supressão de instâncias			
	Classe limões		Classe elementos cinza	
	Treino	Validação	Treino	Validação
Config B e Config C	0 %	4.80 %	65.69 %	74.10 %

Perante este cenário poder-se-ia trocar a ordem dos classificadores SVM apresentados na Fig. 7.9, de modo a avaliar, primeiro a presença de elementos laranja e, só depois de elementos cinza. Neste caso, há uma menor supressão de instâncias de elementos laranja do que dos cinza,

conforme é representado na Tabela 7.11. Contudo, não era possível recuperar as instâncias de treino da classe cinza, sendo previsível um aumento da supressão de instâncias relativamente às apresentadas na Tabela 7.10.

Tabela 7.11. Supressão de instâncias da classe elementos laranja.

	Supressão de instâncias			
	Classe limões		Classe elementos cinza	
	Treino	Validação	Treino	Validação
Config B e Config C	0 %	4.80 %	22.99 %	25.55 %

Nestas circunstâncias é proposto agrupar as classes dos elementos cinza e laranja numa nova classe, por forma a promover o equilíbrio das instâncias de treino da classe positiva e negativa, e assim procurar um classificador único que sirva as duas classes sem prejuízo da taxa de TP. A Tabela 7.12 apresenta os dados relativos à supressão de instâncias da nova classe, resultante da união das duas classes, admitindo que a “Config B” é aplicada para a segmentação da folhagem e a “Config C” para os ramos.

Tabela 7.12. Supressão de instâncias da classe elementos cinza e laranja.

	Supressão de instâncias			
	Classe limões		Classe elementos cinza e laranja	
	Treino	Validação	Treino	Validação
Config B e Config C	0 %	4.80 %	46.03 %	63.52 %

Com esta alteração, a estrutura da árvore de decisão inicialmente projetada, e esquematizada na Fig. 7.9, passa a assumir o esquema apresentado na figura seguinte.

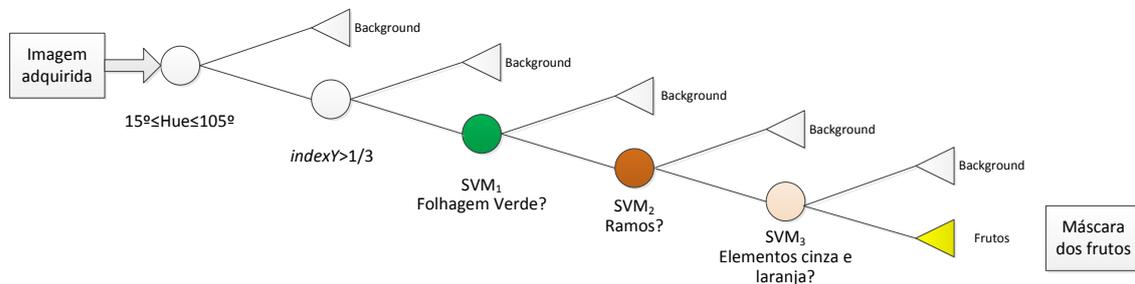


Fig. 7.31. Estrutura da árvore de decisão usada na estratégia “um contra um” em cadeia.

Com a reestruturação proposta para a árvore de decisão, consegue-se atingir taxas de TP a rondar os 94 %; as variações ocorrem em função da *FPr* selecionada. Os valores da *TPr* apresentados na Fig. 7.32 representam os valores finais, relativos aos dados de validação, pois não existem nós de decisão posteriores.

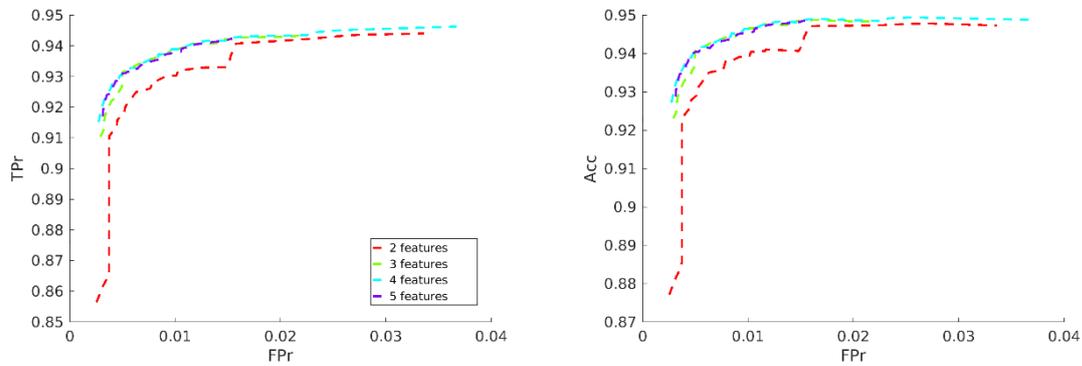


Fig. 7.32. Desempenho de classificadores SVM em cadeia para elementos cinza e laranja, até cinco características, para a Config B, Config C e  $\Delta t_{rel} \leq 500$  ms: (esquerda) curva ROC; (direita) Acc correspondente.

Nestas condições são as combinações de quatro características que permitem alcançar o melhor rácio *TPr/FPr* generalizado. À semelhança do sucedido no treino dos classificadores independentes para a classe dos elementos cinza (capítulo 7.2.3), também aqui a introdução da quinta característica provoca uma quebra no desempenho dos classificadores; por este motivo, e tal como nos casos anteriores, não foram avaliadas combinações com um número superior de características.

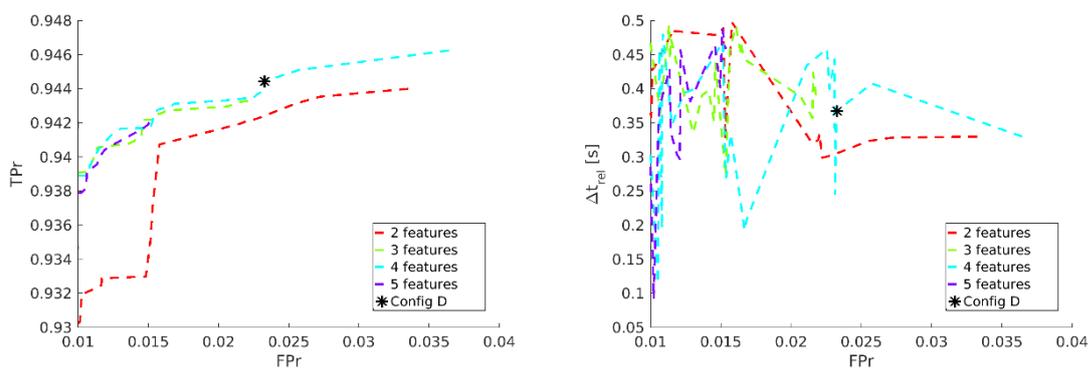


Fig. 7.33. Detalhe do desempenho de classificadores SVM em cadeia para elementos cinza e laranja até cinco características, para a Config B e Config C: (esquerda) curva ROC; (direita)  $\Delta t_{rel}$  correspondente.

A configuração escolhida para esta classe, “Config D”, permite atingir uma taxa de *TPr* de 94.44 % nos dados de validação, conforme representado na Fig. 7.33 e na Tabela 7.13. A confirmar-se o bom desempenho dos classificadores nos dados de teste, os resultados alcançados com a abordagem proposta são equivalentes aos melhores resultados anunciados no estado da arte com a vantagem de se usar um sistema de visão artificial de menor complexidade do que a maioria das experiências realizadas por outros autores.

Tabela 7.13. Configuração selecionada de limões vs elementos cinza e laranja.

	Config D
$\sigma$	2.4434
Características	<i>g b Cb Val</i>
<i>C</i>	6.6374
<i>c</i>	0.5000
<i>outliers</i>	0%
<i>TPr</i>	94.44 %
<i>FPr</i>	2.33 %
<i>Acc</i>	94.92 %
$\Delta t_{rel}$ [s]	0.367

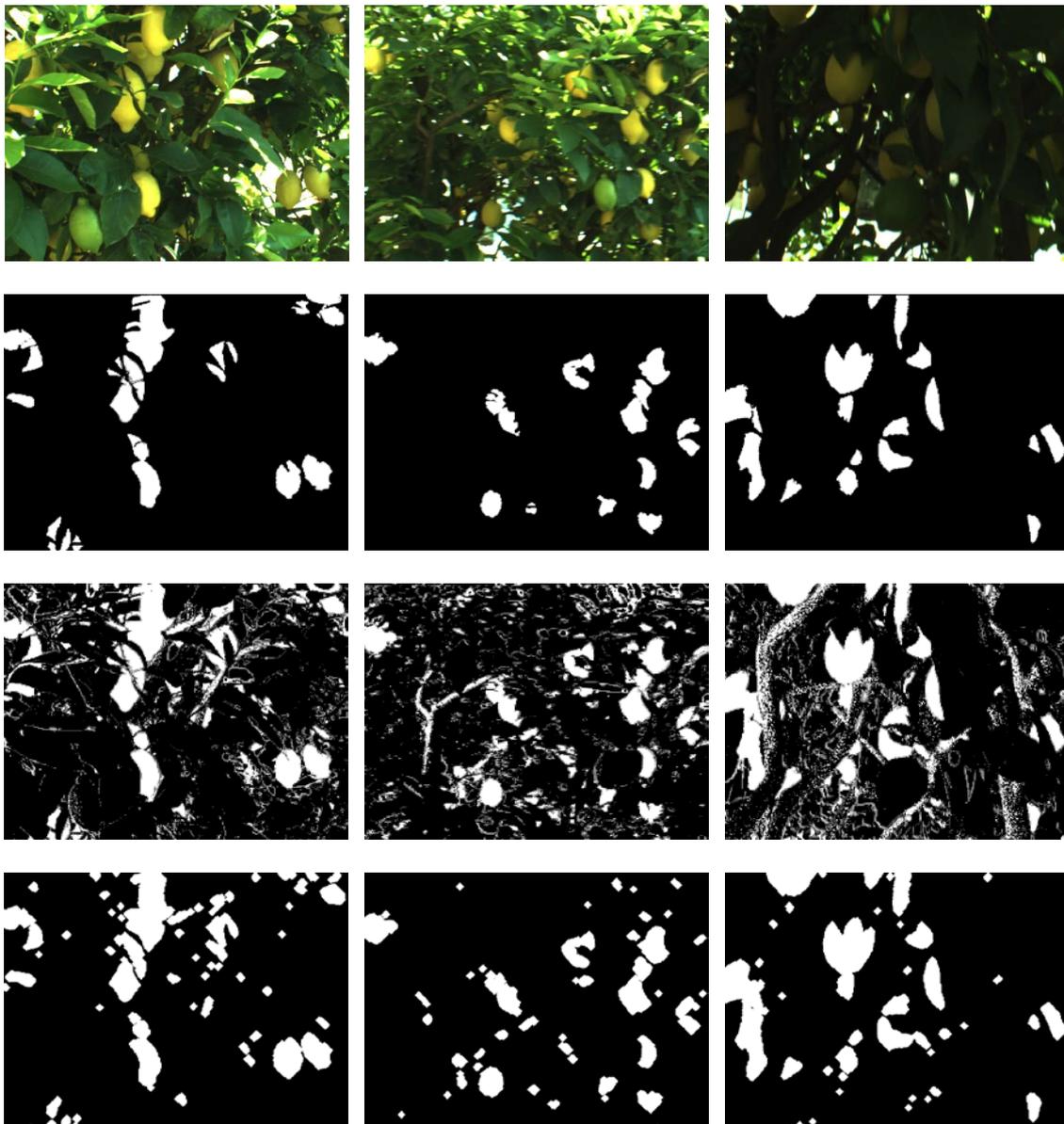
### 7.3.4 Resultados de teste

Aplicando a classificação proposta pela árvore decisão (Fig. 7.31) nas 133 imagens de teste, com as configurações dos classificadores SVM sugeridas nas secções anteriores, são obtidos, em média, os seguintes valores:

- $\overline{TPr} = 95.55 \%$
- $\overline{FPr} = 16.42 \%$
- $\overline{\Delta t} = 361.6 \text{ ms}$

Apesar da taxa de TP ser boa, há uma eleva taxa de FP. Contudo, pela análise da máscara resultante desta metodologia de classificação apura-se a existência de uma quantidade significativa de ruído, Fig. 7.34.

Como os resultados provenientes do método de classificação da árvore de decisão contêm ruído, é proposto fazer uma filtragem através de operações morfológicas (erosão e dilatação). Como há uma amplitude elevada relativamente à distância expetável da câmara aos limões é realizada uma avaliação, nas imagens de teste, com o objetivo de determinar as dimensões adequadas para as máscaras do processo de erosão e dilatação.



*Fig. 7.34. Exemplos da classificação resultante da aplicação do método de classificadores SVM em cadeia: (primeira linha) imagem RGB; (segunda linha) máscara de referência relativa aos limões; (terceira linha) máscara resultante da árvore de decisão; (quarta linha) máscara resultante das operações morfológicas.*

Usando um elemento estruturante com forma de losango, cujo raio foi feito variar entre 3 e 8 (*pixels*) para a erosão e entre 3 e 9 (*pixels*) para a dilatação, obtiveram-se os resultados médios finais apresentados na Fig. 7.35.

A seleção da configuração ótima estará dependente dos requisitos impostos para cada aplicação, ou do desempenho atingido pela metodologia de detecção a ser aplicada futuramente.

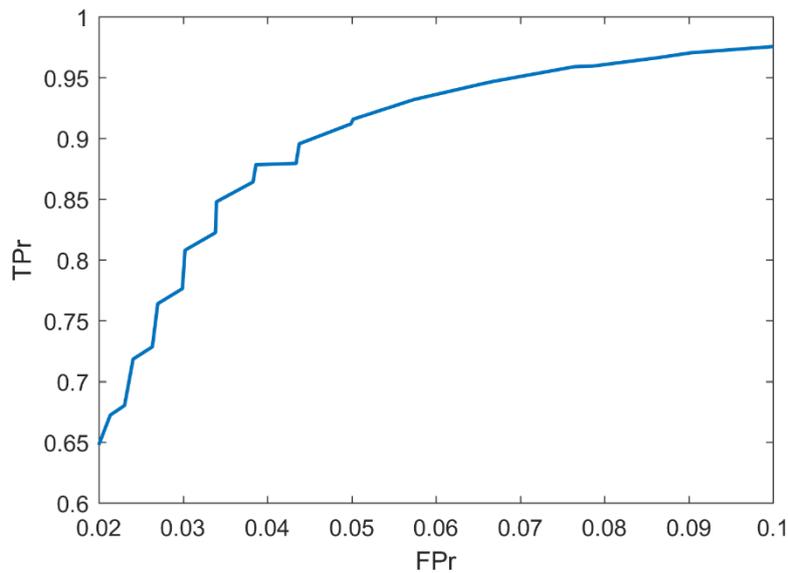


Fig. 7.35. Curva ROC das operações morfológicas avaliadas após os classificadores SVM em cadeia nas imagens de teste.

Admitindo que é selecionada a configuração que permite obter uma taxa de TP média global de 93.18 %, corresponde a uma  $FPr = 5.74 \%$  obtida com um raio de 6 na erosão e um raio de 8 na dilatação. São ainda apresentados, na Fig. 7.36, os diagramas de extremos e quartis obtidos com esta parametrização nos dados de teste.

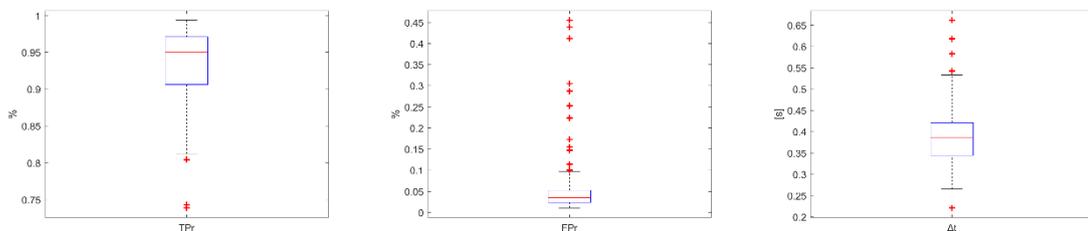


Fig. 7.36. Diagrama de extremos e quartis relativo à aplicação do método de classificação em cadeia nos dados de teste: (esquerda)  $TPr$ ; (centro)  $FPr$ ; (direita)  $\Delta t$ .

## 7.4 Resultados da detecção dos frutos

Os resultados dos ensaios apresentados em [28], referentes à utilização do classificador de *Haar* treinado em cascata, sugerem para esse caso que o melhor compromisso entre frutos bem detetados (TD) e falsos-alarmes (FA) é obtido usando uma cascata com 18 estágios e com  $FA_{estágio} = 0.5$ .

Fazendo uma breve análise aos efeitos das parametrizações da cascata, aquando do treino dos classificadores, importa reter que, geralmente:

- Quanto maior for o número de estágios, menor será a ocorrência de FA a nível global, com o risco de se prejudicar a TD;
- Um valor  $FA_{estágio}$  de referência para o treino mais elevado significa que será permitida a ocorrência de mais FA em cada estágio, e poderão ser necessários mais estágios para se atingir a meta global de FA.

Os resultados apresentados nesta secção relativos aos três métodos avaliados (*Haar*, HOG e LBP) são obtidos para um valor de referência de  $FA_{estágio} = 0.8$ , fazendo variar entre 15 e 35 o número de estágios. Aplicando a equação (5.1) às condições de treino dos classificadores é, teoricamente, previsto que a  $FA_{global}$  seja, no máximo, de 3.5 %.

