

FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO



Sistema de Inspeção Visual Automática de Moldes de Pneus Baseado em CAD

Ricardo Ferreira da Silva

Mestrado Integrado em Engenharia Eletrotécnica e de Computadores

Orientador: Professor Doutor Paulo José Cerqueira Gomes da Costa

Coorientador: Doutor Andry Maykol Gomes Pinto

Outubro de 2016

Resumo

O controlo de qualidade enquanto conjunto de técnicas e processos que visam assegurar que um produto apresenta as características que lhe são expectáveis, surge nos mais variados tipos de produção. Este é fortemente sustentado em indústrias que prezam a qualidade dos seus produtos, como por exemplo, a indústria aeronáutica ou automóvel. Mais concretamente, a indústria de pneus automóveis pode mesmo incorrer em ações judiciais sempre que se verifique que um defeito de fabrico foi a causa de um acidente. Nesse contexto, a inspeção da conformidade das características de um pneu em relação às suas especificações prévias de produção é a principal garantia que visa assegurar a qualidade do mesmo.

Verifica-se que há, efetivamente, um esforço crescente na investigação de técnicas de controlo de qualidade, objetivando-se a melhoria e inovação dos processos de inspeção de forma a garantir a qualidade extrema dos produtos. Nesta área de estudo, a inspeção de defeitos utilizando técnicas de visão artificial é um assunto atual e largamente estudado.

A presente dissertação incide no estudo e implementação do *software* integrante de um sistema de inspeção visual automática tendo em vista a sua aplicação num caso de estudo real proposto pela empresa Continental Mabor. Apresenta-se, assim, um sistema capaz de efetuar a inspeção dos moldes usados para inserir a informação na parede dos pneus durante a sua produção. O *software* tem como entradas o modelo CAD do molde a inspecionar, bem como uma imagem resultante da aquisição desse mesmo molde. Com efeito, é proposta uma arquitetura de inspeção de maneira a verificar se os moldes estão em conformidade com o seu modelo.

O *software* em estudo é então composto por várias etapas. Primeiramente, apresenta-se um método automático para gerar uma imagem a partir do referido modelo CAD do molde. Esta imagem é um elemento imprescindível ao funcionamento do sistema, na medida em que serve como elemento de comparação, ou seja, a informação que nela aparece pressupõe-se correta e deve constar da imagem do molde real. Seguidamente, descreve-se o processo de uniformização relativamente à forma e conteúdo das imagens gerada e adquirida, tornando-as mais facilmente comparáveis. Dada a complexidade do molde, em virtude de este possuir diversos elementos de natureza distinta, nem toda a informação pode ser interpretada de igual forma. Além disso, há certos campos de informação específicos de cada produção que são aplicados ao molde em uso e que não constam do seu modelo CAD, sendo assim impossível a sua comparação apenas com as referidas imagens. Por estas razões, é ainda proposto um método baseado na leitura de elementos alfanuméricos nas imagens de forma a interpretar toda a informação nelas presente.

Finalmente, fruto do estudo e validação de algoritmos de comparação de imagens e leitura de elementos alfanuméricos, o *software* apresenta como saídas a informação lida e a identificação das não conformidades do molde em relação ao seu projeto em CAD.

Abstract

Quality Control is present in various industries and can be defined as the set of techniques and processes that ensure a product is built and behaves as expected. This is of the utmost importance in industries where the quality of the products needs to be guaranteed, like the aeronautics and automotive industries. To be more specific, the car tire industry can suffer from severe lawsuits if a defective product is found to be the cause of an accident. Therefore, the control of the characteristics of a tire in comparison with its production specifications is the main technique to ensure its quality.

There has been a growing effort in the research and study of quality control techniques. The main goal of such research is the improvement of inspection processes as to ensure the highest possible quality of the products. Within this industry, quality control using computer vision systems is a current and widely studied subject.

The following thesis focuses on the research and implementation of a software to integrate an automated vision inspection system to be applied in a real case study as proposed by Continental Mabor company. Thus, a system capable of performing the inspection of the moulds, used to enter information in the tire wall during its production, is presented. The software has as inputs the CAD model and a resulting image acquisition of the mould under inspection. An inspection architecture is proposed in order to verify that the moulds are in accordance with their model.

The software in study is composed by several stages. Firstly, is presented an automatic method for an image generation from the CAD model of the mould. This is an essential element for the function of the system, since it works as a comparison point, that is, the information that appears on it is assumed to be correct and must appear on the acquired image. Next, is described the standardization process of the form and content of the generated and acquired images, turning them more easily comparable. Given the complexity of the mould, by means of having many elements of different nature, not all the information can be interpreted in the same way. In addition, there are certain specific information fields that are modified in the mould for each production and so are not included in its CAD model, being impossible to compare using only the two images. For these reasons, it is further proposed a method based on optical character recognition on the images in order to interpret all the information.

Finally, as a result of the study and validation of image comparison and optical character recognition algorithms, the software displays as outputs the information read and the identification of non-compliances between the mould and its CAD model.

Agradecimentos

Ainda que qualquer dissertação preveja um trabalho de investigação individual, há contributos que merecem ser verdadeiramente distinguidos. Além disso, pela marca de final de um ciclo que este documento inevitavelmente representa para mim enquanto estudante, encontro a oportunidade de reconhecer o apoio de algumas pessoas que se revelaram autênticos modelos de coragem e persistência durante todo o meu percurso académico. Assim, ainda que em poucas palavras, quero deixar a minha gratidão às pessoas e instituições cujo envolvimento foi essencial.

Em primeiro lugar à minha mãe, Silvia, ao meu irmão, Daniel, ao meu pai, Leonel, ao meu avô, Fernando, e demais família, por toda a preocupação, apoio e presença desde o primeiro dia do meu percurso académico.

Ao meu orientador, o Professor Doutor Paulo Costa e coorientador, o Doutor Andry Pinto, por toda a simpatia, disponibilidade e sabedoria partilhada.

À Continental Mabor, nomeadamente, à Doutora Eduarda Silva, ao Eng. Eduardo Moreira e ao Doutor Marcos Ferreira, pelo envolvimento no projeto e, ainda, pela ajuda e atenção.

Ao INESC TEC, nomeadamente, ao Professor Doutor António Paulo Moreira, ao Eng. Carlos Costa, ao Eng. Héber Sobreira e ao Doutor Luís Rocha, pelo acolhimento e por toda a ajuda prestada.

Ao Fernando, ao Ivo, à Marta e ao Patrick por toda a ajuda, boa disposição e companhia no espaço de trabalho.

À Andreia, à Catarina, ao João Costa, ao João Sá, ao Nuno e à Sara, muito resumidamente, por toda a ajuda, amizade, paciência e presença. Cada um à sua maneira, ocupou um papel que muito valorizo durante o meu percurso académico.

Aos *Fraquitos*, pela hospitalidade de cada reencontro e, ainda, por todo o companheirismo e amizade.

À Orquestra Clássica da FEUP que, além de um projeto fantástico no qual me envolvi, se revelou como uma parte muito importante na minha formação pessoal.

Por último, a todos aqueles que, embora não mencionados, de alguma maneira deram o seu contributo.

Como disse Sir Isaac Newton: "Se vi mais longe, foi por estar de pé sobre ombros de gigantes." A todos esses "*gigantes*", o meu muito obrigado!

Ricardo Silva

*“If you really believe in what you’re doing, work hard, take nothing personally
and if something blocks one route, find another.
Never give up.”*

Laurie Notaro

Conteúdo

1	Introdução	1
1.1	Motivação	1
1.2	Objetivos	3
1.3	Estrutura da dissertação	4
2	Contexto	5
2.1	A Continental Mabor	5
2.2	Produção de pneus	6
2.3	Moldes de pneus	7
2.4	Informação na parede de um pneu	9
2.4.1	Logótipos de identificação	9
2.4.2	Dimensões e características	9
2.4.3	Informação de fabrico	10
2.4.4	País de fabrico	12
2.4.5	Aviso de segurança	12
2.4.6	Marcações de Inverno	12
2.4.7	Constituição do piso e da parede	12
2.4.8	Desgaste do piso, tração e temperatura	13
2.4.9	Pressão e carga	14
2.4.10	Números identificativos do pneu	14
2.4.11	Considerações finais	14
3	Revisão Bibliográfica	17
3.1	Sistemas de inspeção visual automática	17
3.1.1	Sistemas de iluminação	18
3.1.2	<i>Software</i> de aplicação	24
3.2	Formato CAD e sua integração numa aplicação de inspeção visual	29
4	O Sistema de Inspeção	33
4.1	Arquitetura do sistema	33
4.2	O <i>software</i> de aplicação	35
5	Geração de Imagem a partir de Modelo CAD	39
5.1	O formato DXF	40
5.2	Interpretação de um modelo CAD em DXF	41
5.3	Imagem gerada	42
5.3.1	Vetor extrusão e conversão de coordenadas	43
5.3.2	Representação das entidades	44