Relativamente à imagem monocromática a analisar, opta-se por indicar explicitamente cada componente individual dos modelos de cor HSV, YCbCr e as cromaticidades  $r$ ,  $g$  e  $b$ , com o mesmo propósito anteriormente usado nos classificadores SVM, para atenuar os efeitos das variações de luz na imagem.

Tal como explicitado na secção 6.4, as métricas usadas para avaliar o desempenho dos modelos de classificação são distintas das usadas na segmentação. Em condições ótimas, a melhor configuração apresentaria o valor máximo nas duas métricas a usar, *Precision* e *Recall* - nos gráficos das Figuras 7.37 a 7.39 são apresentadas as configurações, posicionadas no canto superior direito, em oposição às curvas ROC, apresentadas para os resultados da segmentação, em que a configuração ótima estaria representada no canto superior esquerdo do gráfico.

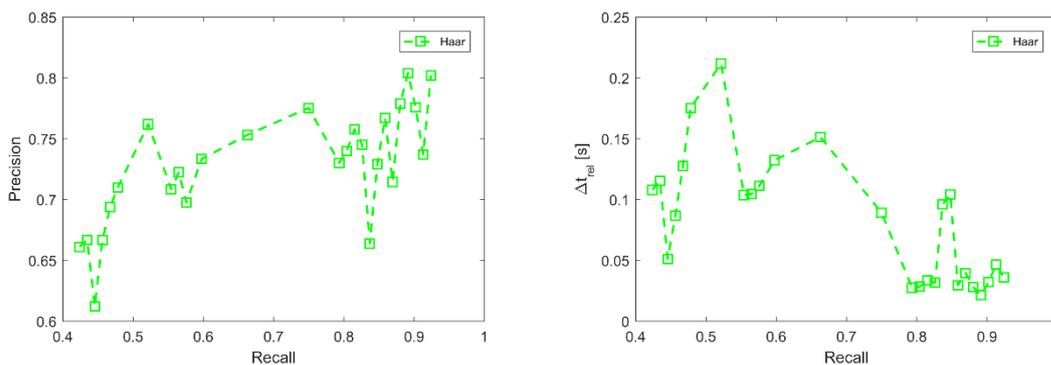


Fig. 7.37. Deteção pelo método de Haar: (esquerda) *Precision* vs *Recall*; (direita)  $\Delta t_{rel}$  correspondente.

As Fig. 7.37, Fig. 7.38 e Fig. 7.39 apresentam os resultados dos diferentes modelos de classificação, obtidos com as diferentes metodologias, aplicadas sobre os dados de validação. De modo a facilitar a interpretação dos gráficos, para cada valor da métrica *Recall* é apenas representado

graficamente aquele que apresenta melhor *Precision*, sendo também apresentado o tempo médio de processamento por imagem (apenas do processo de detecção).

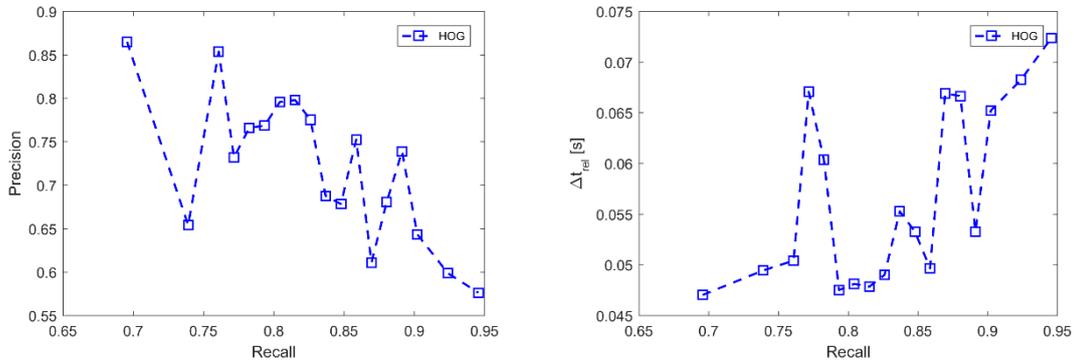


Fig. 7.38. Detecção pelo método de HOG: (esquerda) *Precision vs Recall*; (direita)  $\Delta t_{rel}$  correspondente.

A comparação direta dos três métodos numa só figura é feita no capítulo 8, assim como a análise dos resultados e mais detalhes sobre configurações pré-selecionadas.

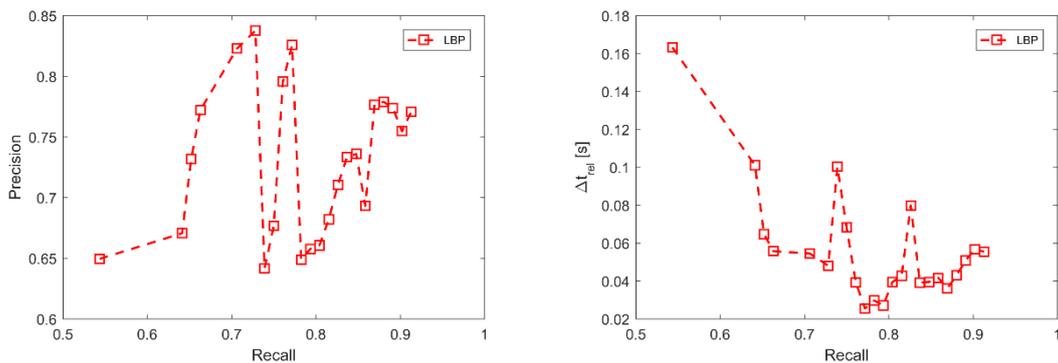


Fig. 7.39. Detecção pelo método de LBP: (esquerda) *Precision vs Recall*; (direita)  $\Delta t_{rel}$  correspondente.

A Fig. 7.40 apresenta os tipos de falhas mais comuns do método de detecção, recorrendo ao classificador de *Haar* “Config dA”, apresentado na Tabela 8.1. Na primeira linha verifica-se a ocorrência de duas ROI estatisticamente tratadas como FA, pois, apesar de representarem na realidade limões, estes não foram assinalados na base de dados de referência como limões, por representarem uma área visível de pequenas dimensões. Na parte central inferior da imagem segmentada (esquerda) é ainda possível identificar outro fruto cuja ROI foi excluída pelo processo de filtragem descrito no capítulo 5.

A segunda linha da Fig. 7.40, representa o caso de uma única ROI a abranger dois limões visivelmente separados. Este é um exemplo de um caso negativo da junção de ROI com uma

determinada percentagem de sobreposição. Estes casos podem ser prevenidos ao ajustar os limiares referentes à percentagem de área sobreposta e à área ativa de cada ROI (capítulo 5). Além disso há a necessidade de estender os testes para avaliar o comportamento do método da deteção de frutos que estejam parcialmente sobrepostos e, se necessário for, introduzir métodos de *declustering*, como a transformada de *Watershed* [10, 81] ou *X-means* [38].

Na terceira linha da Fig. 7.40 ocorre um FA originado pela folhagem com tonalidade aparente semelhante à dos limões. E, por fim, na última linha da mesma figura, um *flare* na imagem também dá origem à ocorrência de um FA.

Além destes casos, verificou-se ainda um caso de falha adicional relativo à ocorrência de mais do que uma ROI para o mesmo fruto.

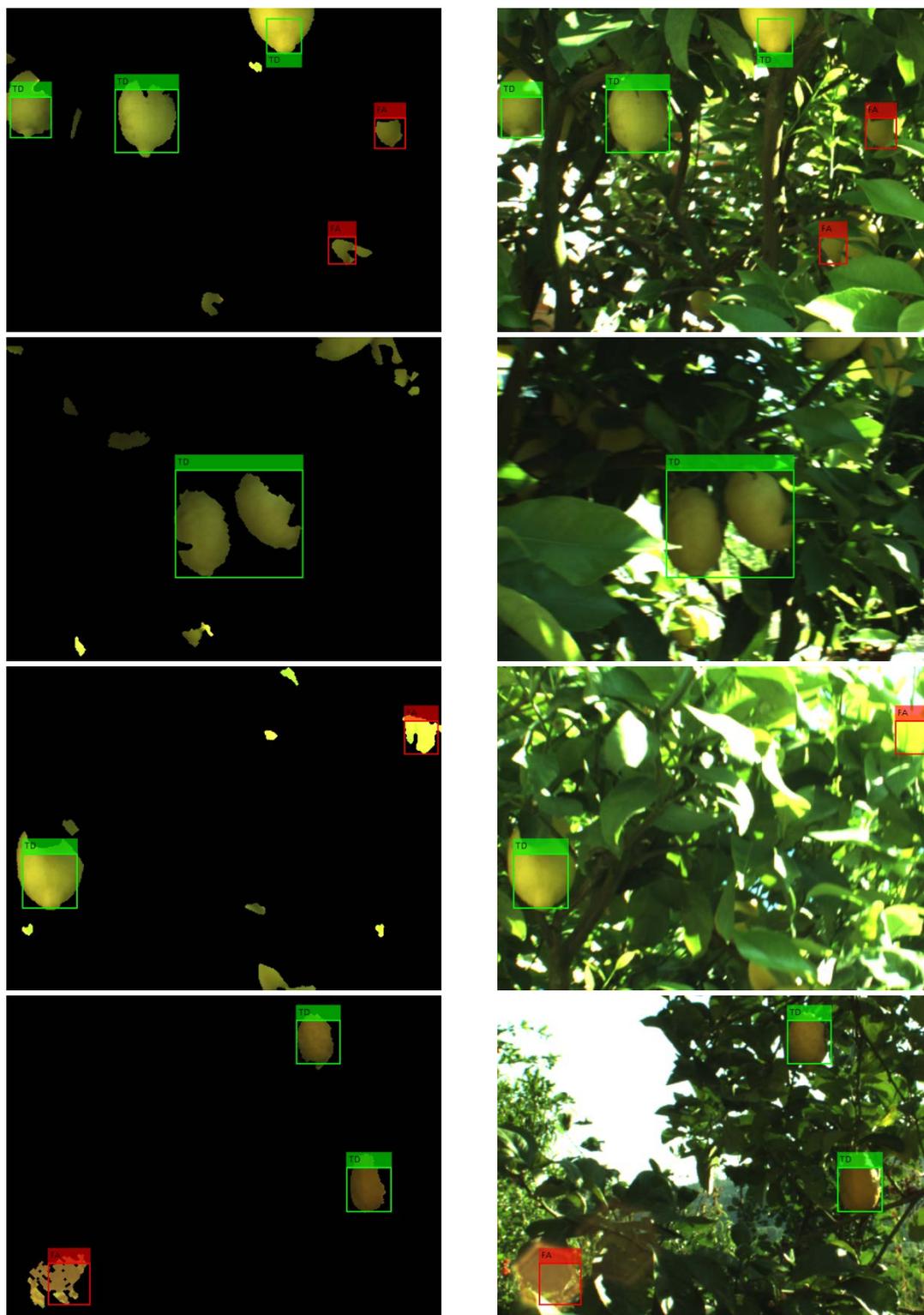


Fig. 7.40. Casos de falha na detecção com ROIs TD a verde e ROIs FA a vermelho: (esquerda) imagem segmentada; (direita) imagem RGB original.



## 8. ANÁLISE DE RESULTADOS

A segmentação dos limões do restante cenário representa uma etapa fundamental do sistema de detecção de frutos. Quanto maior for a  $TPr$  e menor for a  $FPr$  melhor o resultado da metodologia de classificação/segmentação aplicada. No entanto, nos resultados alcançados, tal como acontece tradicionalmente, a uma maior  $TPr$  corresponde também uma maior  $FPr$ , tal como ilustram as curvas ROC da Fig. 8.1.

No presente caso, como após a segmentação dos frutos é aplicada uma metodologia de detecção, a seleção do ponto ótimo da segmentação dependerá da influência das métricas  $TPr$  e  $FPr$  na detecção dos frutos.

Nas três metodologias de segmentação avaliadas, foram apresentados os respetivos resultados sobre os dados de teste nas respetivas secções desta monografia. Contudo, de modo a facilitar a comparação das metodologias, compilaram-se os resultados alcançados por cada metodologia após a aplicação das operações morfológicas, tal como apresentado nas figuras seguintes.

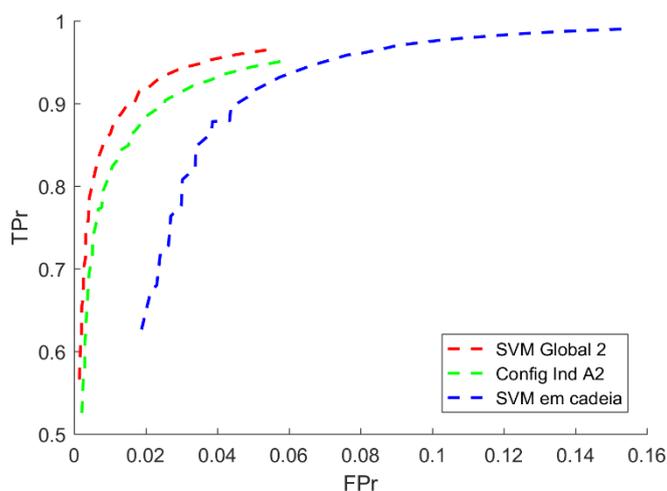


Fig. 8.1. Curvas ROC das metodologias de classificação dos limões nos dados de teste.

Os resultados representados na Fig. 8.1 demonstram que a motivação inicialmente existente em subdividir o problema da segmentação dos limões, em vários problemas mais simples, não produziu os resultados esperados. Há uma vantagem bastante clara, neste caso, em usar um classificador global.

Até a metodologia de treinar de forma independente classificadores SVM para separar os limões de uma (sub)classe específica do *background* supera os resultados alcançados pela metodologia de treino em cadeia.

Analisando com maior detalhe os resultados alcançados por cada metodologia nos dados de teste, Fig. 8.2, verifica-se que, em termos de tempo de classificação, a metodologia de treino dos classificadores SVM em cadeia apresenta os melhores resultados. Mais uma vez, uma das motivações da avaliação dos classificadores SVM em cadeia produziu os efeitos expectáveis.

A metodologia de classificador SVM global é, de todas as analisadas, a que apresenta melhores resultados de *FPr* e de tempo de processamento (para a classificação) por imagem e com menor número de valores discrepantes.

Quanto à taxa de TP, Fig. 8.2 esquerda, os resultados das três metodologias são mais equilibrados, com o método de classificador global a apresentar o maior número de valores discrepantes. Em termos práticos, podem ser criadas medidas que ignorem determinadas imagens e seja realizada a aquisição da imagem seguinte em condições diferentes, seja da posição relativa câmara-objeto, ou dos parâmetros relativos à exposição do sensor ótico à luz.

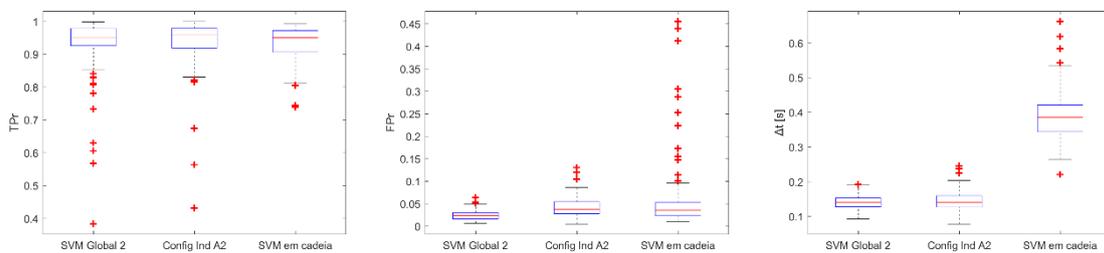


Fig. 8.2. Diagramas e extremos e quartis das três metodologias de classificação nos dados de teste: (esquerda) *TPr*; (centro) *FPr*; (direita)  $\Delta t$ .

Em termos médios, a configuração do classificador SVM global, nos dados de teste permite alcançar:

- $\overline{TPr} = 93.14 \%$
- $\overline{FPr} = 2.34 \%$
- $\overline{\Delta t} = 145 \text{ ms}$

Comparando com os resultados alcançados por outros autores, com diferentes metodologias e endereçados a frutos diferentes (Tabela 2.1), constata-se que a metodologia proposta alcança resultados similares ao estado da arte. Comparando diretamente com os resultados de segmentação relatados em [14], a metodologia proposta nesta monografia permite obter um rácio *TPr/FPr* melhor; relativamente às experiências apresentadas em [17] fica aquém dos resultados alcançados nas maçãs, mas supera os resultados obtidos para as uvas.

De salientar que a abordagem descrita em [17] recorre a um sistema de visão multiespectral e a um sensor de profundidade, permitindo obter uma visão tridimensional do cenário captado de forma direta. Os resultados reportados por esses autores são referentes a um modelo de classificação com classificadores SVM nos nós de uma árvore de decisão; interessou aos autores

identificar cada elemento do cenário na respetiva classe e não apenas distinguir os frutos de tudo o resto.

É precisamente em situações onde é requerido classificar cada elemento do cenário captado na respetiva classe que o classificador global poderá deixar de ser a metodologia mais adequada. No entanto, pode continuar a optar-se pela estratégia “um contra todos” no treino dos classificadores para a identificação dos elementos de cada classe, sendo criados vários classificadores “globais”.

Perante os resultados alcançados, e de acordo com as metodologias adotadas, podem ser melhoradas determinadas condições com o objetivo de aperfeiçoar o desempenho dos classificadores.

No treino dos classificadores SVM em cadeia deveriam ser criadas instâncias das restantes subclasses do *background*, conforme fossem sendo aplicados os modelos de decisão nos níveis anteriores da árvore. Isto faria com que não ocorresse a supressão de instâncias e, muito provavelmente, seriam determinados por uma classificação mais robusta a essas condições.

No entanto, como tinha sido criada uma base de dados, optou-se por realizar o treino de todos os classificadores usando as mesmas instâncias, de modo a ser possível comparar os resultados alcançados pelas diferentes abordagens apresentadas.

Relativamente ao classificador global pode-se recorrer ao método clássico de *k-fold* para a subdivisão dos dados em treino, validação e teste. Uma vez que os limões estão exaustivamente marcados em cada imagem, o seu complemento é fácil de obter e, assim se obtinha um valor superior de instâncias da classe negativa em relação à abordagem utilizada. Porém, as classes em minoria poderão perder peso no treino dos classificadores e dar origem a um pior desempenho global. São situações onde só se conhecem os reais efeitos de uma metodologia ou de outra depois de serem efetuadas todas as avaliações.

No presente caso, onde apenas se pretende separar os limões dos restantes elementos das imagens e onde não existem imagens captadas com filtros óticos, em determinados intervalos espectrais, para aumentar o contraste entre determinadas classes prova-se, pelos resultados alcançados, que a melhor metodologia a seguir é pelo uso de um classificador global, treinado segundo a estratégia “um contra todos”.

Além dos resultados apresentados pelo método de treino dos classificadores SVM em cadeia, há a desvantagem adicional de um processo de treino mais moroso do que as restantes abordagens, pois só se consegue treinar o classificador do último nível depois de estar terminado o treino e selecionada a configuração a usar nos níveis anteriores. Nas outras duas metodologias apresentadas pode-se iniciar o treino dos classificadores de qualquer classe em paralelo. O único procedimento sequencial é o do aumento do número de características analisadas, seguindo a metodologia proposta nesta dissertação.

Relativamente à deteção dos frutos, faz-se uso do processo de segmentação apresentado, suportado num classificador SVM global, responsável pela eliminação da maioria dos elementos

que não pertencem à classe dos limões e, depois, aplicam-se os classificadores baseados nas metodologias *Haar*, HOG ou LBP para realizar a detecção dos frutos.

Comparando o desempenho das três metodologias nas imagens de validação (Fig. 8.3), este não consegue identificar um método que seja globalmente melhor do que os restantes, como acontece em [34], onde o LBP supera os restantes métodos quando aplicados individualmente. Note-se que na Fig. 8.3 e na Fig. 8.4 são omitidas algumas das configurações representadas nas imagens da secção 7.4, de modo a facilitar a interpretação dos resultados e a comparação dos métodos.

O valor máximo de *Precision* é obtido para um classificador baseado no método de HOG. No entanto, para taxas de *Recall* mais elevadas, a vantagem está no uso de classificadores baseados no método de *Haar*.

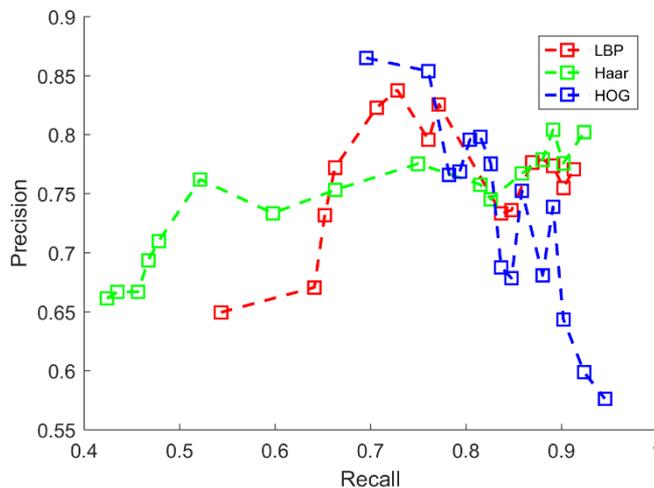


Fig. 8.3. Curvas *Precision vs Recall* de comparação dos métodos de *Haar*, *HOG* e *LBP* na detecção dos frutos.

A escolha da melhor configuração está, mais uma vez, indexada à importância atribuída à falha na detecção de frutos (MD) e à ocorrência de falsos-alarmes (FA). Se for dada prioridade absoluta à detecção de todos os frutos, com base nos resultados da Fig. 8.3, a solução ideal será usar um classificador de *Haar*; por outro lado, se o objetivo for minimizar a ocorrência de FA, deverá optar-se por um classificador baseado no método de HOG. Contudo, conforme o peso atribuído a cada um destes casos de falha, qualquer solução intermédia pode ser selecionada.

Contrariamente ao inicialmente esperado, pelos resultados enunciados em [34], os classificadores baseados na metodologia LBP não representam uma solução genericamente vantajosa face aos restantes.

As causas deste acontecimento podem estar relacionadas com a segmentação prévia da imagem com o método proposto nesta dissertação, evidenciando as transições de tons de cinza na detecção

dos contornos dos frutos através da metodologia HOG. Nos classificadores de *Haar*, como a maioria das regiões circundantes aos limões é eliminada, há uma maior diferença em aplicar a mesma característica de *Haar* na imagem segmentada do que na imagem original. Por outro lado, é mais difícil os classificadores distinguirem vários frutos parcialmente sobrepostos, sendo este um tópico a desenvolver futuramente.

Relativamente aos tempos de processamento, representados na Fig. 8.4, as configurações mais vantajosas são capazes de detetar os frutos existentes numa imagem em menos de 100 ms.

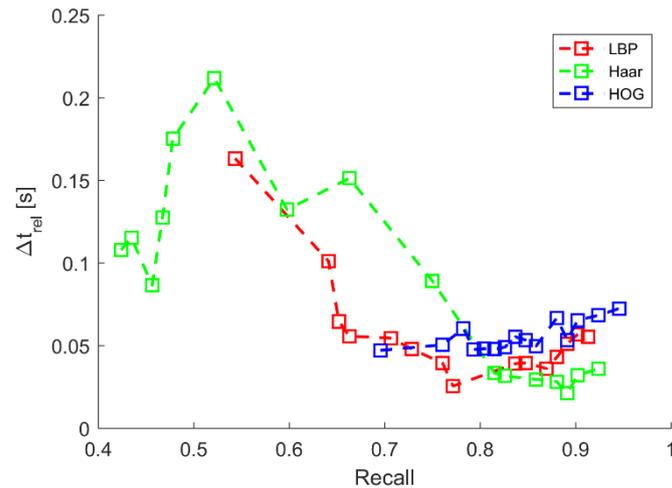


Fig. 8.4. Tempo médio de processamento por imagem dos métodos de Haar, HOG e LBP na deteção dos frutos.

A Tabela 8.1 caracteriza de forma mais detalhada quatro configurações pré-selecionadas. Assim, estão disponíveis os dados essenciais para se realizar qualquer análise comparativa com os resultados de outros autores. Nesta tabela também são disponibilizadas as parametrizações usadas no treino desses classificadores.

Dos resultados alcançados, a “Config dA” é aquela que permite detetar uma maior quantidade de frutos, com a penalização de maior ocorrência de FA relativamente a outras configurações. A menor ocorrência de FA é alcançada através da “Config dC”. Em termos práticos, a seleção da melhor combinação dependerá da aplicação ou das metodologias a aplicar posteriormente; é uma análise a realizar individualmente em função dos requisitos de cada aplicação.

Assumindo que o objetivo final é a integração desta metodologia num sistema robotizado de colheita de fruta, a opção pela configuração que reporta menos FA pode ser vantajosa, ao evitar que o sistema despenda recursos a analisar e, eventualmente, a tentar colher um fruto que não existe na realidade. Quanto aos frutos que não são inicialmente detetados, como é expetável que o sistema robotizado recolha várias imagens da árvore, em diferentes posições, esperando-se que os frutos inicialmente não detetados venham a ser detetados noutras imagens. Contudo, só com testes em ambiente real é possível tomar a decisão sobre qual a melhor configuração a aplicar.

Tabela 8.1. Detalhes das configurações pré-selecionadas dos modelos de detecção dos frutos.

	Config dA	Config dB	Config dC	Config dD
Método base	<i>Haar</i>	<i>Haar</i>	HOG	HOG
FA <sub>estágio</sub>	0.800	0.800	0.800	0.800
Nº estágios	32	35	26	25
Componente de cor	<i>Cb</i>	<i>Cb</i>	<i>b</i>	<i>b</i>
<i>Precision</i>	80.19 %	77.57 %	86.49 %	85.37 %
<i>Recall</i>	92.39 %	90.22 %	69.57 %	76.09 %
TD	85	83	64	70
MD	7	9	28	22
FA	21	24	10	12
$\Delta t_{rel}$ [s]	0.036	0.032	0.047	0.050

Através da análise comparativa dos resultados alcançados com a metodologia proposta nesta dissertação com outros autores, a mesma não é possível realizar em todos os trabalhos identificados no estado da arte, devido a serem usadas métricas de avaliação diferentes e a não serem publicados detalhes dos resultados alcançados, para que seja possível a determinação do seu desempenho segundo as métricas utilizadas nesta dissertação.

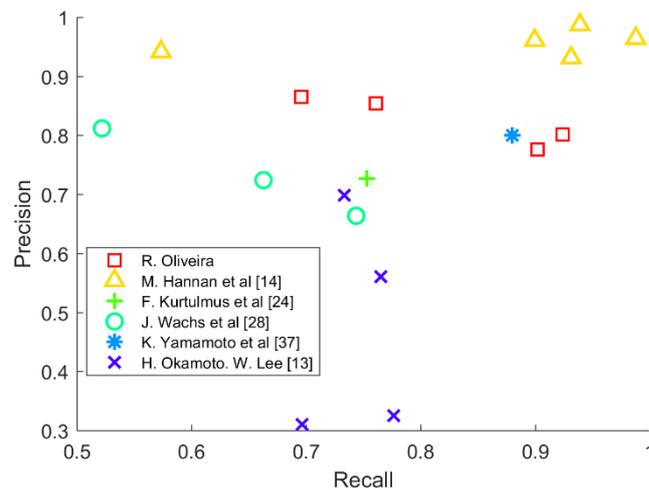


Fig. 8.5. Comparação da metodologia proposta com o estado da arte.

Contudo, nas referidas publicações, onde é possível determinar as métricas *Precision* e *Recall*, fazem-se representar graficamente os seus resultados na Fig. 8.5, juntamente com os alcançados nos limões pela metodologia proposta.

Deve salientar-se, novamente, que os resultados representados na Fig. 8.5 são obtidos com metodologias distintas e para diferentes tipos de frutos.

Os melhores resultados são os enunciados em [14] na detecção de laranjas, com uma larga vantagem sobre todos os restantes resultados apresentados. Aliás, esses resultados são equivalentes aos apresentados em [77] obtidos através de *Deep Learning*.

Relativamente às restantes abordagens, a metodologia proposta nesta dissertação supera as restantes abordagens. A proposta de Yamamoto *et al.*, [38], na detecção de tomates, é única, com resultados similares à metodologia proposta, relativamente às que se encontram representadas na Fig. 8.5.

Além da metodologia proposta apresentar vantagens nos resultados apresentados, os mesmos são obtidos com um sistema de visão artificial de menor complexidade, fazendo uso apenas de imagens a cores.

Em [35] é usada uma métrica para avaliar os FA em função do número de imagens avaliadas. Através dos dados publicados pelos autores não é possível representar os seus resultados segundo as métricas adotadas nesta dissertação. Desta forma, são representados na Tabela 8.2 os resultados obtidos com a metodologia proposta nesta dissertação, de acordo com as métricas definidas em [35].

Tabela 8.2. Comparação dos resultados alcançados com os anunciados em [35].

	DR [35]	aFPI [35]
C. McCool <i>et al.</i> [35]	0,76	1,20
C. McCool <i>et al.</i> [35]	0,69	2,10
R. Oliveira	0,92	0,48
R. Oliveira	0,90	0,55
R. Oliveira	0,70	0,23
R. Oliveira	0,76	0,27

Em [35] são avaliados classificadores baseados nas metodologias HOG, LBP e SAE para a detecção de pimentos, onde os autores concluem que há benefícios em combinar as respostas dos três classificadores. Conforme é apresentado na Tabela 8.2, a metodologia proposta nesta dissertação supera os resultados enunciados em [35].

A Fig. 8.7 apresenta alguns casos de sucesso na segmentação e posterior detecção dos frutos nas imagens. Note-se que, apesar da diferença aparente nas tonalidades apresentadas pelos frutos, a metodologia proposta consegue segmentar e detetar os frutos com sucesso.

Outros dois casos particulares na detecção de frutos é apresentado na Fig. 8.6. As imagens da primeira linha apresentam o caso da ocorrência um *flare*, mas sem prejuízo na detecção dos frutos, contrariamente ao caso apresentado na última linha da Fig. 7.40.

Na segunda imagem da Fig. 8.6 verifica-se a presença de um limão verde que é corretamente eliminado pelo método de segmentação, seguindo-se a correta detecção dos frutos na imagem.

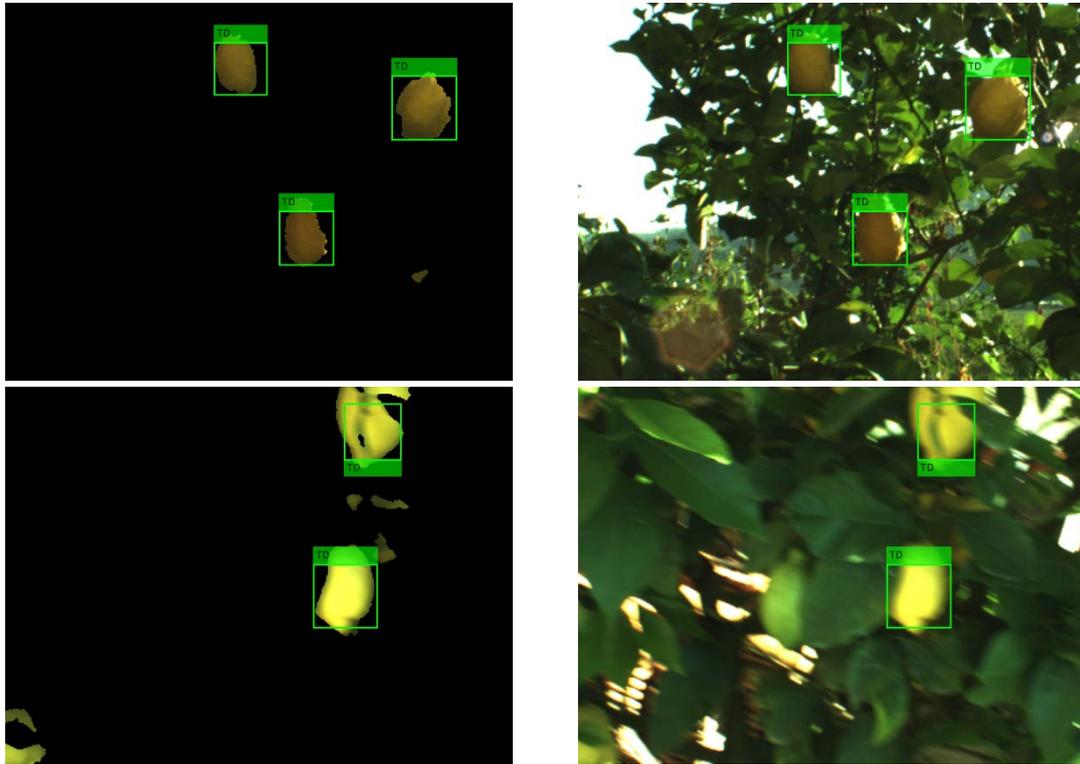


Fig. 8.6. Exemplos de sucesso na detecção de frutos na presença de um flare e limão verde: (esquerda) imagem segmentada; (direita) imagem RGB original.