5.4	Resultados	55
5.5	Conclusões	60
6	Transformação da Imagem Gerada Polar para Cartesiana	61
6.1	Definição da região de interesse	61
6.2	Determinação da espessura da secção anelar	62
6.3	Transformação do referencial de imagem	64
6.4	Resultados	66
6.5	Conclusões	69
7	Sincronização das Imagens Gerada e Adquirida	71
7.1	Definição das zonas de sincronismo	72
7.1.1	<i>Template Matching</i>	72
7.1.2	<i>Feature Matching</i>	75
7.1.3	Definição da região de interesse	77
7.2	Escalamento das imagens	79
7.3	Seleção da região de interesse da imagem adquirida	80
7.4	Rearranjo da imagem gerada	81
7.5	Resultados	82
7.6	Conclusões	85
8	Comparação das Imagens Gerada e Adquirida	87
8.1	Informação Lida	88
8.2	Informação Comparável	94
8.3	Resultados	99
8.4	Conclusões	103
9	Conclusões e Trabalho Futuro	105
9.1	Conclusões	105
9.2	Trabalho Futuro	106
A	Índices de Carga e Velocidade de um pneu	107
A.1	Valores para o Índice de Velocidade (SI) de um pneu	107
A.2	Valores tabelados para o Índice de Carga (LI) de um pneu	108
B	Sistemas de Aquisição de Imagem e de Iluminação	109
B.1	Princípio de funcionamento da câmara	109
B.2	Sistemas de aquisição e iluminação	111
B.2.1	Constituição do sistema	111
B.2.2	Estrutura de funcionamento	112
B.3	Imagem adquirida	114
C	Informação extraída do modelo CAD para teste da biblioteca	117
C.1	Linha 1	117
C.2	Circunferência 2	118
C.3	Linha 3	118
C.4	Linha 4	119
C.5	Circunferência 5	119
C.6	Circunferência 6	120
C.7	Arco de circunferência 7	120

D	Significado dos atributos comuns às entidades do formato DXF	121
E	Atributos de entidades do formato DXF	123
E.1	<i>ELLIPSE</i>	123
E.2	<i>SPLINE</i>	124
E.3	<i>POINT</i>	125
E.4	<i>INSERT</i>	125
	Referências	127

Lista de Figuras

2.1	Logótipo da Continental, em cima, e vista aérea das instalações da Continental Mabor, em baixo.	6
2.2	Exemplo de moldes após e antes de se completar a assemblagem dos seus componentes, das esquerda para a direita, respetivamente.	7
2.3	Molde da parede de um pneu.	8
2.4	Peças de encaixe, zona de encaixe e zona após encaixe de uma peça no molde, da esquerda para a direita, respetivamente.	8
2.5	Exemplo de um defeito num molde.	8
2.6	Exemplo da vista lateral de um pneu.	9
2.7	Exemplo de logótipos identificativos da empresa fabricante e do modelo do pneu presentes na parede do mesmo.	10
2.8	Exemplo de informação de dimensão e características de um pneu presentes na parede do mesmo.	10
2.9	Representação das dimensões de um pneu: A: largura; B: altura; e C: diâmetro interior.	11
2.10	Informação relativa ao tipo de construção e à ausência de câmara-de-ar na parede de um pneu.	11
2.11	Exemplo de código identificativo de fabrico.	11
2.12	Exemplo de informação relativa ao país de fabrico na parede de um pneu.	12
2.13	Exemplo de um excerto de marcação de aviso de segurança na parede de um pneu.	12
2.14	Exemplo de marcação de Inverno na parede de um pneu.	13
2.15	Exemplo de informação de constituição do piso e da parede de um pneu.	13
2.16	Exemplo de informação dos valores de desgaste de piso, tração e temperatura na parede de um pneu.	14
2.17	Exemplo de informação de valores de pressão e carga na parede de um pneu.	14
2.18	Exemplo de informação referente aos números identificativos de um pneu na sua parede.	15
3.1	Exemplo de sistema de iluminação difusa em que se utiliza uma superfície refletora para difundir a luz. [1]	19
3.2	Exemplo de um sistema de iluminação direta: um painel de LEDs com luz focalizada. [1]	19
3.3	Exemplo de sistema de iluminação frontal. [1]	20
3.4	Exemplo de sistema de iluminação em contraluz. [1]	20
3.5	Exemplo de sistema de iluminação em campo claro. [1]	20
3.6	Exemplo de sistema de iluminação em campo escuro. [1]	21
3.7	Esquema de distinção das técnicas de iluminação dependentes das propriedades de propagação da luz.	21

3.8	Exemplo de aplicação de um sistema de iluminação em campo claro, frontal: e difusa, à esquerda; e direta, à direita. [1]	22
3.9	Exemplo de aplicação de um sistema de iluminação frontal, direta e em campo escuro. À esquerda, o realce de letras e à direita, o realce de uma impressão em <i>Braille</i> . [1]	23
3.10	Exemplos de aplicação de um sistema de iluminação difusa, em contraluz e em campo claro. [1]	23
3.11	Exemplificação da diferença entre um sistema de iluminação em contraluz e em campo claro com luz difusa, à esquerda, e luz direta, à direita. [1]	23
3.12	Comparação entre a utilização de um sistema de iluminação difusa, em contraluz e em campo claro (à esquerda) e de iluminação difusa, em contraluz e em campo escuro (à direita), para evidenciar o contorno de um copo de vidro.[1]	24
3.13	Iluminação de uma PCB com luz branca, vermelha, verde e azul, da esquerda para a direita, respetivamente, evidenciando-se o efeito da cor da luz nas imagens adquiridas. [1]	24
3.14	Diagrama de blocos do algoritmo de OCR proposto em [2].	25
3.15	Exemplo de uma imagem binarizada. [2]	25
3.16	Resultado da transformação geométrica para endireitar a imagem. [2]	26
3.17	Resultado da operação de erosão da imagem. [2]	26
3.18	Identificação e contorno de cada caractere. [2]	26
3.19	Imagem resultante da binarização, à esquerda, e imagem resultante da deteção de orlas do lado direito, à direita. [3]	27
3.20	Diagrama do algoritmo de OCR, à esquerda, e a seleção dos 23 pontos para comparação da imagem, à direita. [4]	27
3.21	Exemplo de aplicação do método da janela deslizante. A verde estão identificadas duas posições possíveis do <i>template</i> sobre a imagem. A vermelho está identificada a posição para a qual foi encontrada uma correspondência do <i>template</i> . [5]	28
3.22	Exemplos de <i>Haar-like features</i> . [5]	28
3.23	Exemplos de imagens <i>templates</i> : <i>template</i> original, à esquerda, e <i>template</i> de baixa resolução para redução da ROI, à direita. [6]	29
3.24	Diagrama de blocos do sistema proposto em [7].	30
3.25	Diagrama do sistema proposto em [8].	31
3.26	Comparação entre imagens adquiridas do topo de um componente (em cima) e as geradas a partir do seu modelo CAD (em baixo) em diversas fases da sua produção. [9]	31
3.27	Exemplos de imagens geradas a partir de modelos CAD (em cima) usados para comparação com as imagens adquiridas das peças (em baixo). [10]	32
4.1	Arquitetura do sistema de inspeção.	35
4.2	Logótipo do <i>software</i> de aplicação - <i>acbs</i>	36
4.3	Esquema de funcionamento do <i>acbs</i>	37
5.1	Ilustração do modelo CAD desenvolvido para teste da biblioteca com identificação das entidades e referencial de desenho.	41
5.2	Exemplo de representação das entidades do tipo <i>CIRCLE</i> do modelo CAD de um molde.	45
5.3	Exemplo de representação das entidades do tipo <i>CIRCLE</i> , a preto , e <i>LINE</i> , a verde , do modelo CAD de um molde.	46
5.4	Exemplo de representação de um arco de circunferência.	47