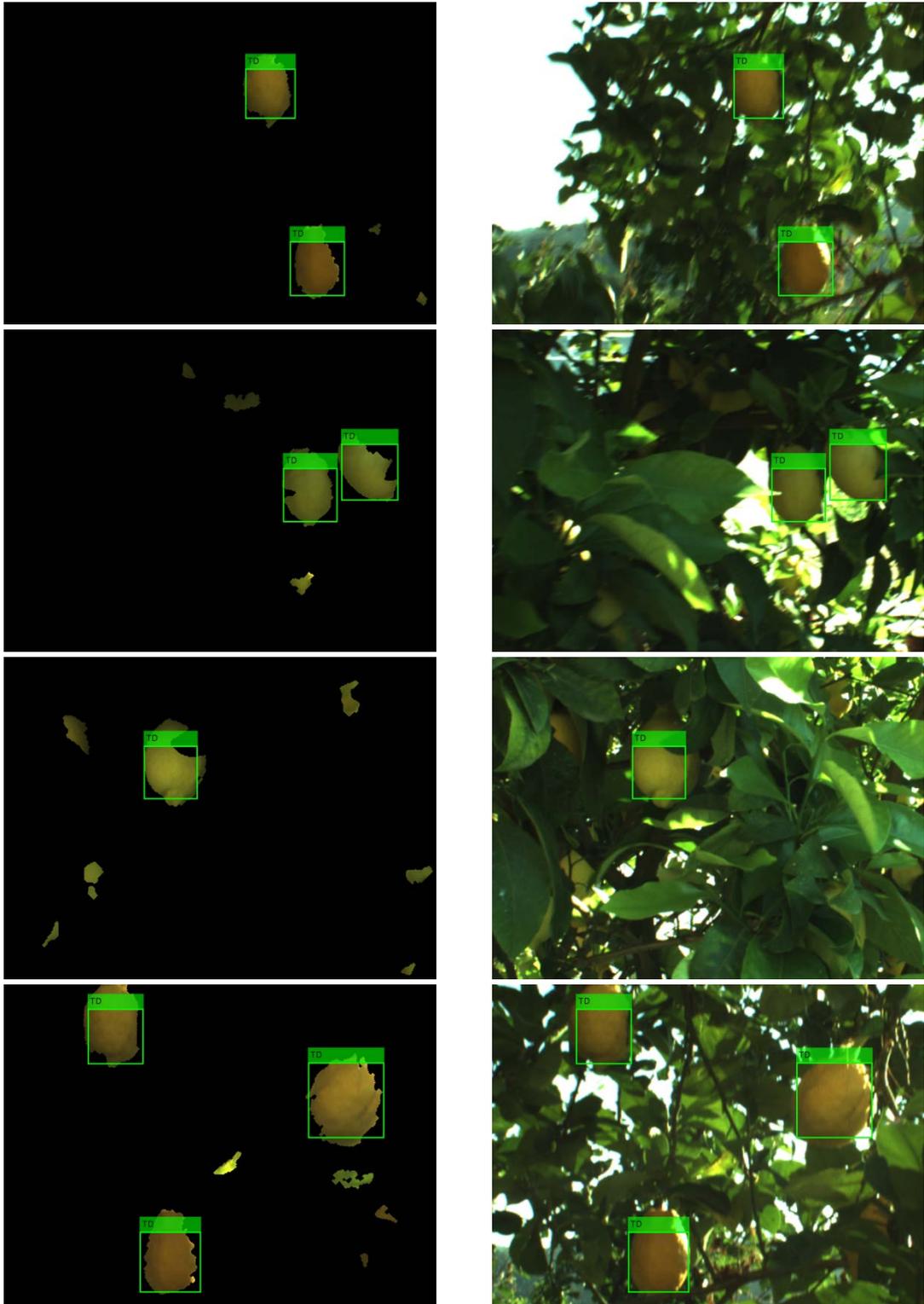


Fig. 8.7. Exemplos de sucesso na detecção de frutos: (esquerda) imagem segmentada; (direita) imagem RGB original.



## 9. CONCLUSÕES

O sistema de visão artificial em cenário natural para a segmentação e detecção de limões, alvo da presente dissertação, tendo em consideração as condições da sua validação, através das metodologias propostas, em cenários sem iluminação artificial, perante os resultados alcançados, foi conseguido com sucesso.

A integração de metodologias, até então dispersas, permitiu alcançar resultados com desempenho semelhante ao nível do estado da arte, com as vantagens adicionais de ser um sistema sem peças móveis que possam estar sujeitas a falhas durante as operações reais, apto a operar em cenários dinâmicos e de baixa latência. Apesar desta não ter sido avaliada desde a aquisição das imagens até à detecção dos frutos, mas atendendo aos resultados apresentados pelas metodologias avaliadas, a operação de segmentação de cada imagem é realizada em 145 ms e, no pior cenário, a detecção é feita em 50 ms, ou seja, o processamento da imagem pode ser realizado em menos de 200 ms nas condições atuais, sem otimização dos algoritmos para execução em baixa latência.

Os resultados conseguidos para a segmentação dos limões podem ser extrapoladas para outras áreas de aplicação, nomeadamente onde existirem outros elementos com tonalidades semelhantes às classes avaliadas, em que, a partir dos resultados apresentados no Anexo B, se pode focar o estudo na avaliação das combinações de componentes de cor, que se provou apresentar bom desempenho. Estes resultados permitem otimizar a gestão de tempo e recursos computacionais no treino de novos modelos de classificação.

A metodologia para treino dos classificadores SVM, baseada no algoritmo designado por trepa-colinas, foi fundamental para se poder fazer a extensão de combinações de componentes de cor e as parametrizações dos modelos de classificação avaliados. Apesar de se poder analisar mais exaustivamente o tempo decorrido no treino de cada classificador individualmente, através do acesso ao tempo despendido por cada processo submetido ao *cluster*, estima-se que as mesmas simulações demorassem de 10 a 12 meses num computador com um processador i7 a processar durante 24 horas. Mesmo com recurso ao *cluster* seria impraticável a avaliação exaustiva de todas as combinações de parâmetros pelo método designado de força-bruta.

Relativamente aos resultados alcançados na segmentação dos frutos do restante cenário foi selecionada uma configuração que atinge uma  $TPR = 93.14\%$  para uma  $FPr = 2.34\%$ , com um tempo médio de execução de 145 ms, usando um classificador SVM global.

Contrariamente ao que se esperava inicialmente, o classificador SVM em cadeia foi o que apresentou pior desempenho. A causa destes resultados pode estar relacionada com o facto das amostras de todas as classes terem sido obtidas em primeiro lugar e, só depois terem sido aplicados os classificadores. Isto significa que, se uma determinada classe (negativa) for

fortemente afetada pelos níveis de classificação anteriores ao classificador criado para a sua classe, pode dar origem a uma supressão significativa das amostras de treino e a um desequilíbrio significativo do número de amostras dessa classe e da classe positiva, a dos limões. A forma de ultrapassar este aspeto negativo é através da ampliação da base de dados, que poderá ser através da aplicação de um determinado classificador e só depois criar as amostras da classe seguinte. Neste processo pode-se concluir que é desnecessário avaliar determinadas classes, e que outras classes (grupo de elementos que partilham as mesmas características) podem surgir com o objetivo de criar um modelo globalmente robusto.

A vantagem apresentada nos resultados dos classificadores SVM independentes, relativamente aos em cadeia, está precisamente no modo como são treinados. Independentemente dos classificadores existentes na árvore de decisão, os classificadores SVM independentes aprendem a distinguir os limões de uma determinada classe com todas as amostras disponíveis, sejam elas durante a aplicação do método de segmentação, suprimidas nos níveis anteriores da árvore de decisão, ou não.

Por motivos de uniformização no treino dos três métodos de segmentação foi usado um método de divisão das amostras existentes para treino, validação e teste distinto do tradicional *k-fold*. Contudo, perante os resultados alcançados, o domínio do classificador SVM global relativamente aos outros avaliados, e o treino deste classificador, podia ter sido realizado pelo tradicional *k-fold*. Porém, as condições em que o treino dos classificadores foi realizado, bem como os seus resultados comprovam que a metodologia sugerida para a divisão das amostras para as fases de treino e validação é válida para populações de pequenas dimensões e para a atenuação da influência do ruído existente.

Os resultados obtidos na segmentação, quando comparados com o estado da arte, superam abordagens anteriores com sistemas monoculares, mas são ligeiramente inferiores à resposta obtida pelos sistemas multispectrais. No entanto, a mesma metodologia responde de forma distinta a diferentes frutos. Não existindo outros resultados relativos à segmentação de limões, e fazendo a analogia possível com os resultados alcançados para outros frutos, pode-se afirmar que a metodologia de segmentação proposta encontra-se numa posição idêntica ao estado da arte para outros frutos.

Relativamente à deteção, esta é realizada a partir da imagem segmentada e não da imagem original.

Quanto aos resultados alcançados, estes superam a maioria das metodologias previamente identificadas. Destacam-se ainda os resultados alcançados por métodos baseados em *Deep Learning*, os quais, pelos requisitos computacionais que exigem no treino dos modelos, são superiores às das restantes metodologias revistas.

Conseguiram-se resultados com baixa latência, inferiores a 50 ms. Destes, destacam-se os resultados alcançados através da aplicação do método de *Haar*, tendo-se conseguido uma

*Precision* = 80.19 % e uma *Recall* = 92.39 %, e ainda um outro classificador HOG que apresenta menos FA, onde se obtém uma *Precision* = 85.37 % e uma *Recall* = 76.09 %.

Perante as condições iniciais adversas, de um cenário de iluminação não controlada e recorrendo apenas a imagens RGB, o sucesso global da metodologia subjacente a esta tese está na integração de metodologias já avaliadas de forma dispersa e das metodologias empregues no treino dos classificadores, que permitem, a partir de uma população inicial reduzida e com ruído, e com a metodologia proposta baseada na designada por trepa-colinas, avaliar a extensão de configurações propostas e a convergência com resultados ao nível do estado da arte.

## 9.1 Perspetivas de desenvolvimento

Não obstante os bons resultados obtidos é expetável que o sistema possa falhar em condições de muito baixa luminosidade, pelo que é desejável a introdução de iluminação artificial para assegurar o correto funcionamento em situações de fraca intensidade de iluminação natural. Excetuando-se as situações de luz solar intensa, a iluminação artificial pode facilitar o processo de segmentação e deteção, ao aumentar o contraste aparente dos frutos com o restante cenário (ou elementos específicos deste).

Sobre a metodologia de deteção, torna-se necessário implementar métodos, ou adaptar os existentes, de modo a conseguir-se detetar corretamente frutos sobrepostos. A transformada de *Watershed* e o método de *X-means* foram aplicados com sucesso na literatura revista. Deverá ser ponderada a avaliação da proposta de Hannan *et al.*, referida na dissertação, para detetar os frutos a partir do centro de massa estimado a partir do perímetro, com as devidas precauções devido à forma natural dos limões.

Ainda sobre a deteção, torna-se necessário alargar os testes dos classificadores a um número superior de imagens, bem como introduzir um método que permita detetar corretamente frutos em função das suas dimensões, ou ainda da distância estimada da câmara aos frutos.

A integração da metodologia proposta num braço robótico, ou noutra equipamento, com vista ao desenvolvimento da "Máquina Autónoma de Classificação de Fruta com Colheita Automática", premiada pelo Prémio CA 2014 na categoria de investigação e inovação tecnológica, os testes ditarão as futuras necessidades para a adequação da proposta às condições reais de funcionamento.

Antevê-se a necessidade de rastrear os frutos ao longo de uma sequência de imagens para confirmação da verdadeira existência dos frutos, estimação da sua posição e determinação do melhor trajeto a realizar para a efetiva colheita individual dos frutos. Para esta função poderá recorrer-se ao filtro de *Kalman* e aos desenvolvimentos efetuados neste campo aquando do início do programa doutoral, cuja aplicação visava a deteção de objetos para aplicações industriais de Realidade Aumentada.

Após a detecção dos frutos existem duas etapas, posteriores, que são fundamentais para o sucesso do equipamento:

- A garra, módulo responsável por efetivar a colheita dos frutos;
- Os movimentos realizados pelo braço robótico a fim de evitar obstáculos naturais desde a detecção do fruto na árvore até à sua colheita.

## 9.2 Contributos

Da dissertação subjacente à presente tese resultaram diversos contributos, dos quais se destacam os referentes à detecção de limões, em cenários naturais de iluminação não controlada, através de um sistema de visão simples, apenas com uma câmara policromática, passível de implementação em ambiente real e apto a operar em cenários dinâmicos.

A real valorização destes contributos terá a sua ênfase na implementação da metodologia proposta num equipamento robotizado de colheita individual de fruta. Neste é recomendável a introdução de iluminação artificial para também poder funcionar nos períodos de fraca iluminação solar e, assim, conseguir-se o retorno do investimento num mais curto intervalo de tempo.

Outro contributo é dado pela metodologia de treino dos classificadores SVM e pela seleção das características (*features*) a analisar na fase da segmentação, a qual se revelou ser eficaz, considerando os resultados de segmentação atingidos e a poupança de tempo conseguida, ao invés de se realizarem as avaliações pelo método designado de força-bruta. Aquele tem o mesmo risco da uma análise PCA, onde não é totalmente eliminado o risco de serem descartadas características aparentemente fracas, mas que, quando combinadas de forma diferente, originam a maximização da diferença entre as classes em análise.

Também pode ser entendido como contributo a metodologia aplicada como alternativa ao *k-fold*, a qual se mostrou ser eficaz no treino dos classificadores, designadamente no caso de imagens adquiridas em ambientes reais de iluminação não uniforme, pelo fato das amostras das várias classes conterem ruído no seu sinal. A metodologia proposta permite filtrar esse ruído e concentrar o treino dos classificadores nas condições que efetivamente caracterizam cada classe. Esta metodologia é válida no treino de classes bem definidas.

Outro contributo será a disponibilização, em livre acesso, da base de dados de imagens e máscaras dos frutos, assim como os resultados alcançados, para que possa servir de base de trabalho a futuros desenvolvimentos da comunidade científica.

# ANEXO A. PROCESSO DE SUBMISSÃO DE TAREFAS NO CLUSTER

A submissão de trabalhos a realizar pelo *cluster* pode ser realizada pela submissão de processos “individuais”, isto é, com uma única tarefa, ou através de processos com múltiplas tarefas.

Um dos motivos de se recorrer a um *cluster* HPC é a execução de pequenos programas em que apenas varia os respetivos parâmetros de entrada. Tipicamente, só interessa analisar a resposta desses programas após terem sido realizadas todas as simulações, pelo que são submetidos processos com múltiplas tarefas associadas.

A submissão de processos com tarefas únicas, nestes casos, obriga a um controlo mais elevado na confirmação de execução de todos os processos. Enquanto na submissão de um único processo com múltiplas tarefas associadas apenas é necessário verificar se esse (único) processo foi executado com sucesso.

Contudo, em função da disponibilidade do *cluster* e dos recursos exigidos, pode haver ganhos em subdividir as tarefas a executar por vários processos, mas evitando a submissão de processos com tarefas únicas.

Como a diferença entre cada tarefa são os parâmetros de entrada dos programas/funções a executar, torna-se mais prático a elaboração de *scripts* simples para gerar os ficheiros de submissão, ou seja, os ficheiros com os processos (Fig. A.1) e as tarefas a executar (Fig. A.2). Porém, há cuidados a ter na geração destes ficheiros e configurações do *cluster*.

É importante nestes casos saber aproximadamente o tempo de execução de cada tarefa e é aconselhável que as tarefas submetidas no mesmo processo possuam um tempo de execução semelhante. O tempo de execução de cada tarefa é necessário para definir o intervalo de tempo a solicitar ao *cluster* os recursos computacionais pretendidos. E todas as tarefas devem ter uma duração semelhante para evitar que o *cluster* bloqueie todos os recursos reservados até à conclusão da última tarefa.

No *cluster* utilizado nos trabalhos apresentados nesta dissertação, é requerido que o número de tarefas a executar seja par, exceção feita aos processos de tarefa única, e esse mesmo número de tarefas deve ser múltiplo do número de cores reservados (designados por “ntasks” na Fig. A.1).

Pode ainda ser criada um terceiro ficheiro que ao ser executado faz a submissão de todos os processos pretendidos, em vez de se ordenar manualmente a execução de cada processo.

A Fig. A.1 representa as instruções do ficheiro (*batch*) usado na submissão de um processo ao *cluster*.

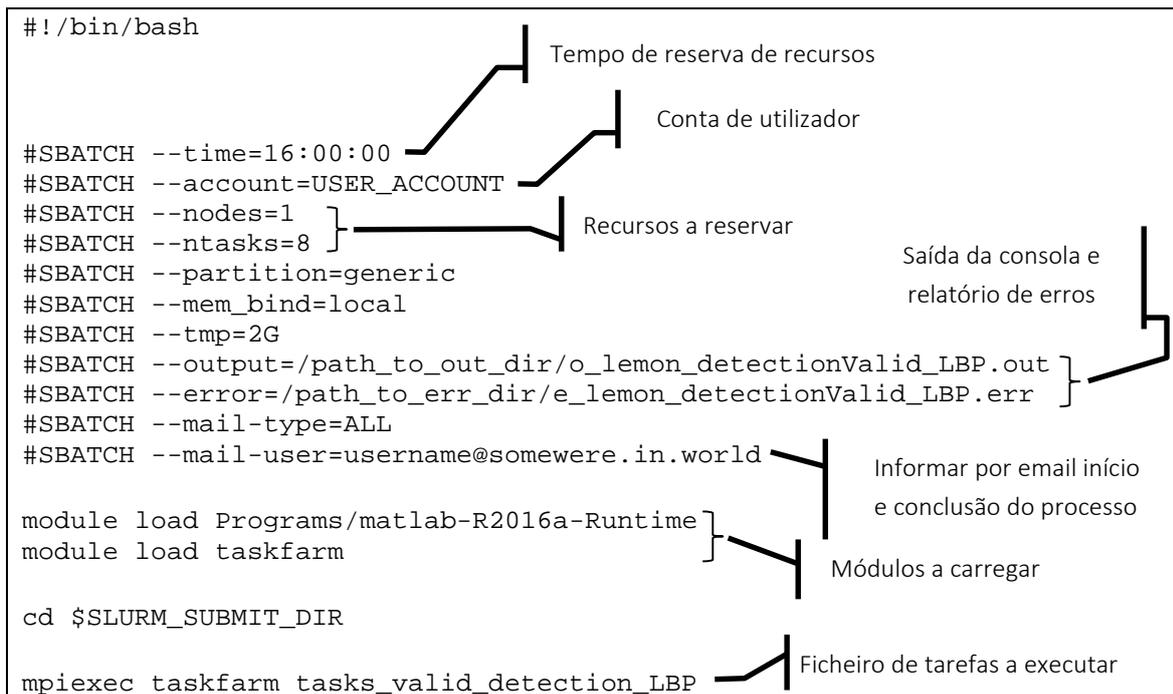


Fig. A.1. Ficheiro batch de submissão do processo.

Nos recursos computacionais a reservar não é explicitada a memória RAM por não haver exigências especiais nas simulações realizadas. O importante é definir corretamente, com números inteiros, o número de *cores* (ntasks) a reservar e os nós de computação (nodes), sabendo que cada nó possui 24 *cores*.

Em função dos requisitos necessários pode ser necessário ajustar a partição (*partition*) de acordo com o definido no manual do *cluster* [102]. Caso o processo execute múltiplas tarefas é invocado o *script* correspondente, exemplificado na Fig. A.2.

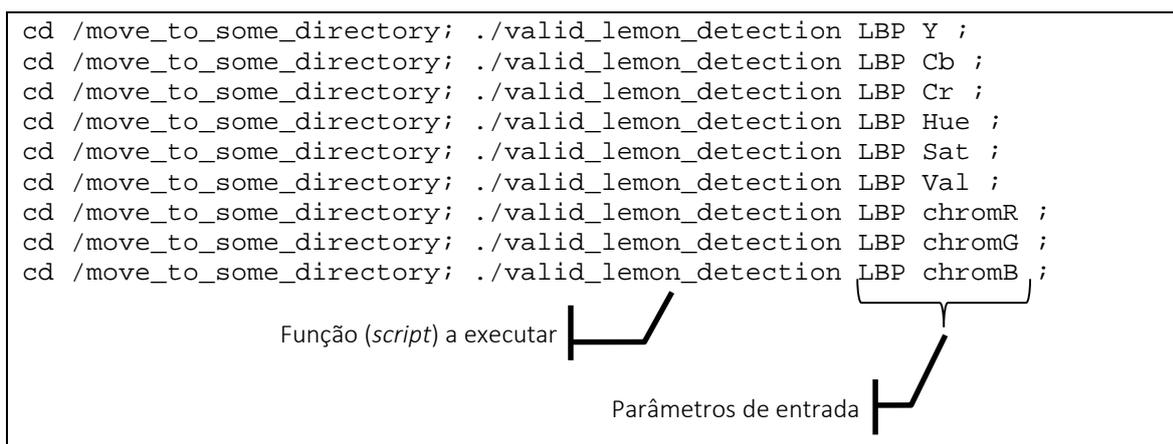


Fig. A.2. Ficheiro com tarefas a submeter ao cluster.

# ANEXO B. COMBINAÇÕES DE CARACTERÍSTICAS COM MELHOR AUC DOS CLASSIFICADORES SVM

Nesta secção são apresentadas as combinações características que representam as dez AUC (*Area Under the Curve*) mais elevadas de todas as configurações avaliadas para cada classe negativa. Os gráficos deste capítulo são relativos aos dados de validação.

Note-se que o facto de uma determinada combinação de características pode apresentar pontualmente uma a melhor *TPr* para uma dada *FPr* e essa combinação de características de cor não ser representada nos gráficos que se seguem, devido a numa análise alargada para um intervalo de *FPr* representar uma AUC inferior a outras combinações de características.

Conforme é ilustrado na Fig. B.3, a melhor configuração para *FPr* > 1.5 % não possui a melhor AUC de acordo com a equação (6.12). Nas simulações realizadas é definido o intervalo de *FPr* para o qual deve ser determinada a respetiva AUC, precisamente para minimizar a ocorrência da situação representada na Fig. B.3. Contudo, há situações onde não foi possível filtrar estes casos.

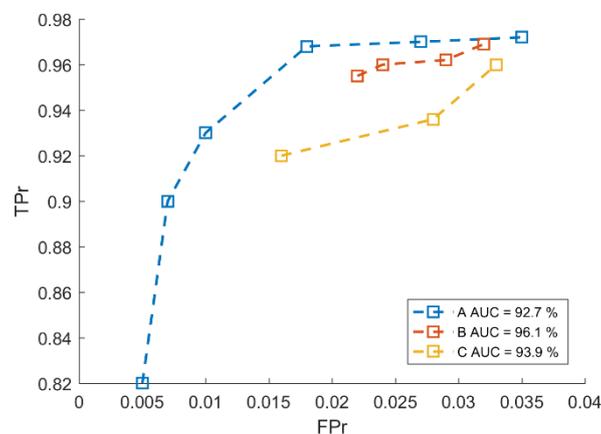


Fig. B.3. Comportamento da métrica AUC.

## 1) Classificador SVM global

Considerando  $\Delta t_{rel} \leq 500$  ms e  $FPr \leq 6\%$ , as combinações de características de cor representativas das dez melhores AUC são representadas na Fig. B.4.

A configuração designada de “SVM Global 2”, mencionada na secção 7.1, é referente à melhor combinação de características de cor representada no gráfico. Por sua vez, a combinação relativa

à “SVM Global 1” não se encontrada representada no gráfico da Fig. B.4 pelo motivo descrito no início deste capítulo.

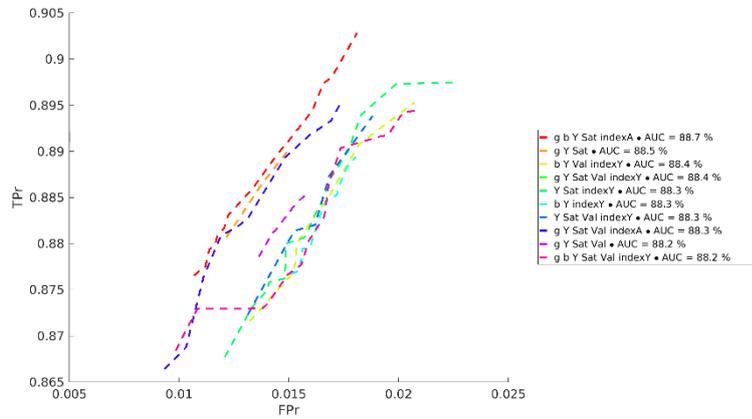


Fig. B.4. Combinações de características com melhor AUC para o classificador SVM global.

## 2) Classificadores SVM independentes

Os gráficos das Fig. B.5 a Fig. B.8 apresentam as combinações de características com melhor AUC. Note-se que devido à construção desta métrica, a combinação das características *r*, *Y* e *indexa* é a que apresenta melhor AUC na Fig. B.5, apesar de haver combinações com melhor rácio *TPr*/*FPr*.

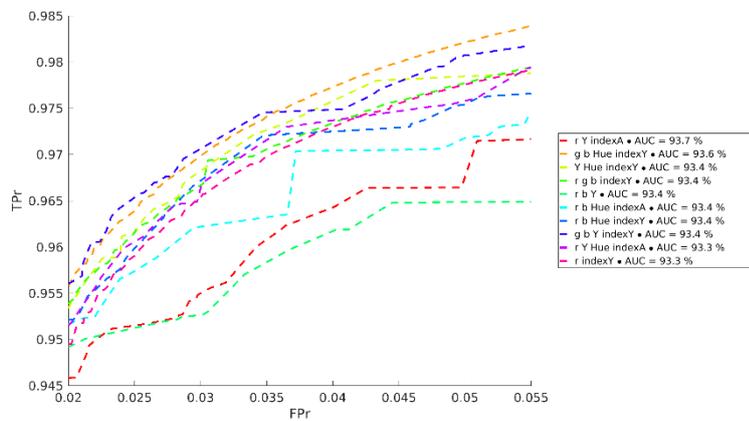


Fig. B.5. Combinações de características com melhor AUC para a classe da folhagem.

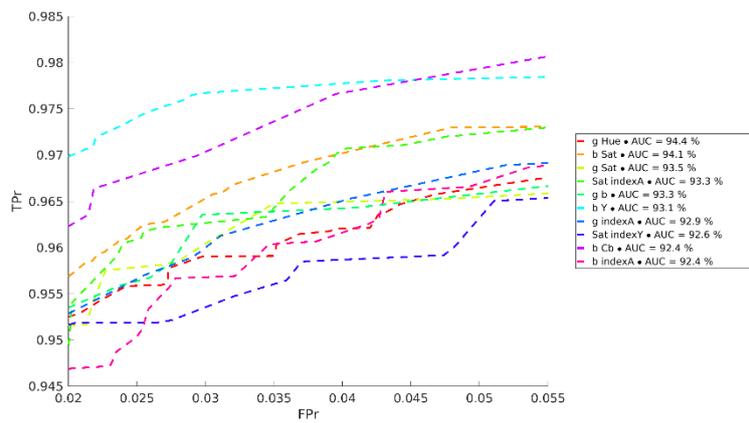


Fig. B.6. Combinações de características com melhor AUC para a classe dos ramos.

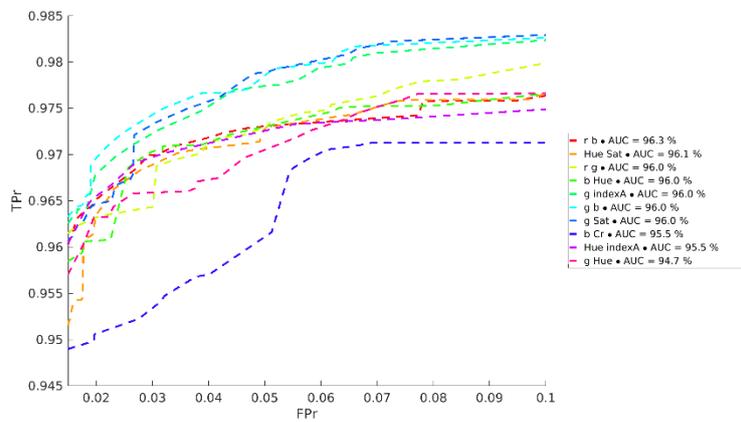


Fig. B.7. Combinações de características com melhor AUC para a classe dos elementos cinza.

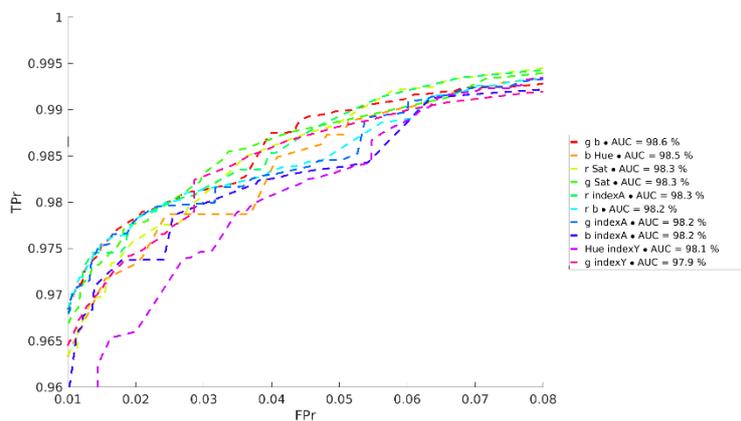


Fig. B.8. Combinações de características com melhor AUC para a classe dos elementos laranja.

### 3) Classificadores SVM em cadeia

As figuras seguintes apresentam as combinações de características com melhor AUC nas avaliações realizadas com os classificadores SVM em cadeia.

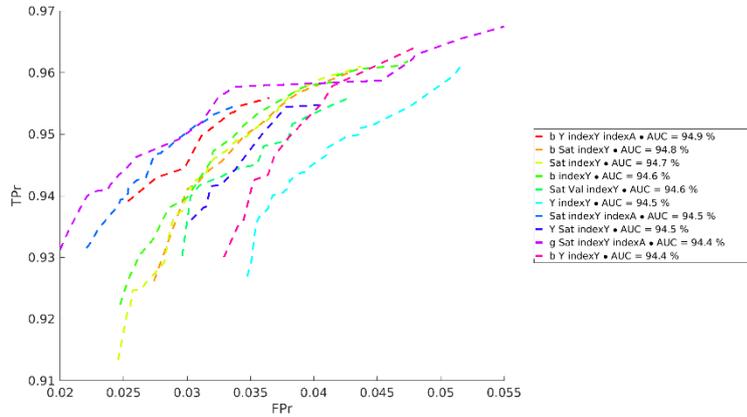


Fig. B.9. Combinações de características com melhor AUC dos classificadores SVM independentes para a classe da folhagem.

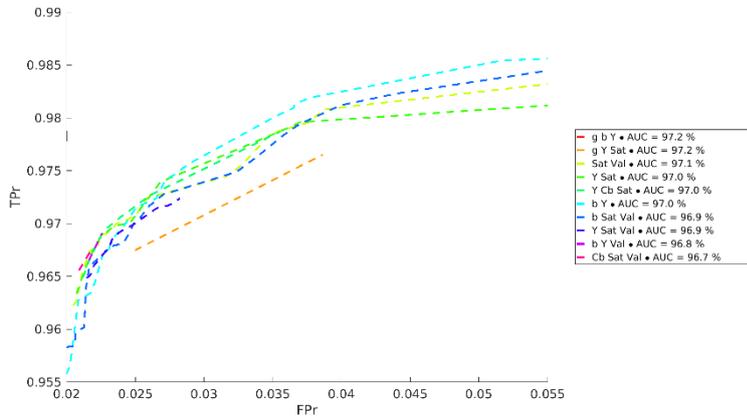


Fig. B.10. Combinações de características com melhor AUC dos classificadores SVM independentes para a classe dos ramos.

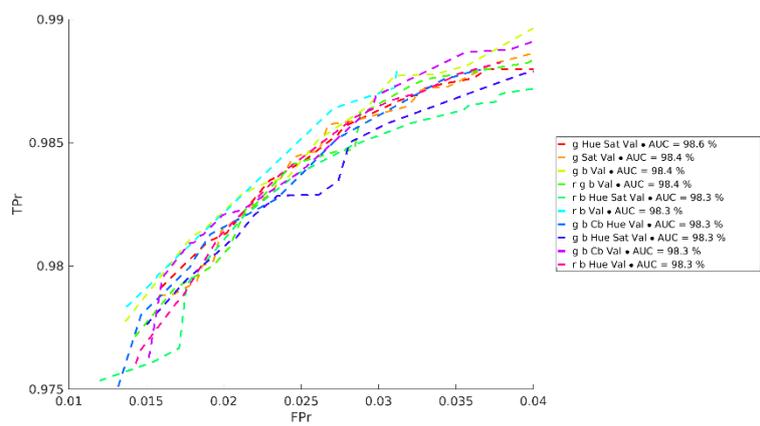


Fig. B.11. Combinações de características com melhor AUC dos classificadores SVM independentes para a classe dos elementos cinza e laranja.

# ANEXO C. DESEMPENHO ISOLADO DOS CLASSIFICADORES SVM EM CADEIA

Neste anexo são apresentados os níveis de desempenho individuais dos classificadores treinados num processo em cadeia, como complemento aos resultados apresentados na secção 7.3.

Devido à estratégia de otimização do processo em cadeia, nem todas as instâncias inicialmente previstas para cada classe (Tabela 6.1) são aqui consideradas nos resultados de validação dos classificadores, dependendo do processo de classificação nos nós anteriores da árvore de decisão, tal como esquematizado na Fig. 7.9.

Assim, é previsto que quanto mais níveis se avançarem na árvore de decisão, maior o *FPr* aqui apresentado, relativamente ao apresentado na secção 7.3 para a mesma configuração. O mesmo raciocínio é válido para o *TPr*.

Nesta perspetiva, uma configuração que nesta secção apresente *TPr* = 98% significa que classifica corretamente 98 % das instâncias de validação (não retidas pelos níveis anteriores) da classe dos limões. Ou seja, a mesma configuração na secção 7.3 irá apresentar como *TPr* o equivalente a 98% da *TPr* da configuração seleccionada no nível anterior.

## 1) Limões vs folhagem verde

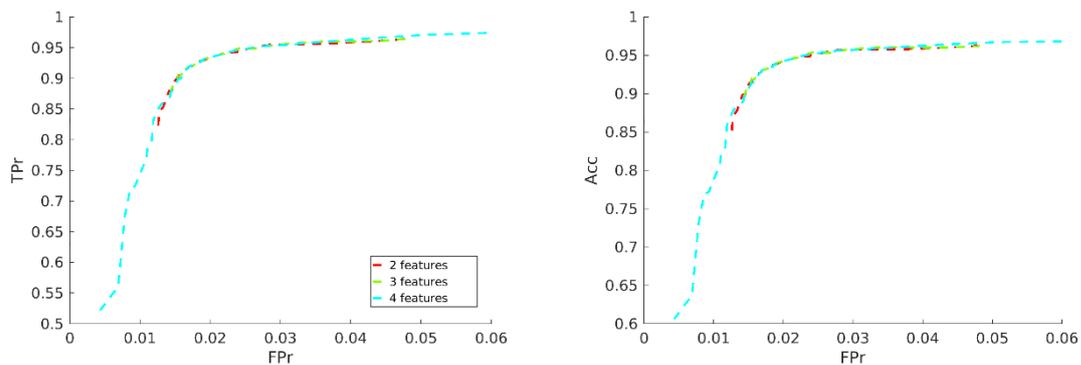


Fig. C.12. Desempenho de classificadores SVM em cadeia para folhagem até quatro características e  $\Delta t_{rel} \leq 500$  ms: (esquerda) curva ROC; (direita) Acc correspondente.