5.5	Ilustração do efeito da conversão de referencial tendo em conta um vetor de extrusão igual a (0, 0, -1) na representação de um arco de circunferência.	48
5.6	Exemplo da inversão dos pontos inicial e final e consequente representação do arco da figura 5.4.	49
5.7	Exemplo de representação das entidades do tipo <i>CIRCLE</i> , a preto , <i>LINE</i> , a verde , e <i>ARC</i> , a azul , do modelo CAD de um molde.	50
5.8	Exemplo de representação de uma elipse.	51
5.9	Representação dos ângulos utilizados na decisão do valor de largura e altura de cada elipse.	52
5.10	Exemplo de representação das entidades do tipo <i>CIRCLE</i> , a preto , <i>LINE</i> , a verde , <i>ARC</i> , a azul e <i>ELLIPSE</i> , a vermelho , do modelo CAD de um molde.	53
5.11	Exemplo de <i>SPLINEs</i> resultantes de uma interpolação de primeira ordem, pelo tracejado, e de terceira ordem, pela linha contínua.	53
5.12	Exemplo de representação das entidades do tipo <i>CIRCLE</i> , a preto , <i>LINE</i> , a verde , <i>ARC</i> , a azul , <i>ELLIPSE</i> , a vermelho , e <i>SPLINE</i> , a cor de laranja , do modelo CAD de um molde.	54
5.13	Resultado da imagem gerada de um modelo CAD que contém 86 entidades <i>CIRCLE</i> , a preto , 3363 <i>LINE</i> , a verde , 22 <i>ARC</i> , a azul , 12 <i>ELLIPSE</i> , a vermelho , 1343 <i>SPLINE</i> , a cor de laranja , e 49 <i>INSERT</i> contendo 3154 <i>LINE</i> , a roxo , do modelo CAD de um molde.	55
5.14	<i>Print screen</i> do AutoCAD para comprovar a correta geração de imagem do modelo da figura 5.13.	56
5.15	Resultado da imagem gerada de um modelo CAD que contém 60 entidades <i>CIRCLE</i> , a preto , 10372 <i>LINE</i> , a verde , e 708 <i>ARC</i> , a azul	57
5.16	Resultado da imagem gerada de um modelo CAD que contém 81 entidades <i>CIRCLE</i> , a preto , 103 <i>LINE</i> , a verde , 14 <i>ARC</i> , a azul , 24 <i>ELLIPSE</i> , a vermelho , e 1242 <i>SPLINE</i> , a cor de laranja	58
5.17	Resultado da imagem gerada de um modelo CAD que contém 42 entidades <i>CIRCLE</i> , a preto , 1523 <i>LINE</i> , a verde , 36 <i>ARC</i> , a azul , 20 <i>ELLIPSE</i> , a vermelho , e 1402 <i>SPLINE</i> , a cor de laranja	59
6.1	Exemplo de representação da <i>bounding box</i> da ROI da imagem gerada do modelo CAD de um molde.	62
6.2	Identificação da coluna de pixels e direção de iteração da imagem para determinação do raio interior da secção anelar.	63
6.3	Representação dos limites internos e externos da secção anelar, paralelos aos lados da imagem.	63
6.4	Ilustração da definição da imagem gerada através de um sistema de coordenadas polares.	64
6.5	Ilustração da definição do passo angular de varrimento.	65
6.6	Atribuição do valor de intensidade a um pixel pelo método do vizinho mais próximo.	66
6.7	Atribuição do valor de intensidade a um pixel pelo método bilinear.	66
6.8	Diferença entre os métodos de atribuição de intensidade a cada pixel num elemento textual: em cima, vizinho mais próximo e, em baixo, bilinear.	67
6.9	Diferença entre os métodos de atribuição de intensidade a cada pixel num logótipo: em cima, vizinho mais próximo e, em baixo, bilinear.	67
6.10	Excerto das imagens cartesianas resultantes dos modelos CAD das figuras 5.13, 5.15, 5.16 e 5.17, respetivamente de cima para baixo.	68
6.11	Efeito do passo angular na imagem retangular resultante.	69

7.1	Imagens <i>template</i> utilizadas para as imagens gerada e adquirida, de cima para baixo, e para o campo da informação de fabrico e do aviso de segurança, da esquerda para a direita.	73
7.2	Localização, a azul, dos <i>templates</i> numa imagem gerada.	74
7.3	Localizações candidatas, a vermelho, dos <i>templates</i> numa imagem adquirida. . .	75
7.4	<i>Features</i> detetadas em cada <i>template</i> , à esquerda, e na zona candidata correta respectiva, à direita, com descritores <i>AKAZE</i>	75
7.5	Aplicação da técnica de <i>feature matching</i> às imagens da figura 7.4.	76
7.6	Identificação, a azul, da zona validada da localização dos <i>templates</i> numa imagem adquirida com a técnica de <i>feature matching</i>	76
7.7	Identificação, a verde, da localização dos <i>templates</i> após definição da região de interesse.	77
7.8	Passos seguidos na determinação da região de interesse da sigla DOT da imagem gerada, à esquerda, e da adquirida, à direita. De cima para baixo: imagens binárias; contornos detetados; circunferências envolventes dos contornos maiores; e, por fim, região de interesse obtida.	78
7.9	Passos seguidos na determinação da região de interesse do aviso de segurança da imagem gerada, à esquerda, e da adquirida, à direita. De cima para baixo: imagens binárias; contornos detetados; circunferência envolvente do maior contorno à esquerda; circunferências envolventes dos contornos maiores; e, por fim, região de interesse obtida.	79
7.10	Ilustração das medidas utilizadas no escalamento das imagens: distância entre o aviso de segurança e a informação de fabrico, dada pelas setas horizontais; e altura do aviso de segurança, pelas setas verticais.	80
7.11	Ilustração do processo de seleção da região de interesse vertical da imagem adquirida.	81
7.12	Excerto do resultado do processo de sincronização evidenciando a zona da informação de fabrico.	82
7.13	Excerto do resultado do processo de sincronização evidenciando a zona da informação das dimensões do pneu.	83
7.14	Excerto do resultado do processo de sincronização evidenciando a zona do logótipo da marca e do número identificativo do pneu.	84
7.15	Excerto do resultado do processo de sincronização evidenciando a zona do aviso de segurança.	84
7.16	Ilustração comparativa do efeito da iluminação segundo duas direções, em cima, e apenas uma direção, em baixo.	85
7.17	Ilustração da presença de elementos de ruído próximos da zona do aviso de segurança.	85
7.18	Ilustração da perda de sincronismo entre as imagens gerada, em cima, e adquirida, em baixo.	86
8.1	Zonas relativas ao campos de informação do país de fabrico, em cima, e dos números identificativos do pneu, em baixo, numa imagem gerada.	88
8.2	<i>Templates</i> criados para localização dos campos de informação do país de fabrico, em cima, e dos números identificativos do pneu, em baixo, numa imagem gerada.	88
8.3	Resultados da deteção de cada <i>template</i> na imagem gerada e respetiva área correspondente na imagem adquirida. De cima para baixo: informação de fabrico, país de fabrico e números identificativos do pneu.	90
8.4	Imagem resultante do processo de aumento de contraste.	90

8.5	Imagem resultante do processo de binarização, em cima, e da sua dilatação, em baixo.	91
8.6	Imagem resultante da representação dos contornos detetados na imagem dilatada.	91
8.7	Imagem resultante do processo de definição da região de interesse.	91
8.8	Imagem resultante da representação das mínimas circunferências envolventes dos contornos, em cima, e da eliminação do ruído, em baixo.	91
8.9	Imagens relativas aos passos seguidos na definição da imagem para leitura da informação de fabrico. De cima para baixo: aumento de contraste, binarização, dilatação, contornos detetados, região de interesse, mínimas circunferências envolventes dos contornos, e eliminação do ruído.	92
8.10	Imagens relativas aos passos seguidos na definição da imagem para leitura do país de fabrico. De cima para baixo: aumento de contraste, binarização, dilatação, contornos detetados, região de interesse, mínimas circunferências envolventes dos contornos, e eliminação do ruído.	93
8.11	Ilustração do resultados do agrupamento de contornos identificados através de <i>bounding boxes</i>	95
8.12	Ilustração do resultado da correção da sobreposição de conjuntos de contornos com o seu agrupamento.	96
8.13	Ilustração da aplicação do processo de verificação da presença do conjunto de contornos relativo ao logótipo <i>ECO PLUS</i> na imagem adquirida. De cima para baixo: zonas de localização candidatas, dadas por <i>template matching</i> ; melhor correspondência de <i>features</i> ; e zona validada para localização.	96
8.14	Ilustração da aplicação do processo de verificação da presença do conjunto de contornos relativo à escrita <i>TECHNOLOGY</i> na imagem adquirida. De cima para baixo: zonas de localização candidatas, dadas por <i>template matching</i> ; melhor correspondência de <i>features</i> ; e zona validada para localização.	97
8.15	Ilustração da aplicação do processo de verificação da presença do conjunto de contornos relativo à escrita <i>CONTINENTAL</i> na imagem adquirida. De cima para baixo: zonas de localização candidatas, dadas por <i>template matching</i> ; melhor correspondência de <i>features</i> ; e zona validada para localização.	98
8.16	Ilustração dos resultados obtidos na leitura da informação relativa aos campos de informação de fabrico, país de fabrico e números identificativos do pneu, respetivamente, de cima para baixo.	99
8.17	Ilustração do resultado do processo de validação da presença de alguns campos de informação na zona do logótipo da <i>CONTINENTAL</i> , com uma <i>bounding box</i> a verde.	100
8.18	Ilustração do resultado do processo de validação da presença de alguns campos na zona da informação relativa ao desgaste do piso, tração e temperatura, com uma <i>bounding box</i> a verde.	101
8.19	Ilustração do resultado do processo de rejeição da presença dos campos de informação, com uma <i>bounding box</i> a vermelho.	102
8.20	Exemplo do campo de informação relativo ao país de fabrico numa imagem adquirida com uma só direção de iluminação.	103
8.21	Ilustração de um conjunto de contornos relativo a um padrão incompleto na imagem gerada, em cima, e a sua deteção na imagem adquirida, em baixo.	104
B.1	Ilustração comparativa da aquisição através de uma câmara de área, à esquerda, e de uma câmara de linha, à direita.	110