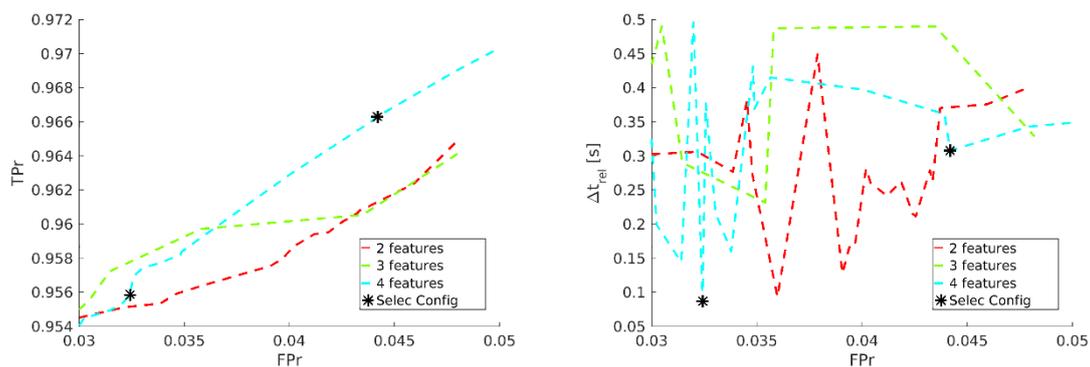


Fig. C.13. Detalhe do desempenho de classificadores SVM em cadeia para folhagem até quatro características: (esquerda) curva ROC; (direita)  $\Delta t_{rel}$  correspondente.

Tabela C.1. Configurações selecionadas limões vs folhagem

	Config A	Config B
$\sigma$	6.7608	35.4813
Características	<i>chromG Sat indexY indexA</i>	<i>chromG Hue Sat indexY</i>
$C$	1.0233	1.2882
$c$	0.3000	1
<i>outliers</i>	0 %	3.50 %
$TPr$	95.59 %	96.63 %
$FPr$	3.24 %	4.42 %
$Acc$	95.79 %	96.44 %
$\Delta t_{rel}$ [s]	0.087	0.308

## 2) Limões vs ramos

Os resultados dos classificadores desta classe variam em função da configuração selecionada no nó de decisão anterior: a configuração selecionada para segmentar os limões da classe da folhagem.

Desta forma, são apresentados os resultados para as duas configurações no nó de decisão anterior e comparados diretamente com os resultados de cada abordagem.

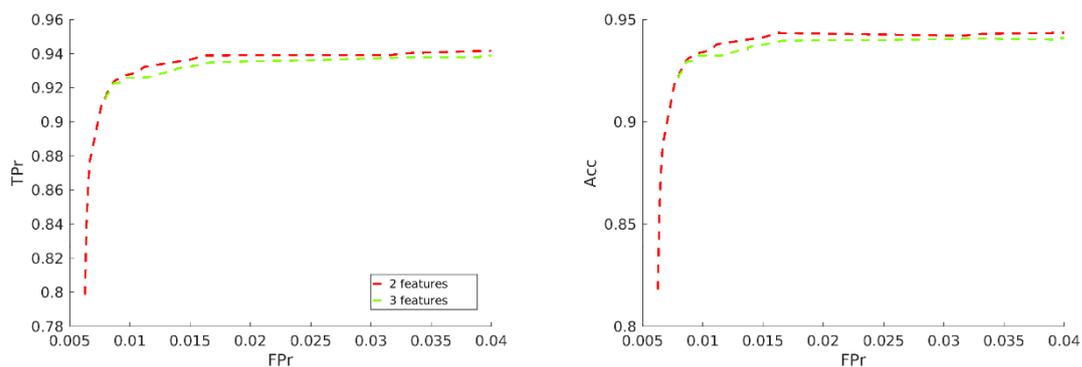


Fig. C.14. Desempenho de classificadores SVM em cadeia para ramos até três características, para a Config A e  $\Delta t_{rel} \leq 500$  ms: (esquerda) curva ROC; (direita) Acc correspondente.

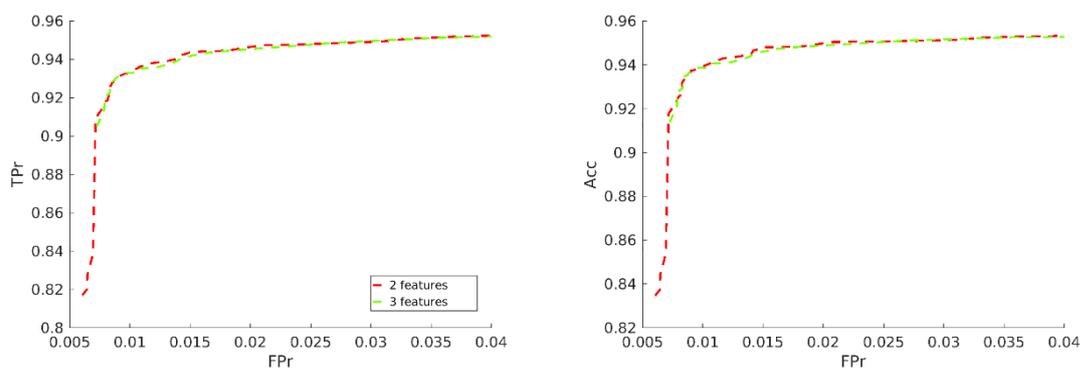


Fig. C.15. Desempenho de classificadores SVM em cadeia para ramos até três características, para a Config B e  $\Delta t_{rel} \leq 500$  ms: (esquerda) curva ROC; (direita) Acc correspondente.

Tabela C.2. Configuração selecionada limões vs ramos

	Config C
$\sigma$	0.1820
Características	<i>chromB Y</i>
$C$	0.6310
$c$	1
<i>outliers</i>	0%
<i>TPr</i>	99,04%
<i>FPr</i>	9,35%

Acc	98,66%
$\Delta t_{rel}$ [s]	0,371

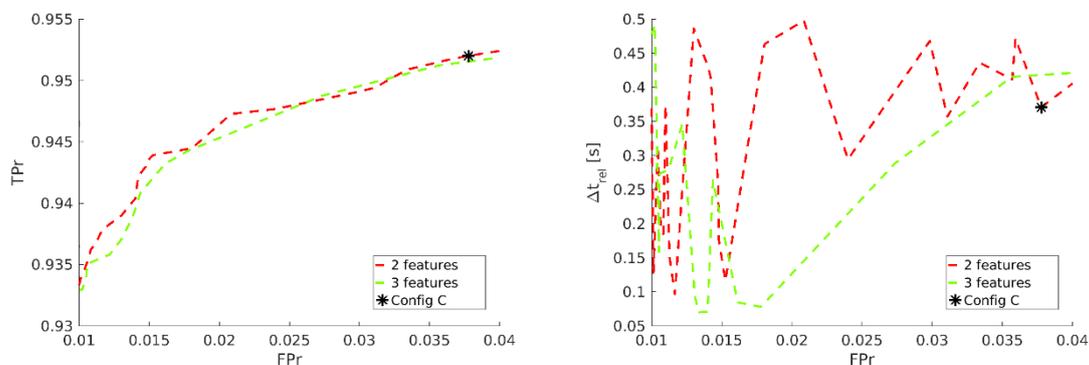


Fig. C.16. Detalhe do desempenho de classificadores SVM em cadeia para ramos até três características, para a Config B: (esquerda) curva ROC; (direita)  $\Delta t_{rel}$  correspondente.

### 3) Limões vs elementos cinza e laranja

Devido à elevada supressão de instâncias de treino da classe elementos cinza (Tabela 7.10) foi adotada a estratégia de treinar um classificador considerando como classe negativa a união das amostras dos elementos cinza e laranja. Assim, a nova estrutura da árvore de decisão assume a representação esquematizada na Fig. 7.31.

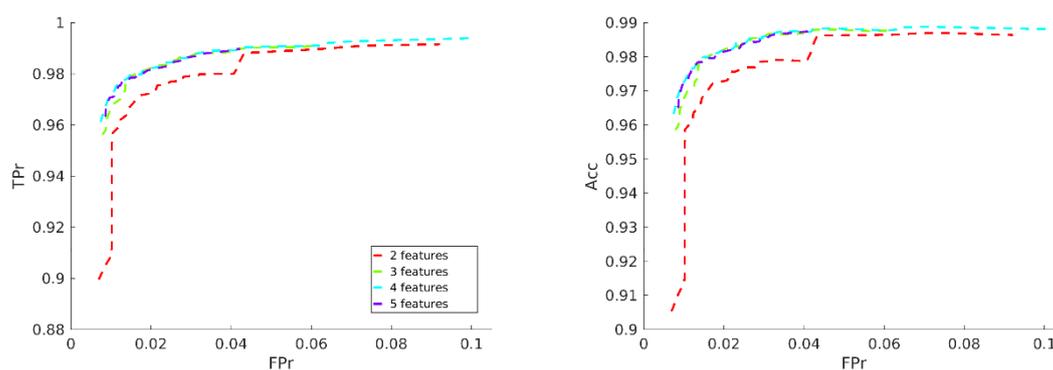


Fig. C.17. Desempenho de classificadores SVM em cadeia para elementos cinza e laranja, até cinco características, para a Config B, Config C e  $\Delta t_{rel} \leq 500$  ms: (esquerda) curva ROC; (direita) Acc correspondente.

Dado ser um processo de treino em cadeia, os resultados apresentados para o treino dos classificadores que opõem os limões aos elementos cinza e laranja são influenciados pela aplicação da “Config B” no classificador SVM no nó de decisão relativo à folhagem e, no nível seguinte, é aplicada a “Config C” para separar os *pixels* dos ramos dos limões.

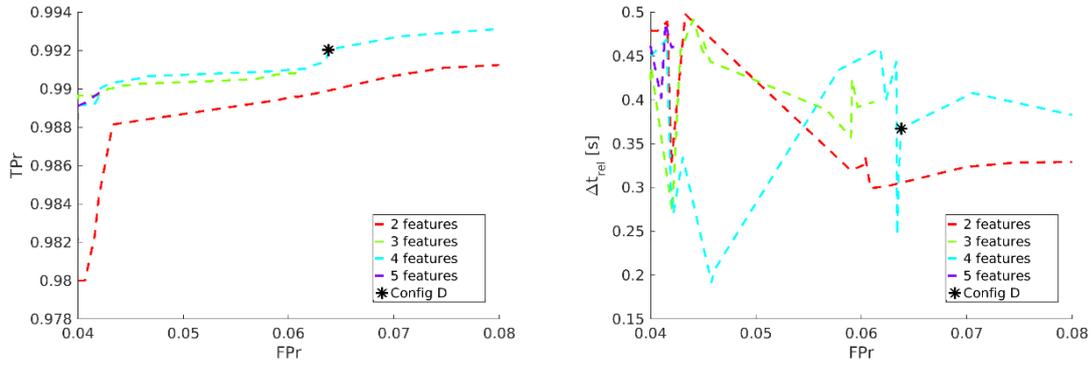


Fig. C.18. Detalhe do desempenho de classificadores SVM em cadeia para elementos cinza e laranja, até cinco características, para a Config B, Config C: (esquerda) curva ROC; (direita)  $\Delta t_{rel}$  correspondente.

Tabela C.3. Configuração selecionada limões vs elementos cinza e laranja

	Config D
$\sigma$	2.4434
Características	<i>chromG chromB Cb Val</i>
$C$	6.6374
$c$	0.5000
<i>outliers</i>	0%
$TPr$	99.20%
$FPr$	6.38%
$Acc$	98.86%
$\Delta t_{rel}$ [s]	0.367

## ANEXO D. PRODUÇÃO CIENTÍFICA

Lista de publicações científicas em revistas internacionais e em *proceedings* de eventos internacionais:

- [111] R. Oliveira, J. T. Farinha, I. Fonseca, and F. M. Barbosa, “Augmented reality system for maintenance of high-voltage systems,” in *Proc. of UPEC*, 2016. (A decorrer indexação ao IEEE XPlore)
- [104] R. Oliveira, T. Farinha, H. Raposo, and N. Pires, “Augmented reality and the future of maintenance,” in *Proc. of Maintenance Performance Measurement and Management (MPMM) Conference 2014*. Coimbra: Imprensa da Universidade de Coimbra, Sept. 2014, pp. 81–88.
- [112] R. Oliveira, J. André, J. T. Farinha, and J. N. Pires, “Monocular model-based on augmented reality,” in *Poster session presented at the VISion Understanding and Machine intelligence – VISUM*, 2014.
- [105] J. T. Farinha, R. Oliveira, I. Fonseca, and F. M. Barbosa, “The convergence between predictive maintenance and augmented reality to aid renewable energy equipment availability,” *International Journal on Engineering Applications (IREA)*, vol. 2, no. 5, 2014. [Online]. Available: <https://doi.org/10.15866/irea.v2i5.1916>
- [113] H. Raposo, J. T. Farinha, R. Oliveira, L. A. Ferreira, and J. André, “Time replacement optimization models for urban transportation buses with indexation to fleet reserve,” in *Proc. of Maintenance Performance Measurement and Management (MPMM) Conference 2014*. Coimbra: Imprensa da Universidade de Coimbra, Sept. 2014, pp. 41–48.
- [114] F. Rodrigues, I. Fonseca, R. Oliveira, and J. T. F. Farinha, “Maintenance management in web asp.net mvc applications,” in *Proc. of Maintenance Performance Measurement and Management (MPMM) Conference 2014*. Coimbra: Imprensa da Universidade de Coimbra, Sept. 2014, pp. 95–101.
- [115] J. Ferreira, L. Oliveira, and R. Oliveira, “Real time data collection and processing for aircraft maintenance enhancement (REACT),” in *Proc. of Maintenance Performance Measurement and Management (MPMM) Conference 2014*. Coimbra: Imprensa da Universidade de Coimbra, Sept. 2014, pp. 11–14.
- [116] J. T. Farinha, I. Fonseca, R. Oliveira, and H. Raposo, “Cmms an integrated view from maintenance management to on-line condition monitoring,” in *Proc. of Maintenance Performance Measurement and Management (MPMM)*

*Conference 2014*. Coimbra: Imprensa da Universidade de Coimbra, Sept. 2014, pp. 137–141.

- [117] S. Singh, D. Galar, U. Kumar, R. Oliveira, and T. Farinha, “Augmented reality - technology for effective maintenance in railway sector,” in Proc. of the 2nd International Conference on Industrial Engineering ICIE 2013. EXCELLENT PUBLISHING HOUSE, 2013, pp. 727–731.
- [118] R. Oliveira, T. Farinha, S. Singh, and D. Galar, “An augmented reality application to support maintenance – is it possible?” in Proc. Maintenance Performance Measurement and Management (MPMM) 2013, 2013, pp. 260–271. [Online]. Available: [https://-online.unileoben.ac.at/~mu\\_online/-voe\\_main2.getVollText?pDocumentNr=142932&pCurrPk=32372](https://-online.unileoben.ac.at/~mu_online/-voe_main2.getVollText?pDocumentNr=142932&pCurrPk=32372)
- [119] J. T. Farinha, R. Oliveira, I. Fonseca, and L. A. Ferreira, “Technical asset management beyond tomorrow,” in Proc. 2nd World Maintenance Forum, Lugano, 2013.
- [120] R. Oliveira, J. T. Farinha, I. Fonseca, L. A. Ferreira, and F. Armas, “Augmented reality as a new level for maintenance efficiency,” in Proc. of The 2nd international workshop and congress on eMaintenance, 2012, pp. 49–55. [Online]. Available: [https://-pure.ltu.se/~portal/~files/-41489674/~Proceedings\\_eMaintenance\\_2012.pdf](https://-pure.ltu.se/~portal/~files/-41489674/~Proceedings_eMaintenance_2012.pdf)
- Encontra-se submetido à revista internacional *Sensors* (indexada ISI) um artigo intitulado: Citrus fruits segmentation in unstructured scenarios using a monocular system and a SVM-based decision tree. O artigo foi aceite após uma Major Revision.
  - Encontra-se em fase de conclusão outro artigo, a submeter à revista internacional *Sensors* (indexada ISI), sobre segmentação e deteção de limões. Paralelamente a esta publicação será disponibilizada a base de dados à comunidade científica.

Lista de publicações científicas em revistas nacionais e em atas de eventos nacionais:

- [121] R. Oliveira, J. T. Farinha, and I. Fonseca, “Realidade aumentada e holografia,” *Manutenção*, vol. 130-131, pp. 58–61, 2016. [Online]. Available: <http://www.revistamanutencao.pt/revista/>
- [122] J. T. Farinha and R. Oliveira, “Máquina autónoma de classificação de fruta com colheita automática,” *Agrobótica (Agrotec)*, vol. 3, pp. 8–11, 2015.
- [123] R. Oliveira and T. Farinha, “A manutenção de equipamentos de saúde - da pré-história ao futuro,” *Tecno Hospital*, vol. 61, pp. 11–17, Jan/Fev 2013.

- [124] R. Oliveira and T. Farinha, “Realidade aumentada – cenários virtuais em hospitais reais,” *Tecno Hospital*, vol. 57, pp. 34–38, Mai/Jun 2013.
- [125] J. T. Farinha, R. Oliveira, I. Fonseca, and L. A. Ferreira, “CMMS - para além do amanhã,” in *Proc. 17.º Congresso Ibero-americano de Manutenção; 12.º Congresso Nacional de Manutenção; 1.º Encontro de Manutenção dos Países de Língua Oficial Portuguesa*, 2013.
- [126] R. Oliveira and T. Farinha, “Realidade aumentada – ferramenta de apoio à manutenção em equipamento hospitalar,” *Tecno Hospital*, vol. 49, pp. 12–16, Jan/Fev 2012.