B.2	Exemplo de imagem resultante da aquisição de um molde utilizando uma câmara de linha.	110
B.3	Montagem da câmara e lente utilizada no sistema de aquisição.	111
B.4	Montagem dos painéis de LEDs utilizados no sistema de aquisição.	112
B.5	Arduíno UNO.	112
B.6	Controlador utilizado para envio de sinal aos painéis de LEDs.	112
B.7	Fontes de alimentação utilizadas para alimentar a câmara, à esquerda, e o controlador, à direita.	113
B.8	Montagem da câmara com a lente e dos painéis de iluminação sobre a mesa giratória onde se encontra o molde.	113
B.9	Estrutura esquematizada do sistema de aquisição.	114
B.10	Exemplo de aquisição tendo a câmara num ponto fixo e várias direções de iluminação. [11]	115
B.11	Imagens resultantes da técnica de <i>photometric stereo</i> , de cima para baixo: imagem com iluminação pela esquerda, imagem com iluminação pela direita, imagem com iluminação pelo topo, e imagem final gerada.	116

Lista de Tabelas

5.1	Informação extraída da entidade do tipo <i>ARC</i> pertencente ao modelo da figura 5.1.	42
A.1	Letra relativa ao Índice de Velocidade (SI) de um pneu e sua correspondência em km/h	107
A.2	Valores para o Índice de Carga (LI) de um pneu e sua correspondência em kg . . .	108
C.1	Atributos da entidade <i>LINE</i> identificada com o número 1 no modelo da figura 5.1	117
C.2	Atributos da entidade <i>CIRCLE</i> identificada com o número 2 no modelo da figura 5.1	118
C.3	Atributos da entidade <i>LINE</i> identificada com o número 3 no modelo da figura 5.1	118
C.4	Atributos da entidade <i>LINE</i> identificada com o número 4 no modelo da figura 5.1	119
C.5	Atributos da entidade <i>CIRCLE</i> identificada com o número 5 no modelo da figura 5.1	119
C.6	Atributos da entidade <i>CIRCLE</i> identificada com o número 6 no modelo da figura 5.1	120
C.7	Atributos da entidade <i>ARC</i> identificada com o número 7 no modelo da figura 5.1.	120
D.1	Atributos comuns às entidades do formato DXF	122
E.1	Exemplo de valores dos atributos de uma entidade do tipo <i>ELLIPSE</i>	123
E.2	Exemplo de valores dos atributos de uma entidade do tipo <i>SPLINE</i>	124
E.3	Exemplo de valores dos atributos de uma entidade do tipo <i>POINT</i>	125
E.4	Exemplo de valores dos atributos de uma entidade do tipo <i>INSERT</i>	125

Abreviaturas e Símbolos

°	Graus
2D	Duas dimensões
3D	Três dimensões
BFMatcher	Brute-Force Matcher
AVI	Automated Vision Inspection
CAD	Computer-Aided Design
CBV	CAD-Based Vision
DOT	Department of Transportation
DWG	Drawing Format
DXF	Drawing Interchange Format
EUA	Estados Unidos da América
GUI	Graphical User Interface
kg	Quilograma
kHz	Quilohertz
kPa	Quilopascal
lb	Libra
LED	Light Emitting Diode
LI	Load Index
mA	milíampere
MES	Manufacturing Execution System
mm	milímetros
OCR	Optical Character Recognition
OCS	Object Coordinate System
psi	Pound per square inch
PCB	Printed Circuit Board
PTZ	Pan-Tilt-Zoom
RANSAC	Random Sample Consensus
RGB	Red Green Blue
ROI	Region Of Interest
SI	Speed Index
V	Volt
WCS	World Coordinate System

Capítulo 1

Introdução

Com o aumento de competitividade nos mercados e simultaneamente com a especificação cada vez mais detalhada dos diversos tipos de produtos, surge a necessidade de uma melhoria do controlo da qualidade da sua conceção. Com efeito, verifica-se um esforço crescente na investigação de técnicas de controlo de qualidade em que a utilização de sistemas automáticos baseados em técnicas de visão artificial é um assunto atual e largamente estudado.

Nos vários tipos de processos de produção, é possível identificar uma vasta gama de produtos cuja manufatura segue um modelo previamente projetado. A verificação da conformidade das características finais de cada produto com o seu modelo, é neste contexto a principal garantia que visa assegurar a qualidade da produção. Assim, esta dissertação incide no estudo e implementação de um sistema de inspeção que utiliza visão artificial para automatizar o processo de verificação de que o produto em estudo segue as especificações estabelecidas no seu modelo de projeto.

De maneira a introduzir o tema, divide-se este capítulo em três secções: em “**Motivação**”, secção 1.1, expõem-se as razões que enaltecem o desenvolvimento desta dissertação e a sua aplicação prática; em “**Objetivos**”, secção 1.2, são descritos os principais pontos propostos a desenvolver; e em “**Estrutura da Dissertação**”, secção 1.3, é indicada a estrutura de conteúdos deste documento.

1.1 Motivação

Os criteriosos requisitos de qualidade e competitividade dos mercados necessitam atualmente de métodos rápidos e fiáveis de inspeção [12] para os mais diversos processos de produção. Questões como a minimização de custos aliada à diminuição e prevenção de desperdícios [13], justificam igualmente a referida necessidade.

Os métodos de inspeção mais desenvolvidos são principalmente sistemas de inspeção visual automática (AVI¹ *systems*), que conseguem efetuar de forma fiável as tarefas repetitivas subjacentes aos processos de inspeção e monitorização [12], outrora levados a cabo inteiramente por mão-de-obra humana. A procura da introdução do uso de AVI *systems* advém do reconhecimento das suas vantagens e da necessidade de colmatar as desvantagens da referida mão-de-obra. Esta

¹AVI - Automated Vision Inspection

última: é afetada pela fadiga, variando com a hora do dia e dia da semana; não preserva a consistência a longo prazo, nem entre diferentes inspetores; e necessita de períodos de descanso [14]. Por outro lado, um sistema de inspeção visual automática tem as vantagens de: poder funcionar 24 horas por dia sem necessitar de paragens; manter a consistência nas inspeções a longo prazo; e poder facilmente ter uma velocidade média superior à da mão-de-obra humana [14].

Além disso, os processos de inspeção podem ser muitas vezes difíceis para inspetores humanos. Em certos casos surgem problemas associados ao número muito elevado de diferentes combinações dos componentes de um determinado produto sob inspeção, algumas bastantes similares e difíceis de distinguir. Agrava-se esta situação quando são detalhes com tamanhos reduzidos para o olho humano. Por outro lado ainda, grande parte dos processos de inspeção concretizam-se em tarefas repetitivas que levam ao cansaço do operador que as realiza, induzindo falhas no processo. Todavia, os processos cognitivos que os seres humanos possuem permitem que estes consigam inspecionar alguns produtos que são muito difíceis para os sistemas de inspeção visual automática.

Percebendo a utilidade que os referidos sistemas de inspeção revelam em contexto industrial pelas razões referidas, esta motivação completa-se com uma aplicação prática. Efetivamente, esta dissertação incide no estudo e implementação de um sistema de inspeção visual automática a aplicar num caso de estudo real proposto pela Continental Mabor, empresa produtora de pneus.

A aplicação prática está relacionada com a etapa final da produção de pneus. Nesta etapa são utilizados moldes, previamente produzidos de acordo com as dimensões e características de cada pneu, no processo que confere a consistência final à borracha que lhe irá dar origem. Além disso, com este processo também se dá uma importante alteração na parede² dos pneus, na qual um conjunto de informação presente no molde é gravada na borracha. Esta informação é composta por um conjunto de caracteres e grafismos que traduzem diversas informações relativamente ao pneu, algumas cuja presença é de carácter obrigatório em virtude de certas legislações.

O problema a colmatar prende-se com o facto de que, na inspeção que é realizada aos pneus depois de produzidos, são por vezes detetadas imperfeições nos referidos caracteres e grafismos presentes na sua parede. A principal causa da ocorrência dessas imperfeições é a produção defeituosa dos moldes, à qual se segue uma inspeção que não as deteta. Adicionalmente, sendo a produção de pneus um processo contínuo, quando são detetadas imperfeições após a produção de um pneu, o molde defeituoso que lhe deu origem poderá já ter sido utilizado em novas produções. Há nestes casos uma quantidade considerável de material que é conseqüentemente descartado. Desta maneira, justifica-se a necessidade de haver uma inspeção fiável que garanta inequivocamente que apenas os moldes sem defeitos sejam validados.

O processo de inspeção em questão é então o de efetuar a verificação de que a informação presente no molde está de acordo com as suas especificações prévias. Ainda que não sejam impostos requisitos temporais restritos para o efeito, esta é uma tarefa muito difícil para um operador. Em primeira instância pela razão de que certos campos de informação apresentam um tamanho muito pequeno. Em segunda instância, toda a informação aparece espelhada para que com o processo

²Face lateral de um pneu.

de moldagem seja gravada no pneu no sentido convencional. Com efeito, somando as referidas razões, a leitura da informação no molde torna-se suscetível a erros. Esta verificação é atualmente feita por operadores utilizando plasticina com o objetivo de encostar ao molde e espelhar de imediato a informação nele presente. Outro método utilizado é também a aquisição de uma imagem do molde para posteriormente efetuar o espelhamento num computador e até aumentar o tamanho dos elementos pequenos. Estas revelam-se soluções que apenas auxiliam a inspeção aos operadores mas que, no entanto, não garantem a deteção de todos os erros.

Em nota conclusiva, os sistemas de inspeção visual automática estão a tornar-se cada vez mais difundidos ao substituir a mão-de-obra humana oferecendo resultados muito satisfatórios. Deve reconhecer-se todavia, que há ainda um caminho a percorrer no que toca à investigação e aperfeiçoamento deste tipo de sistemas de forma a torná-los cada vez mais inteligentes e menos suscetíveis a falhas.

1.2 Objetivos

Esta dissertação centra-se no estudo e implementação de um sistema de inspeção visual automática a ser aplicado no caso de estudo real da inspeção dos moldes usados na produção de pneus. Mais concretamente, o seu objetivo final é o de efetuar a verificação de que a informação a ser gravada na parede dos pneus na sua moldagem, está em conformidade com as suas especificações prévias.

Para a aplicação prática em questão, este sistema pressupõe a aquisição de uma imagem do molde para verificar que a informação nele presente está correta, isto é, de acordo com o que é expectável. Com efeito, um elemento no qual esteja a informação correta é necessário para que através de um processo de comparação, seja possível efetuar a referida verificação. Neste caso, o elemento que melhor traduz essa informação é o modelo CAD³ do molde. Este é projetado para a produção do molde e inclui a maior parte da informação que deve constar em cada pneu com ele produzido. No entanto, como se refere na subsecção 2.4.11, após a produção do molde de acordo com o seu modelo, algumas informações específicas de cada pneu são acrescentadas nas diversas fábricas. Em suma, o modelo CAD apenas inclui a informação genérica presente em cada pneu produzido, sendo que nem toda a informação é diretamente comparável.

O primeiro objetivo é então o de estudar o formato do ficheiro do modelo CAD do molde e desenvolver um método para gerar automaticamente uma representação desse modelo em formato de imagem. Seguidamente, o objetivo é o de processar essa imagem anteriormente gerada e a imagem adquirida do molde de maneira a torná-las o mais semelhantes possível, com o intuito de facilitar o seu processo de comparação. Com efeito, pretendem-se depois estudar e validar técnicas de visão artificial para concretizar a comparação entre as duas imagens em questão. Por outro lado, propõe-se ainda a investigação de uma forma de efetuar a verificação da informação presente na imagem do molde, referida como não sendo diretamente comparável com o modelo CAD.

³CAD - *Computer-Aided Design*

Por fim, o objetivo é o de identificar as não conformidades entre as imagens e apresentar o resultado da inspeção do molde real face ao seu projeto em CAD. É importante referir que, como se pode concluir, o cerne desta dissertação é o desenvolvimento do *software* que recebe o modelo CAD e a imagem do molde para efetuar a comparação anteriormente mencionada. Apesar de ser apenas uma parte integrante do sistema de inspeção visual automática em concreto, refere-se que esta dissertação pressupõe o estudo e implementação do sistema em si. Efetivamente, o entendimento dos conceitos a ele associados considera-se como trabalho necessário e, naturalmente, parte integrante dos objetivos desta dissertação.

1.3 Estrutura da dissertação

Esta secção pretende apresentar a estrutura da presente dissertação, que se encontra dividida em 9 capítulos.

No capítulo 1 é apresentada uma nota introdutória, seguindo-se a motivação e, ainda, os objetivos propostos.

Já no capítulo 2 faz-se uma contextualização do problema inerente a esta dissertação.

O capítulo 3 inclui a revisão bibliográfica dos assuntos relacionados com o presente tema. Nomeadamente, os sistemas de inspeção visual automática e a integração do formato CAD em aplicações de inspeção visual.

Segue-se o capítulo 4 que apresenta a arquitetura do sistema de inspeção, o *software* desenvolvido e as etapas de funcionamento.

No capítulo 5, valida-se um método para gerar uma imagem a partir de um modelo CAD em formato DXF. Para tal, inclui-se uma explicação do referido formato e uma forma para efetuar a sua interpretação.

O processo de uniformização das imagens divide-se em 2 capítulos. Assim, a transformação da forma da imagem anteriormente gerada através da conversão do sistema de coordenadas polares para cartesianas, aborda-se no capítulo 6. Por sua vez, o capítulo 7 trata da sincronização do conteúdo das imagens resultantes da aquisição e da transformação anterior. Nestes capítulos explicam-se todos os métodos implementados para tornar as imagens mais facilmente comparáveis.

O capítulo 8 dedica-se à apresentação da concretização do processo de inspeção do molde. Nele explicam-se os métodos propostos para efetuar a verificação da presença da informação no molde, bem como da leitura de alguma informação para apresentar a um operador.

Finalizando, no capítulo 9 são apresentadas algumas conclusões de todo o trabalho concretizado e alguns pontos a ter em consideração no que toca a trabalhos futuros.

Capítulo 2

Contexto

Como foi referido em 1.1, este tema de dissertação surge da identificação de um caso prático da sua aplicabilidade em contexto industrial. O principal objetivo é o de efetuar a inspeção da conformidade da informação presente no molde da parede dos pneus com as especificações previamente estabelecidas.

Desta maneira, este capítulo serve para contextualização do problema: na secção 2.1, encontra-se uma breve apresentação da empresa envolvida neste projeto; na secção 2.2, é resumido o processo de produção de um pneu, de forma a perceber o envolvimento dos referidos moldes; na secção 2.3, apresentam-se os moldes e os os defeitos que podem ocorrer; e finaliza-se na secção 2.4, onde se enumera a informação que aparece na parede dos pneus, esclarecendo-se o seu significado.

2.1 A Continental Mabor

Esta empresa trata-se do polo português do grupo Continental, que é sediado em Hanôver (Alemanha) e é um dos principais fornecedores da indústria automóvel de diversos materiais e tecnologias. Todavia, o grupo empresarial destaca-se como sendo dos maiores fabricantes de pneus a nível mundial. Em Portugal, o referido polo ocupa-se justamente da produção de pneus.

Na figura 2.1 observa-se o logótipo do grupo Continental, em cima, e a vista aérea da unidade fabril do polo português, em baixo.



Figura 2.1: Logótipo da Continental, em cima, e vista aérea das instalações da Continental Mabor, em baixo.

2.2 Produção de pneus

A produção de pneus como atualmente é concebida, segue um processo que pode ser dividido em quatro etapas.

A primeira etapa denomina-se por **misturação**. Nesta fase, a borracha de origem natural ou sintética é misturada com vários produtos químicos (resinas, óleos, óxido de zinco e enxofre, sílica e negro de fumo¹).

Seguidamente, a **preparação** é a etapa onde são produzidas as várias peças que compõem o pneu: a camada interna, que garante que o pneu é estanque, isto é, que não perde ar; as telas, materiais têxteis normalmente constituídas por fibras de *rayon*², poliéster e *nylon*, que nesta fase são impregnadas de composto de borracha, num processo que se chama de calandragem, e posteriormente são cortadas de acordo com as dimensões requeridas para cada pneu; os talões, aros de arame de aço impregnados de composto de borracha e com dimensões específicas em número, diâmetro e combinação de arames; as cintas metálicas, cordas de aço impregnadas de composto de borracha, também estas cortadas de acordo com os requisitos de cada pneu; os perfis de piso, que são a peça do pneu que se encontra em contacto com o solo; e por fim, as paredes, que são as faces laterais do pneu.