Participação nos seguintes eventos:

- 2016 Apresentação do tema: “Realidade Aumentada e Holografia”, integrado nas Jornadas de Manutenção: Papel da Manutenção na Indústria 4.0, organizado pela APMI.
- 2014 Apresentação do tema: “O Potencial da Realidade Aumentada no Aumento de Eficiência das Intervenções de Manutenção”, integrado nas Jornadas de Manutenção, organizado pela APMI.
- 2013 Apresentação do tema: “Realidade Aumentada – um novo paradigma na manutenção”, integrado no seminário de Manutenção de veículos pesados de transporte, organizado pela APMI.

Outras funções de âmbito científico durante o processo de doutoramento:

- Funções de revisor científico em:
  - IEEE Consumer Electronics Magazine.
  - Maintenance Performance Measurement and Management Conference (MPMM) 2014.
  - FAIM 2013 - 23<sup>rd</sup> International Conference on Flexible Automation & Intelligent Manufacturing.
- Foi membro do *Local Organizing Committee* no evento internacional Maintenance Performance Measurement and Management Conference (MPMM) 2014.



# REFERÊNCIAS

- [1] Agrobot. [Online]. Available: <http://www.agrobot.com/>
- [2] E. van Henten, J. Hemming, B. van Tuijl, J. Kornet, J. Meuleman, J. Bontsema, and E. van Os, "An autonomous robot for harvesting cucumbers in greenhouses," *Autonomous Robots*, vol. 13, no. 3, pp. 241–258, 2002. [Online]. Available: <http://dx.doi.org/10.1023/A:1020568125418>
- [3] G. Muscato, M. Prestifilippo, N. Abbate, and I. Rizzuto, "A prototype of an orange picking robot: past history, the new robot and experimental results," *Industrial Robot: An International Journal*, vol. 32, no. 2, pp. 128–138, 2005. [Online]. Available: <http://dx.doi.org/10.1108/01439910510582255>
- [4] R. Harrel, "Economic analysis of robotic citrus harvesting in florida," *Transactions of the ASAE - American Society of Agricultural Engineers*, vol. 30, no. 2, pp. 298–304, 1987.
- [5] A. Cavaco, R. Guerra, J. Silva, and D. Antunes, "Comparison of two different integration sphere configurations for 'rocha' pear determination of firmness and soluble solids by vis/nir spectroscopy," in *4th IASME/WSEAS International Conference on ENERGY, ENVIRONMENT, ECOSYSTEMS and SUSTAINABLE DEVELOPMENT (EEESD'08)*, June 2008, pp. 526–530.
- [6] Y. Liu, X. Sun, H. Zhang, and O. Aiguo, "Nondestructive measurement of internal quality of nanfeng mandarin fruit by charge coupled device near infrared spectroscopy," *Computers and Electronics in Agriculture*, vol. 71, Supplement 1, no. 0, pp. S10 – S14, 2010, special issue on computer and computing technologies in agriculture. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0168169909001835>
- [7] A. M. Fernandes, C. Franco, A. Mendes-Ferreira, A. Mendes-Faia, P. L. da Costa, and P. Melo-Pinto, "Brix, ph and anthocyanin content determination in whole port wine grape berries by hyperspectral imaging and neural networks," *Computers and Electronics in Agriculture*, vol. 115, pp. 88 – 96, 2015. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0168169915001490>
- [8] H. Lee, M. S. Kim, D. Jeong, S. R. Delwiche, K. Chao, and B.-K. Cho, "Detection of cracks on tomatoes using a hyperspectral near-infrared reflectance imaging system," *Sensors*, vol. 14, no. 10, pp. 18837–18850, 2014. [Online]. Available: <http://www.mdpi.com/1424-8220/14/10/18837>
- [9] N. V. Rivera, J. Gómez-Sanchis, J. Chanona-Pérez, J. J. Carrasco, M. Millán-Giraldo, D. Lorente, S. Cubero, and J. Blasco, "Early detection of mechanical damage in mango using {NIR} hyperspectral images and machine learning," *Biosystems Engineering*, vol. 122, pp. 91 – 98, 2014. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1537511014000506>

- [10] D. M. Bulanon, T. F. Burks, and V. Alchanatis, "A multispectral imaging analysis for enhancing citrus fruit detection," *Environmental Control in Biology*, vol. 48, no. 2, pp. 81–91, 2010.
- [11] F. Pla, F. Juste, and E. Molto, "An approach to citrus vision systems in robotic harvesting," in *1st International Seminar on use of On-Machine vision systems for the agricultural and bio industries*, 1991, pp. 47–59.
- [12] R. Fernández, H. Montes, C. Salinas, J. Sarria, and M. Armada, "Combination of RGB and multispectral imagery for discrimination of cabernet sauvignon grapevine elements," *Sensors*, vol. 13, no. 6, pp. 7838–7859, 2013. [Online]. Available: <http://www.mdpi.com/1424-8220/13/6/7838>
- [13] H. Okamoto and W. S. Lee, "Green citrus detection using hyperspectral imaging," *Computers and Electronics in Agriculture*, vol. 66, no. 2, pp. 201 – 208, 2009. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0168169909000258>
- [14] M. W. Hannan, T. F. Burks, and D. M. Bulanon, "A machine vision algorithm combining adaptive segmentation and shape analysis for orange fruit detection," *CIGR Ejournal*, vol. XI, p. 1281, 2009.
- [15] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, no. 1, pp. 62–66, Jan 1979.
- [16] K. E. Kane and W. S. Lee, "Multispectral imaging for in-field green citrus identification," in *2007 ASABE Annual International Meeting*, 2007.
- [17] R. Fernández, C. Salinas, H. Montes, and J. Sarria, "Multisensory System for Fruit Harvesting Robots. Experimental Testing in Natural Scenarios and with Different Kinds of Crops," *Sensors*, vol. 14, no. 12, pp. 23885–23904, 2014, filter wheel. [Online]. Available: <http://www.mdpi.com/1424-8220/14/12/23885>
- [18] P. Li, S.-H. Lee, and H.-Y. Hsu, "Use of a cold mirror system for citrus fruit identification," in *Computer Science and Automation Engineering (CSAE), 2011 IEEE International Conference on*, vol. 2, June 2011, pp. 376–381.
- [19] P. Li, S. heon Lee, and H.-Y. Hsu, "Study on citrus fruit image data separability by segmentation methods," *Procedia Engineering*, vol. 23, pp. 408 – 416, 2011, {PEEA} 2011. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1877705811053653>
- [20] P. Li, S.-H. Lee, and H.-Y. Hsu, "Study on citrus fruit image using fisher linear discriminant analysis," in *Computer Science and Automation Engineering (CSAE), 2011 IEEE International Conference on*, vol. 4, June 2011, pp. 175–180.
- [21] —, "Fusion on citrus image data from cold mirror acquisition system," *Int. J. Comput. Vis. Image Process.*, vol. 2, no. 4, pp. 11–24, Oct. 2012. [Online]. Available: <http://dx.doi.org/10.4018/ijcvip.2012100102>

- [22] M. Reis, R. Morais, E. Peres, C. Pereira, O. Contente, S. Soares, A. Valente, J. Baptista, P. Ferreira, and J. B. Cruz, "Automatic detection of bunches of grapes in natural environment from color images," *Journal of Applied Logic*, vol. 10, no. 4, pp. 285 – 290, 2012, selected papers from the 6th International Conference on Soft Computing Models in Industrial and Environmental Applications UTAD - Vinhas do Douro. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1570868312000535>
- [23] H. OKAMOTO and W. S. LEE, "Machine vision for green citrus detection in tree images," *Environmental Control in Biology*, vol. 48, no. 2, pp. 93–99, 2010.
- [24] F. Kurtulmus, W. S. Lee, and A. Vardar, "Green citrus detection using 'eigenfruit', color and circular gabor texture features under natural outdoor conditions," *Computers and Electronics in Agriculture*, vol. 78, no. 2, pp. 140 – 149, 2011. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0168169911001475>
- [25] F. Kurtulmus, W. S. Lee, and A. Vardar, "Immature peach detection in colour images acquired in natural illumination conditions using statistical classifiers and neural network," *Precision Agriculture*, vol. 15, no. 1, pp. 57–79, 2014. [Online]. Available: <http://dx.doi.org/10.1007/s11119-013-9323-8>
- [26] J. Lu, N. Sang, Y. Hu, and H. Fu, "Detecting citrus fruits with highlight on tree based on fusion of multi-map," *Optik - International Journal for Light and Electron Optics*, vol. 125, no. 8, pp. 1903 – 1907, 2014. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0030402613014174>
- [27] J. Yin, H. Mao, and Y. Xie, "Segmentation methods of fruit image and comparative experiments," in *Computer Science and Software Engineering, 2008 International Conference on*, vol. 1, Dec 2008, pp. 1098–1102.
- [28] J. Wachs, H. Stern, T. Burks, and V. Alchanatis, "Apple detection in natural tree canopies from multimodal images," in *Joint International Agricultural Conference, 2009*, pp. 293–302.
- [29] P. Viola and M. J. Jones, "Robust real-time face detection," *Int. J. Comput. Vision*, vol. 57, no. 2, pp. 137–154, May 2004. [Online]. Available: <http://dx.doi.org/10.1023/B:VISI.0000013087.49260.fb>
- [30] J. Zhao, J. Tow, and J. Katupitiya, "On-tree fruit recognition using texture properties and color data," in *Intelligent Robots and Systems, 2005. (IROS 2005). 2005 IEEE/RSJ International Conference on*, Aug 2005, pp. 263–268.
- [31] W. Ji, D. Zhao, F. Cheng, B. Xu, Y. Zhang, and J. Wang, "Automatic recognition vision system guided for apple harvesting robot," *Computers & Electrical Engineering*, vol. 38, no. 5, pp. 1186 – 1195, 2012, special issue on Recent Advances in Security and Privacy in Distributed Communications and Image processing. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0045790611001819>

- [32] S. Chaivivatrakul and M. N. Dailey, "Texture-based fruit detection," *Precision Agriculture*, vol. 15, no. 6, pp. 662–683, 2014.
- [33] C. Hung, J. Nieto, Z. Taylor, J. Underwood, and S. Sukkarieh, "Orchard fruit segmentation using multi-spectral feature learning," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Nov 2013, pp. 5314–5320.
- [34] I. Sa, C. McCool, C. Lehnert, and T. Perez, "On visual detection of highly-occluded objects for harvesting automation in horticulture," in *ICRA 2015 : IEEE International Conference on Robotics and Automation*. Seattle, Washington: ICRA, 2015. [Online]. Available: <http://eprints.qut.edu.au/90394/>
- [35] C. McCool, I. Sa, F. Dayoub, C. Lehnert, T. Perez, and B. Upcroft, "Visual detection of occluded crop: For automated harvesting," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, May 2016, pp. 2506–2512.
- [36] JAI. (2017) Camera jai ad-080ge. [Online]. Available: <http://www.jai.com/en/products/ad-080ge>
- [37] J. Hemming, J. Ruizendaal, J. W. Hofstee, and E. J. van Henten, "Fruit detectability analysis for different camera positions in sweet-pepper," *Sensors*, vol. 14, no. 4, pp. 6032–6044, 2014.
- [38] K. Yamamoto, W. Guo, Y. Yoshioka, and S. Ninomiya, "On plant detection of intact tomato fruits using image analysis and machine learning methods," *Sensors*, vol. 14, pp. 12191–12206, 2014.
- [39] D. Pelleg and A. W. Moore, "X-means: Extending k-means with efficient estimation of the number of clusters," in *Proceedings of the Seventeenth International Conference on Machine Learning*, ser. ICML '00. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2000, pp. 727–734. [Online]. Available: <http://dl.acm.org/citation.cfm?id=645529.657808>
- [40] G. Feng, C. Qixin, and N. Masateru, "Fruit detachment and classification method for strawberry harvesting robot," *International Journal of Advanced Robotic Systems*, vol. 5, no. 1, p. 4, 2008. [Online]. Available: <http://dx.doi.org/10.5772/5662>
- [41] Y.-I. Ohta, T. Kanade, and T. Sakai, "Color information for region segmentation," *Computer Graphics and Image Processing*, vol. 13, pp. 222–241, 1980.
- [42] F. Qingchun, C. Wei, Z. Jianjun, and W. Xiu, "Design of structured-light vision system for tomato harvesting robot," *Int J Agric & Biol Eng*, vol. 7, no. 2, pp. 19–26, 2014, laser scanning.
- [43] C. Aloisio, R. K. Mishra, C. Y. Chang, and J. English, "Next generation image guided citrus fruit picker," in *2012 IEEE International Conference on Technologies for Practical Robot Applications (TePRA)*, April 2012, pp. 37–41.
- [44] S. Mehta and T. Burks, "Vision-based control of robotic manipulator for citrus harvesting," *Computers and Electronics in Agriculture*, vol. 102, pp. 146 – 158, 2014. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0168169914000052>

- [45] A. J. Scarfe, R. C. Flemmer, H. H. Bakker, and C. L. Flemmer, "Development of an autonomous kiwifruit picking robot," in *2009 4th International Conference on Autonomous Robots and Agents*, Feb 2009, pp. 380–384.
- [46] A. J. Scarfe, "Development of an autonomous kiwifruit harvester," Ph.D. dissertation, Massey University, 2012.
- [47] N. Irie, N. Taguchi, T. Horie, and T. Ishimatsu, "Development of asparagus harvester coordinated with 3-d vision sensor," *Journal of Robotics and Mechatronics*, vol. 21, pp. 583–589, 2009.
- [48] J. Baeten, K. Donné, S. Boedrij, W. Beckers, and E. Claesen, "Autonomous Fruit Picking Machine: A Robotic Apple Harvester," in *6th International Conference on Field and Service Robotics - FSR 2007*, 2007, pp. 1–9. [Online]. Available: <https://hal.inria.fr/inria-00194739>
- [49] D. y. Kong, D. a. Zhao, Y. Zhang, J. j. Wang, and H. x. Zhang, "Research of apple harvesting robot based on least square support vector machine," in *2010 International Conference on Electrical and Control Engineering*, June 2010, pp. 1590–1593.
- [50] J. Lv, D.-A. Zhao, W. Ji, Y. Chen, and H. Shen, "Design and research on vision system of apple harvesting robot," in *Intelligent Human-Machine Systems and Cybernetics (IHMSC), 2011 International Conference on*, vol. 1, Aug 2011, pp. 177–180.
- [51] D. M. Bulanon and T. Kataoka, "Fruit detection system and an end effector for robotic harvesting of fuji apples," *CIGR Journal*, vol. 12, no. 1, pp. 203–210, Mar 2010.
- [52] T. T. Nguyen, K. Vandevoorde, N. Wouters, E. Kayacan, J. G. D. Baerdemaeker, and W. Saeys, "Detection of red and bicoloured apples on tree with an rgb-d camera," *Biosystems Engineering*, vol. 146, pp. 33 – 44, 2016, special Issue: Advances in Robotic Agriculture for Crops. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1537511016000088>
- [53] D. Font, T. Pallejà, M. Tresanchez, D. Runcan, J. Moreno, D. Martínez, M. Teixidó, and J. Palacín, "A proposal for automatic fruit harvesting by combining a low cost stereovision camera and a robotic arm," *Sensors*, vol. 14, pp. 11557–11579, 2014.
- [54] M. Teixidó, M., D. Font, T. Pallejà, T. , M. Tresanchez, M. Nogués, and J. Palacín, "An embedded real-time red peach detection system based on an ov7670 camera, arm cortex-m4 processor and 3d look-up tables," *Sensors*, vol. 12, no. 10, pp. 14129–14143, 2012. [Online]. Available: <http://www.mdpi.com/1424-8220/12/10/14129>
- [55] K. Tanigaki, T. Fujiura, A. Akase, and J. Imagawa, "Cherry-harvesting robot," *Computers and Electronics in Agriculture*, vol. 63, no. 1, pp. 65 – 72, 2008, special issue on bio-robotics. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0168169908000458>
- [56] S. Amatya, M. Karkee, A. Gongal, Q. Zhang, and M. D. Whiting, "Detection of cherry tree branches with full foliage in planar architecture for automated sweet-cherry harvesting," *Biosystems Engineering*, vol. 146, pp. 3 – 15, 2016, special Issue: Advances in Robotic Agriculture

for Crops. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1537511015001683>

[57] A. Moghimi, M. Aghkhani, M. Golzarian, A. Rohani, and C. Yang, "A robo-vision algorithm for automatic harvesting of green bell pepper," in *American Society of Agricultural and Biological Engineers Annual International Meeting 2015*, vol. 4. United States: American Society of Agricultural and Biological Engineers, 2015, pp. 3185–3193.

[58] C. Lehnert, I. Sa, C. McCool, B. Upcroft, and T. Perez, "Sweet pepper pose detection and grasping for automated crop harvesting," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, May 2016, pp. 2428–2434.

[59] E. Vitzrabin and Y. Edan, "Adaptive thresholding with fusion using a {RGBD} sensor for red sweet-pepper detection," *Biosystems Engineering*, vol. 146, pp. 45 – 56, 2016, special Issue: Advances in Robotic Agriculture for Crops. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1537511015001828>

[60] C. Lehnert, A. English, C. McCool, A. W. Tow, and T. Perez, "Autonomous sweet pepper harvesting for protected cropping systems," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 872–879, April 2017.