Estando todos os elementos prontos, passa-se em terceiro lugar à **construção**. Nesta etapa todas as peças são montadas sequencialmente, desde a camada interna (primeira peça) até ao piso (última peça). Esta fase é um processo bastante complexo não só porque se tratam de peças de borracha, que neste ponto ainda têm um comportamento plástico, mas também porque na montagem

¹Também conhecido como fuligem, uma das variedades mais puras do carvão.

²Nome da seda artificial, uma fibra de celulose.

tudo tem que ser colocado com muita precisão.

Por último, a quarta etapa designa-se por **vulcanização**. Este é o processo que confere a configuração final ao pneu, através do uso dos moldes envolvidos no projeto desta dissertação. Para o efeito, esta etapa ocorre num ambiente de elevada temperatura e pressão durante um intervalo de tempo que varia em função das características do pneu.

Concluindo, é de notar que em função das características de cada pneu, como por exemplo o seu tipo de construção (radial, diagonal, maciça), o processo de produção é suscetível a alterações. Serve assim esta explicitação apenas de contextualização do processo geral de produção.

2.3 Moldes de pneus

Como foi referido na secção anterior, durante a produção de pneus os moldes são utilizados na etapa de vulcanização. Estes têm a função de moldar a borracha resultante da etapa de construção, dando assim origem a cada pneu.

A constituição dos moldes pode ser dividida em diversas partes. De forma simplificada, distinguem-se as partes destinadas à moldagem dos perfis de piso e da parede dos pneus. Para cada produção, é feita a assemblagem destas partes de maneira a envolver a borracha a moldar, de acordo com as especificações prévias de cada parte. A figura 2.2 ilustra um molde após e antes de se completar a assemblagem dos seus componentes.



Figura 2.2: Exemplo de moldes após e antes de se completar a assemblagem dos seus componentes, das esquerda para a direita, respetivamente.

Tendo em conta a aplicação prática subjacente a esta dissertação, apenas a parte do molde relativa à parede do pneu tem interesse. Desta maneira, para efeito de nomenclatura, refere-se daqui em diante ao molde da parede apenas por molde. Na figura 2.3 evidencia-se um desses moldes.

Por sua vez, é também possível distinguir vários componentes nesses moldes. Parte da sua construção é feita através do encaixe de peças com a informação desejada em certas zonas do molde. Ilustram-se na figura 2.4 algumas destas peças, uma zona de encaixe e ainda uma zona depois do encaixe de uma peça num molde.

Estas peças de encaixe são um assunto de interesse para esta dissertação na medida em que um dos tipos de defeitos a inspecionar nos moldes é precisamente a ocorrência de peças trocadas. Na



Figura 2.3: Molde da parede de um pneu.



Figura 2.4: Peças de encaixe, zona de encaixe e zona após encaixe de uma peça no molde, da esquerda para a direita, respetivamente.

verdade, após serem usados na produção de um pneu, os moldes são alterados em função da nova produção à qual se destinam. O processo de alteração é principalmente composto pela troca de algumas das peças encaixadas. Deste modo, ocorre por vezes a produção de moldes defeituosos como resultado da escolha da peça errada na construção do molde.

Para finalizar, formas defeituosas de certos elementos da informação presente no molde são também exemplo de erros que devem ser sempre detetados. Exemplifica-se um destes defeitos na figura 2.5, na qual se deveria ler a sigla DOT³ (espelhada), verificando-se no entanto um erro na letra *D*.

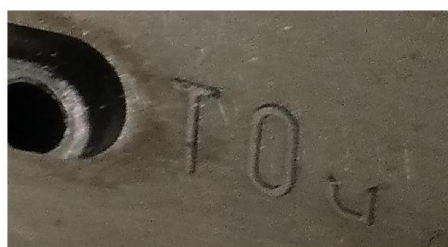


Figura 2.5: Exemplo de um defeito num molde.

³DOT - Department Of Transportation

2.4 Informação na parede de um pneu

A informação presente no molde que é gravada na parede de um pneu após a sua produção, dispõe-se ao longo de toda a secção anelar definida pela face lateral do mesmo. Na figura 2.6, observa-se a vista lateral de um pneu onde se pode observar alguma dessa informação.



Figura 2.6: Exemplo da vista lateral de um pneu.

Como forma de contextualização, é interessante perceber que tipo de informação poderá ser lida e o seu significado. Com efeito, enumeram-se nas subsecções seguintes os vários campos de informação identificados.

2.4.1 Logótipos de identificação

Na parede de um pneu está sempre presente o logótipo da empresa fabricante. Poderá ainda estar o nome ou logótipo do pneu específico, dentro da gama de oferta do fabricante.

Na figura 2.7, observa-se um exemplo do logótipo de identificação do fabricante, em cima, e de identificação do pneu em concreto, em baixo.

2.4.2 Dimensões e características

Um campo que pode ser lido é o relativo à informação das dimensões e características do pneu. Na figura 2.8 pode-se observar um exemplo do formato de apresentação dessa informação.

Assim, neste caso, 225 indica a largura do pneu (letra A da figura 2.9), expressa em milímetros. O valor 50 corresponde à razão entre a altura do pneu e a sua largura, expressa em percentagem (razão B/A da figura 2.9). Por sua vez, a letra R serve como indicação do tipo de construção do pneu, significando neste caso que segue uma construção radial. O número 17 refere-se ao diâmetro interior do pneu, expresso em polegadas (letra C da figura 2.9). Por outro lado, o valor 98 é o



Figura 2.7: Exemplo de logótipos identificativos da empresa fabricante e do modelo do pneu presentes na parede do mesmo.



Figura 2.8: Exemplo de informação de dimensão e características de um pneu presentes na parede do mesmo.

Índice de Carga⁴ (LI) do pneu. Este é um valor tabelado cuja correspondência, em quilogramas, representa a carga máxima que o pneu pode suportar. Por fim, a letra H traduz o Índice de Velocidade⁵ (SI), que indica a categoria de velocidade do pneu, isto é, a velocidade máxima que o pneu suporta com segurança.

Poderá ainda aparecer informação relativa à característica do pneu no que respeita ao uso ou não de câmara-de-ar. Esta informação aparece como pode ser visto no exemplo da figura 2.10.

2.4.3 Informação de fabrico

Outra informação que está sempre presente na parede dos pneus é a que se pode observar na figura 2.11.

O código que se observa na figura 2.11 traduz a informação sobre o fabrico do pneu, que permite saber quando e onde foi produzido. Desta maneira, DOT representa a sigla do Departamento de Transportes dos EUA, que é a entidade responsável pelas normas de segurança dos pneus.

⁴Valores do Índice de Carga (LI) de um pneu disponíveis na tabela A.2 do anexo A.2.

⁵Valores do Índice de Velocidade (SI) de um pneu disponíveis na tabela A.1 do anexo A.1.



Figura 2.9: Representação das dimensões de um pneu: A: largura; B: altura; e C: diâmetro interior.

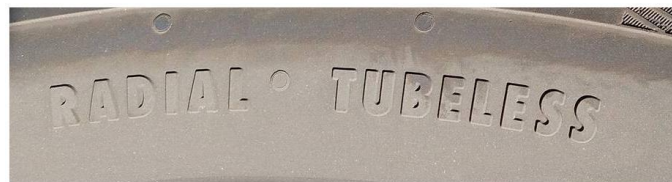


Figura 2.10: Informação relativa ao tipo de construção e à ausência de câmara-de-ar na parede de um pneu.



Figura 2.11: Exemplo de código identificativo de fabrico.

Os dois primeiros caracteres, *Y6* neste caso, representam o código da fábrica onde o pneu foi produzido. Os seis caracteres seguintes, *RB AXF4* na figura 2.11, traduzem as características do pneu, segundo o código da fábrica.

Por fim, os quatro últimos caracteres, que são sempre dígitos, representam a data de fabrico do pneu. Assim, os últimos dois dígitos traduzem o ano e os dois primeiros traduzem a semana em que o pneu foi produzido. No caso em análise, a produção ocorreu durante a 23^a semana do ano de 2007.

2.4.4 País de fabrico

Na figura 2.12 pode observar-se o aspeto da informação identificativa do país de fabrico presente na parede de um pneu.



Figura 2.12: Exemplo de informação relativa ao país de fabrico na parede de um pneu.

2.4.5 Aviso de segurança

Outro campo presente na parede de um pneu é um aviso de segurança, relativo à sua montagem num veículo. Na figura 2.13 pode ser visto um exemplo deste campo.



Figura 2.13: Exemplo de um excerto de marcação de aviso de segurança na parede de um pneu.

2.4.6 Marcações de Inverno

Na figura 2.14 está representado um exemplo de marcações de Inverno. Estas poderão fazer parte do conjunto de campos de informação dependendo do tipo de pneu.

A simbologia *M+S* (iniciais de lama e neve, em inglês) significa que o piso do pneu está desenhado de forma a oferecer mais tração em superfícies suaves. Por outro lado, o desenho da montanha com três picos e um floco de neve no interior, indica que o pneu foi testado e cumpre os requisitos de segurança para utilização na neve.

2.4.7 Constituição do piso e da parede

A constituição do piso e da parede do pneu é outra informação que pode ser lida na sua parede. Um exemplo pode ser visto na figura 2.15.



Figura 2.14: Exemplo de marcação de Inverno na parede de um pneu.

No caso da figura mencionada, a informação é de que a construção do piso do pneu engloba uma tela de *rayon*, duas lonas com cinta de aço e duas telas de poliamida⁶. Por sua vez, a parede é constituída por uma tela de *rayon*.

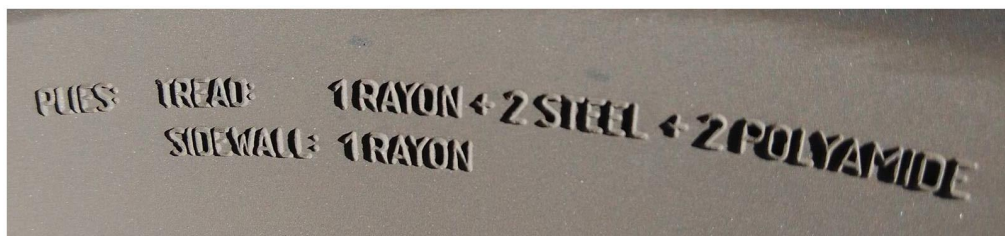


Figura 2.15: Exemplo de informação de constituição do piso e da parede de um pneu.

2.4.8 Desgaste do piso, tração e temperatura

Outro ponto a assinalar é em relação à informação da figura 2.16. O valor de treadwear representa um valor comparativo de desgaste do piso do pneu, que pode variar entre 60 e 700. Ou seja, um pneu com um valor de 400 neste campo, durará duas vezes mais que um com valor 200, em condições de utilização semelhantes. De notar que estes valores poderão mudar consoante o fabricante.

Por outro lado, o valor de tração poderá tomar os valores AA, A, B ou C. Esta letra traduz a avaliação feita ao pneu em relação à capacidade de paragem de um veículo. Assim, um pneu AA terá uma melhor tração e um com C, pior.

Em relação à temperatura, este campo poderá ter os valores A, B ou C. Este atributo representa a resistência que o pneu tem ao calor e a capacidade da sua dissipação. Deste modo, um pneu com temperatura A será mais resistente ao calor, dissipando-o mais. Por sua vez, um pneu com valor C neste campo, será menos eficaz.

⁶Polímero plástico, como por exemplo o *nylon*.



Figura 2.16: Exemplo de informação dos valores de desgaste de piso, tração e temperatura na parede de um pneu.

2.4.9 Pressão e carga

Os valores de carga e pressão máximos aparecem identificados num outro campo de informação. De notar que os valores de carga são indicados quer em kg quer em lb, e os valores de pressão são registados não só em kPa, como também em psi.

Um exemplo desta informação pode ser observado na figura 2.17.



Figura 2.17: Exemplo de informação de valores de pressão e carga na parede de um pneu.

2.4.10 Números identificativos do pneu

Outro campo de informação registado na parede do pneu são os números de série que identificam o mesmo. Na figura 2.18 pode ser visto um exemplo dessa identificação.

2.4.11 Considerações finais

Finalizando, é importante ressaltar que poderão haver ainda outros campos de informação visíveis na parede de um pneu. Isto deve-se ao facto de que poderão haver variações tendo em conta o país da sua comercialização ou a empresa fabricante, que podem incluir códigos ou símbolos específicos.

Outro ponto a assinalar é o de que, como foi referido no ponto 1.2 da página 3, há certos campos de informação que não aparecerão no modelo CAD do molde utilizado na produção do



Figura 2.18: Exemplo de informação referente aos números identificativos de um pneu na sua parede.

pneu. Isto deve-se ao facto de que há certos campos que são mutáveis. Por exemplo, a informação referida em 2.4.3, que inclui dígitos referentes à data de produção, facilmente se percebe que serão constantemente atualizados. Por outro lado, a informação referente ao país de fabrico, 2.4.4, também será atualizada em cada fábrica, não estando presente no modelo CAD. No caso da produção na Continental Mabor, isto acontece pela razão de que os modelos CAD dos moldes são feitos na sede do grupo empresarial, sendo depois enviados para os polos localizados em diversos países. Por fim, ainda os números identificativos do pneu, 2.4.10, são alterados consoante a série de produção do pneu em questão.

Capítulo 3

Revisão Bibliográfica

De maneira a conhecer os conceitos mais importantes associados aos objetivos desta dissertação, serve o presente capítulo para expor os principais assuntos visitados com a pesquisa bibliográfica efetuada. Desta forma, este capítulo encontra-se dividido nas secções: 3.1, "**Sistemas de inspeção visual automática**"; e 3.2, "**Formato CAD e sua integração numa aplicação de inspeção visual**".

Na secção 3.1, incluem-se duas subsecções: a 3.1.1, onde se referem os principais "**Sistemas de iluminação**" aplicáveis à aquisição de imagem; e a 3.1.2, "**Software de aplicação**", na qual são documentados alguns algoritmos que se revelam úteis ao desenvolvimento do sistema de inspeção visual automática em estudo.

Por fim, são expostos na secção 3.2 exemplos de aplicações de sistemas de inspeção visual que incluem modelos CAD no seu funcionamento.

3.1 Sistemas de inspeção visual automática

Os sistemas de inspeção visual automática, ou *AVI systems* como referido anteriormente, são geralmente constituídos por um sistema de aquisição de imagem, um sistema de iluminação e o *software* de aplicação, sendo o seu funcionamento baseado no processamento das imagens adquiridas [12]. É no entanto preciso ter consciência de que os *AVI systems* deverão ser sempre adaptados a cada tarefa de inspeção concreta, tendo em conta as condições em que a mesma ocorre. Fatores como a iluminação do local, posição do objeto a inspecionar ou variações do mesmo [14], devem ser tidos em conta aquando da implementação dos referidos sistemas, pois apenas desta forma se garantem resultados corretos.

Desta maneira, os *AVI systems* podem ser utilizados numa série de aplicações. Em primeira instância pode-se referir a sua aplicabilidade no problema genérico da inspeção de cortes, falhas, arranhões e/ou outros defeitos em peças metálicas [15], peças plásticas, telhas, tecido, entre outros [12]. Em segunda instância, há uma série de casos concretos a referir. Por exemplo, em [12] está descrito o uso de um *AVI system* para inspeção de um molde de injeção plástica e identificação de defeitos nos seus componentes e em [16] é apresentada uma aplicação em que as imagens

adquiridas são utilizadas para gerar o modelo CAD de uma peça metálica. Já em [17] e [18] estão documentadas utilizações práticas de *AVI systems* na indústria alimentar, quer para inspecionar a qualidade de alimentos, quer para os identificar.

Concluindo, percebe-se que há uma vasta gama de aplicações onde os *AVI systems* desempenham tarefas precisas e são sempre adaptados às condições em que estão inseridos e aos objetivos específicos para os quais são projetados. Para o efeito, é imperioso que os sistemas de iluminação e o *software* de aplicação, seguidamente abordados, sejam estudados e escolhidos de forma cuidada.

3.1.1 Sistemas de iluminação

Na área da visão artificial são comumente utilizadas várias técnicas de condicionamento da cena na qual se pretende efetuar a aquisição de imagem. Neste contexto, os sistemas de iluminação são um tema de grande relevo e tornam-se essenciais numa grande variedade de aplicações.

O principal objetivo da iluminação é o de evidenciar, ou até mesmo tornar visíveis, partes/zonas do objeto cuja imagem se pretende adquirir [1]. Por outro lado, também podem ser utilizados sistemas de iluminação para suprimir as partes/zonas não desejadas [1]. Com efeito, as condições de iluminação permitem não só aumentar a informação das imagens adquiridas, mas também reduzir a complexidade dos processos de análise de imagem necessários à extração dessa informação [19].