[61] N. Kondo, K. Yamamoto, H. Shimizu, K. Yata, M. Kurita, T. Shiigi, M. Monta, and T. Nishizu, "A machine vision system for tomato cluster harvesting robot," *Engineering in Agriculture, Environment and Food*, vol. 2, no. 2, pp. 60 – 65, 2009. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1881836609800177>

[62] H. Liu, P. Li, C. Saunders, and S. Lee, "Development of a green plant image segmentation method of machine vision system for no-tillage fallow weed detection," in *2013 Society for Engineering in Agriculture Conference : innovative agricultural technologies for a sustainable future*, 2013, pp. 95–108.

[63] X.-E. Pantazi, D. Moshou, and C. Bravo, "Active learning system for weed species recognition based on hyperspectral sensing," *Biosystems Engineering*, vol. 146, pp. 193 – 202, 2016, special Issue: Advances in Robotic Agriculture for Crops. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1537511016000143>

[64] R. Berenstein, O. B. Shahr, A. Shapiro, and Y. Edan, "Grape clusters and foliage detection algorithms for autonomous selective vineyard sprayer," *Intell. Serv. Robot.*, vol. 3, no. 4, pp. 233–243, Oct. 2010. [Online]. Available: <http://dx.doi.org/10.1007/s11370-010-0078-z>

[65] R. Oberti, M. Marchi, P. Tirelli, A. Calcante, M. Iriti, E. Tona, M. Hocevar, J. Baur, J. Pfaff, C. Schutz, and H. Ulbrich, "Selective spraying of grapevines for disease control using a modular agricultural robot," *Biosystems Engineering*, vol. 146, pp. 203 – 215, 2016, special Issue: Advances in Robotic Agriculture for Crops. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1537511015001865>

- [66] S. Cubero, N. Aleixos, F. Albert, A. Torregrosa, C. Ortiz, O. Garc a-Navarrete, and J. Blasco, "EnglishOptimised computer vision system for automatic pre-grading of citrus fruit in the field using a mobile platform," *EnglishPrecision Agriculture*, vol. 15, no. 1, pp. 80–94, 2014. [Online]. Available: <http://dx.doi.org/10.1007/s11119-013-9324-7>
- [67] S. Sankaran, J. M. Maja, S. Buchanon, and R. Ehsani, "Huanglongbing (citrus greening) detection using visible, near infrared and thermal imaging techniques," *Sensors*, vol. 13, pp. 2117–2130, 2013.
- [68] D. Dey, L. Mummert, and R. Sukthankar, "Classification of plant structures from uncalibrated image sequences," in *2012 IEEE Workshop on the Applications of Computer Vision (WACV)*, Jan 2012, pp. 329–336.
- [69] S. T. Namin and L. Petersson, "Classification of materials in natural scenes using multi-spectral images," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Oct 2012, pp. 1393–1398.
- [70] C. Bac, J. Hemming, and E. van Henten, "Robust pixel-based classification of obstacles for robotic harvesting of sweet-pepper," *Computers and Electronics in Agriculture*, vol. 96, pp. 148 – 162, 2013. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0168169913001099>
- [71] P. Li, S. heon Lee, and H.-Y. Hsu, "Review on fruit harvesting method for potential use of automatic fruit harvesting systems," *Procedia Engineering*, vol. 23, pp. 351 – 366, 2011, {PEEA} 2011. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1877705811053574>
- [72] K. Kapach, E. Barnea, R. Mairon, Y. Edan, and O. Ben-Shahar, "Computer vision for fruit harvesting robots &#150; state of the art and challenges ahead," *Int. J. Comput. Vision Robot.*, vol. 3, no. 1/2, pp. 4–34, Apr. 2012. [Online]. Available: <http://dx.doi.org/10.1504/IJCVR.2012.046419>
- [73] C. W. Bac, E. J. van Henten, J. Hemming, and Y. Edan, "Harvesting robots for high-value crops: State-of-the-art review and challenges ahead," *Journal of Field Robotics*, vol. 31, no. 6, pp. 888–911, 2014. [Online]. Available: <http://dx.doi.org/10.1002/rob.21525>
- [74] A. Gongal, S. Amatya, M. Karkee, Q. Zhang, and K. Lewis, "Sensors and systems for fruit detection and localization: A review," *Computers and Electronics in Agriculture*, vol. 116, pp. 8 – 19, 2015. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0168169915001581>
- [75] (2014) Crops - clever robots for crops. [Online]. Available: <http://www.crops-robots.eu/>
- [76] Sweeper. [Online]. Available: <http://www.sweeper-robot.eu/>
- [77] I. Sa, Z. Ge, F. Dayoub, B. Ucroft, T. Perez, and C. McCool, "Deepfruits: A fruit detection system using deep neural networks," *Sensors*, vol. 16, p. 1222, 2016. [Online]. Available: <http://www.mdpi.com/1424-8220/16/8/1222>

- [78] S. Bargoti and J. P. Underwood, "Deep fruit detection in orchards," *CoRR*, vol. abs/1610.03677, 2016. [Online]. Available: <http://arxiv.org/abs/1610.03677>
- [79] M. Stein, S. Bargoti, and J. P. Underwood, "Image based mango fruit detection, localisation and yield estimation using multiple view geometry," *Sensors*, vol. 16, no. 11, p. 1915, 2016. [Online]. Available: <http://dx.doi.org/10.3390/s16111915>
- [80] C. McCool, T. Perez, and B. Upcroft, "Mixtures of lightweight deep convolutional neural networks: Applied to agricultural robotics," *IEEE Robotics and Automation Letters*, vol. 2, no. 3, pp. 1344–1351, July 2017.
- [81] J. P. Wachs, H. I. Stern, T. Burks, and V. Alchanatis, "Low and high-level visual feature-based apple detection from multi-modal images," *Precision Agriculture*, vol. 11, no. 6, pp. 717–735, 2010. [Online]. Available: <http://dx.doi.org/10.1007/s11119-010-9198-x>
- [82] R. Chinchuluun and W. Lee, "Citrus yield mapping system in natural outdoor scenes using the watershed transform," in *ASAE Annual Meeting*, 2006. [Online]. Available: <http://elibrary.asabe.org/abstract.asp?aid=20853&t=5>
- [83] L. Qiang, C. Jianrong, L. Bin, D. Lie, and Z. Yajing, "Identification of fruit and branch in natural scenes for citrus harvesting robot using machine vision and support vector machine," *International Journal of Agricultural and Biological Engineering (IJABE)*, vol. 7, no. 2, pp. 115–121, April 2014. [Online]. Available: <https://ijabe.org/index.php/ijabe/article/view/1215>
- [84] S. Sengupta and W. S. Lee, "Identification and determination of the number of immature green citrus fruit in a canopy under different ambient light conditions," *Biosystems Engineering*, vol. 117, pp. 51 – 61, 2014, image Analysis in Agriculture. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1537511013001141>
- [85] A. Plebe and G. Grasso, "English Localization of spherical fruits for robotic harvesting," *English Machine Vision and Applications*, vol. 13, no. 2, pp. 70–79, 2001. [Online]. Available: <http://dx.doi.org/10.1007/PL00013271>
- [86] W. C. Seng and S. H. Mirisae, "A new method for fruits recognition system," in *2009 International Conference on Electrical Engineering and Informatics*, vol. 01, Aug 2009, pp. 130–134.
- [87] R. Linker, O. Cohen, and A. Naor, "Determination of the number of green apples in {RGB} images recorded in orchards," *Computers and Electronics in Agriculture*, vol. 81, pp. 45 – 57, 2012. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0168169911002638>
- [88] M. Regunathan and W. S. Lee, "Citrus fruit identification and size determination using machine vision and ultrasonic sensors," in *2005 ASAE Annual International Meeting*, 2005.
- [89] S. C. S. Pinto, "Processamento de linguagem natural e extração de conhecimento," Master's thesis, Universidade de Coimbra, 2015.

- [90] R. B. Girshick, "Fast R-CNN," *CoRR*, vol. abs/1504.08083, 2015. [Online]. Available: <http://arxiv.org/abs/1504.08083>
- [91] S. Ren, K. He, R. B. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," *CoRR*, vol. abs/1506.01497, 2016. [Online]. Available: <http://arxiv.org/abs/1506.01497>
- [92] M. H. Nguyen and F. de la Torre, "Optimal feature selection for support vector machines," *Pattern Recognition*, vol. 43, no. 3, pp. 584 – 591, 2010. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0031320309003409>
- [93] J. Bi, K. Bennett, M. Embrechts, C. Breneman, and M. Song, "Dimensionality reduction via sparse support vector machines," *J. Mach. Learn. Res.*, vol. 3, pp. 1229–1243, Mar. 2003. [Online]. Available: <http://dl.acm.org/citation.cfm?id=944919.944971>
- [94] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *J. Mach. Learn. Res.*, vol. 3, pp. 1157–1182, Mar. 2003. [Online]. Available: <http://dl.acm.org/citation.cfm?id=944919.944968>
- [95] M. Kubat and S. Matwin, "Addressing the curse of imbalanced training sets: One-sided selection," in *In Proceedings of the Fourteenth International Conference on Machine Learning*. Morgan Kaufmann, 1997, pp. 179–186.
- [96] H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, no. 9, pp. 1263–1284, Sept 2009.
- [97] C. wei Hsu, C. chung Chang, and C. jen Lin, "A practical guide to support vector classification," 2016. [Online]. Available: <http://www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf>
- [98] A. Silva, "Optimização de métodos de núcleo utilizando algoritmos de enxame," Ph.D. dissertation, Universidade de Évora, Julho 2014. [Online]. Available: <http://hdl.handle.net/10174/12322>
- [99] F. J. B. Pereira, "Estudo das interacções entre evolução e aprendizagem em ambientes de computação evolucionária," Ph.D. dissertation, 2002. [Online]. Available: <http://hdl.handle.net/10316/1744>
- [100] A. R. G. Soares, "Integrated management of residential energy resources: Models, algorithms and application," Ph.D. dissertation, Universidade de Coimbra, 2016. [Online]. Available: <http://hdl.handle.net/10316/31243>
- [101] S.-U. Guan and F. Zhu, "An incremental approach to genetic-algorithms-based classification," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 35, no. 2, pp. 227–239, April 2005.
- [102] M. Couceiro and L. Santos, *The Minerva Cluster User's Guide*, LaCED, 2017. [Online]. Available: <https://www.isec.pt/PT/Instituto/investigacao/minerva/Minerva-UserGuide-v1.0.pdf>

- [103] A. B. Yoo, M. A. Jette, and M. Grondona, *SLURM: Simple Linux Utility for Resource Management*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 44–60. [Online]. Available: [http://dx.doi.org/10.1007/10968987\\_3](http://dx.doi.org/10.1007/10968987_3)
- [104] R. Oliveira, T. Farinha, H. Raposo, and N. Pires, “Augmented reality and the future of maintenance,” in *Proc. of Maintenance Performance Measurement and Management (MPMM) Conference 2014*. Coimbra: Imprensa da Universidade de Coimbra, Sept. 2014, pp. 81–88.
- [105] J. T. Farinha, R. Oliveira, I. Fonseca, and F. M. Barbosa, “The convergence between predictive maintenance and augmented reality to aid renewable energy equipment availability,” *International Journal on Engineering Applications (IREA)*, vol. 2, no. 5, 2014. [Online]. Available: <https://doi.org/10.15866/irea.v2i5.1916>
- [106] P. Viola and M. Jones, “Fast and robust classification using asymmetric adaboost and a detector cascade,” in *Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic*, ser. NIPS’01. Cambridge, MA, USA: MIT Press, 2001, pp. 1311–1318. [Online]. Available: <http://dl.acm.org/citation.cfm?id=2980539.2980709>
- [107] Opencv. [Online]. Available: <http://opencv.org/>
- [108] *Studio encoding parameters of digital television for standard 4:3 and wide-screen 16:9 aspect ratios*, ITU-R Std. BT.601-7, 3 2011. [Online]. Available: [http://www.itu.int/dms\\_pubrec/itu-r/rec/bt/R-REC-BT.601-7-201103-!!PDF-E.pdf](http://www.itu.int/dms_pubrec/itu-r/rec/bt/R-REC-BT.601-7-201103-!!PDF-E.pdf)
- [109] T. Fawcett, “An introduction to roc analysis,” *Pattern Recogn. Lett.*, vol. 27, no. 8, pp. 861–874, Jun. 2006. [Online]. Available: <http://dx.doi.org/10.1016/j.patrec.2005.10.010>
- [110] B. Ozdemir, S. Aksoy, S. Eckert, M. Pesaresi, and D. Ehrlich, “Performance measures for object detection evaluation,” *Pattern Recognition Letters*, vol. 31, no. 10, pp. 1128 – 1137, 2010, pattern Recognition in Remote Sensing. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0167865509002918>
- [111] R. Oliveira, J. T. Farinha, I. Fonseca, and F. M. Barbosa, “Augmented reality system for maintenance of high-voltage systems,” in *Proc. of UPEC*, 2016.
- [112] R. Oliveira, J. André, J. T. Farinha, and J. N. Pires, “Monocular model-based on augmented reality,” in *Poster session presented at the VISion Understanding and Machine intelligence – VISUM*, 2014.
- [113] H. Raposo, J. T. Farinha, R. Oliveira, L. A. Ferreira, and J. André, “Time replacement optimization models for urban transportation buses with indexation to fleet reserve,” in *Proc. of Maintenance Performance Measurement and Management (MPMM) Conference 2014*. Coimbra: Imprensa da Universidade de Coimbra, Sept. 2014, pp. 41–48.
- [114] F. Rodrigues, I. Fonseca, R. Oliveira, and J. T. F. Farinha, “Maintenance management in web asp.net mvc applications,” in *Proc. of Maintenance Performance Measurement and*

*Management (MPMM) Conference 2014*. Coimbra: Imprensa da Universidade de Coimbra, Sept. 2014, pp. 95–101.

[115] J. Ferreira, L. Oliveira, and R. Oliveira, “Real time data collection and processing for aircraft maintenance enhancement (react),” in *Proc. of Maintenance Performance Measurement and Management (MPMM) Conference 2014*. Coimbra: Imprensa da Universidade de Coimbra, Sept. 2014, pp. 11–14.

[116] J. T. Farinha, I. Fonseca, R. Oliveira, and H. Raposo, “Cmms – an integrated view from maintenance management to on-line condition monitoring,” in *Proc. of Maintenance Performance Measurement and Management (MPMM) Conference 2014*. Coimbra: Imprensa da Universidade de Coimbra, Sept. 2014, pp. 137–141.

[117] S. Singh, D. Galar, U. Kumar, R. Oliveira, and T. Farinha, “Augmented reality - technology for effective maintenance in railway sector,” in *Proc. of the 2nd International Conference on Industrial Engineering ICIE 2013*. EXCELLENT PUBLISHING HOUSE, 2013, pp. 727–731.

[118] R. Oliveira, T. Farinha, S. Singh, and D. Galar, “An augmented reality application to support maintenance – is it possible?” in *Proc. Maintenance Performance Measurement and Management (MPMM) 2013*, 2013, pp. 260–271. [Online]. Available: [https://online.unileoben.ac.at/-mu\\_online/voe\\_main2.getVollText?pDocumentNr=142932&pCurrPk=32372](https://online.unileoben.ac.at/-mu_online/voe_main2.getVollText?pDocumentNr=142932&pCurrPk=32372)

[119] J. T. Farinha, R. Oliveira, I. Fonseca, and L. A. Ferreira, “Technical asset management beyond tomorrow,” in *Proc. 2nd World Maintenance Forum*, Lugano, 2013.

[120] R. Oliveira, J. T. Farinha, I. Fonseca, L. A. Ferreira, and F. Armas, “Augmented reality as a new level for maintenance efficiency,” in *Proc. of The 2nd international workshop and congress on eMaintenance*, 2012, pp. 49–55. [Online]. Available: [https://pure.ltu.se/portal/files/41489674/-Proceedings\\_eMaintenance\\_2012.pdf](https://pure.ltu.se/portal/files/41489674/-Proceedings_eMaintenance_2012.pdf)

[121] R. Oliveira, J. T. Farinha, and I. Fonseca, “Realidade aumentada e holografia,” *Manutenção*, vol. 130-131, pp. 58–61, 2016. [Online]. Available: <http://www.revistamanutencao.pt/revista/>

[122] J. T. Farinha and R. Oliveira, “Máquina autónoma de classificação de fruta com colheita automática,” *Agrobótica (Agrotec)*, vol. 3, pp. 8–11, 2015.

[123] R. Oliveira and T. Farinha, “A manutenção de equipamentos de saúde - da pré-história ao futuro,” *Tecno Hospital*, vol. 61, pp. 11–17, Jan/Fev 2013.

[124] R. Oliveira and T. Farinha, “Realidade aumentada – cenários virtuais em hospitais reais,” *Tecno Hospital*, vol. 57, pp. 34–38, Mai/Jun 2013.

[125] J. T. Farinha, R. Oliveira, I. Fonseca, and L. A. Ferreira, “Cmms - para além do amanhã,” in *Proc. 17.º Congresso Ibero-americano de Manutenção; 12.º Congresso Nacional de Manutenção; 1.º Encontro de Manutenção dos Países de Língua Oficial Portuguesa*, 2013.

[126] R. Oliveira and T. Farinha, "Realidade aumentada – ferramenta de apoio à manutenção em equipamento hospitalar," *Tecno Hospital*, vol. 49, pp. 12–16, Jan/Fev 2012.