A escolha do melhor sistema de iluminação para cada aplicação é crucial e depende de vários aspetos que devem ser tidos em conta. Assim, as propriedades dos materiais que constituem os objetos cuja imagem se pretende adquirir, devem sempre ser tomadas em consideração. Questões como a refletividade, a transparência, a regularidade da superfície e a cor desses materiais, influenciam o desempenho dos sistemas de iluminação. De modo semelhante, também as características intrínsecas dos objetos devem ser alvo de atenção. Aqui entendem-se características como a ocorrência de furos, a existência de marcas, a forma do objeto e do seu contorno. Por outro lado, torna-se ainda relevante considerar fatores como o número e a localização de objetos, pois é importante saber se há um número fixo ou variável de objetos no campo de visão da câmara e se os mesmos aparecem separados ou sobrepostos [19].

Posto isto, no ponto 3.1.1.1 referem-se um conjunto de técnicas de iluminação básicas que tiram partido das propriedades de propagação da luz e que devem ser utilizadas em aplicações em função das características dos objetos acima mencionadas. No ponto 3.1.1.2, é ainda apresentada uma breve análise de outras técnicas de iluminação. Por fim, no ponto 3.1.1.3 são exemplificadas algumas aplicações que tiram partido das técnicas referidas.

3.1.1.1 Técnicas de iluminação dependentes das propriedades de propagação da luz

As técnicas de iluminação dependentes das propriedades de propagação da luz podem ser divididas segundo três critérios: a direção da luz, a posição da fonte de luz e a quantidade de luz.

Em primeiro lugar, identificam-se as técnicas de iluminação dependentes da direção da luz. Estas podem ser divididas em técnicas de iluminação difusa (ou omnidirecional) e de iluminação direta. No primeiro caso, a fonte luminosa emite luz em aproximadamente todas as direções, originando situações de reflexão homogénea. Na figura 3.1, observa-se uma ilustração de um sistema de iluminação difusa. [1]

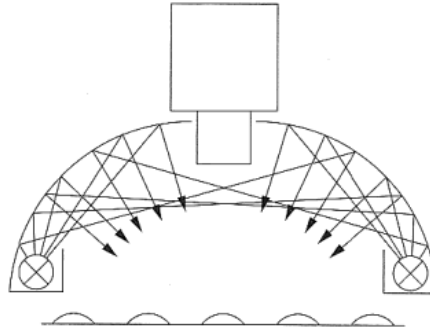


Figura 3.1: Exemplo de sistema de iluminação difusa em que se utiliza uma superfície refletora para difundir a luz. [1]

No caso da iluminação direta, a luz é emitida segundo um conjunto reduzido de direções. Na figura 3.2, ilustra-se um exemplo de um sistema de iluminação direta.

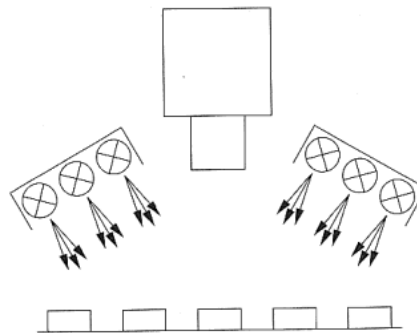


Figura 3.2: Exemplo de um sistema de iluminação direta: um painel de LEDs com luz focalizada. [1]

Distinguem-se seguidamente as técnicas de iluminação dependentes da posição da fonte de luz relativamente ao objeto e à câmara. Neste contexto, poderão ter-se sistemas de iluminação frontal ou de iluminação em contraluz. Se a fonte de luz e a câmara estiverem do mesmo lado do objeto, trata-se de iluminação frontal. Na figura 3.3 pode observar-se um exemplo de iluminação frontal. Contrariamente, se a fonte luminosa e a câmara ocuparem posições opostas em relação ao objeto, tem-se um sistema de iluminação em contraluz, como se pode observar na figura 3.4. [1]

Por último, identificam-se ainda as técnicas de iluminação dependentes da quantidade de luz incidente no objeto que é captada pela câmara. Desta maneira, no caso da fonte de luz estar colocada de tal forma que a maior parte da luz é refletida para a câmara, denomina-se por iluminação

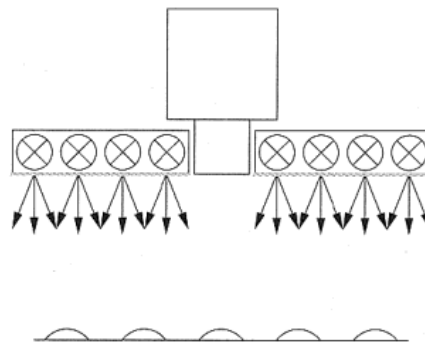


Figura 3.3: Exemplo de sistema de iluminação frontal. [1]

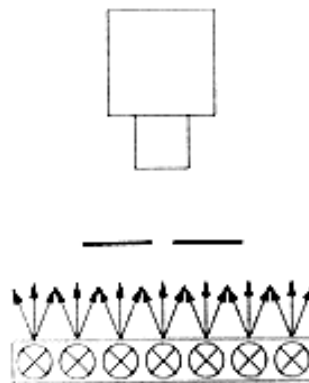


Figura 3.4: Exemplo de sistema de iluminação em contraluz. [1]

em campo claro. Um sistema deste género ilustra-se na figura 3.5. [1]

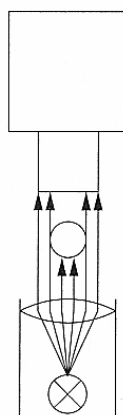


Figura 3.5: Exemplo de sistema de iluminação em campo claro. [1]

Se por outro lado a fonte luminosa estiver colocada de tal maneira que apenas a luz de algumas partes do objeto é refletida para a câmara, sendo a maior parte da luz refletida para fora do seu

campo de visão, tem-se um caso de iluminação em campo escuro. Na figura 3.6 apresenta-se este último caso. [1]

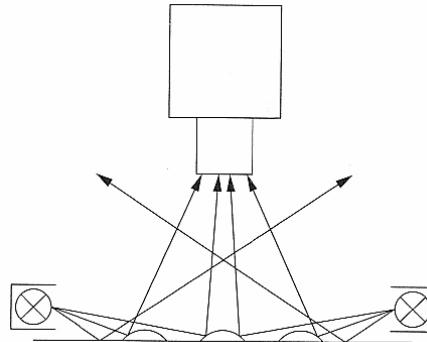


Figura 3.6: Exemplo de sistema de iluminação em campo escuro. [1]

De forma a resumir o que foi supracitado, apresenta-se na figura 3.7 um esquema da distinção das técnicas de iluminação dependentes das propriedades de propagação da luz mencionadas.

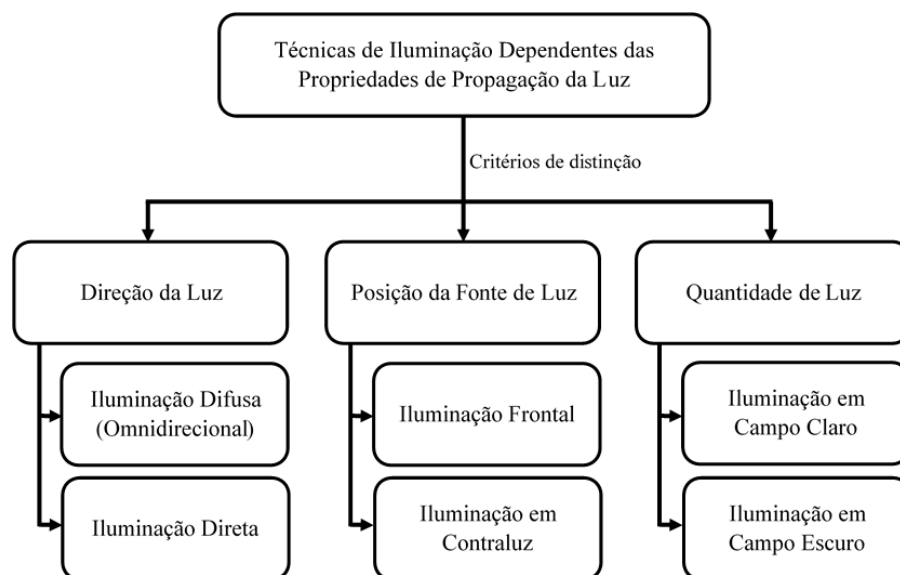


Figura 3.7: Esquema de distinção das técnicas de iluminação dependentes das propriedades de propagação da luz.

Não obstante, ainda que essas técnicas de iluminação tenham sido divididas segundo os três critérios apresentados, os sistemas de iluminação mais comuns são resultado da combinação de várias das referidas técnicas. Ou seja, observando a figura 3.1 pode-se concluir que para além de ser um exemplo de iluminação difusa, também partilha a característica de iluminação em campo claro. De forma semelhante, a figura 3.6 é um exemplo de um sistema de iluminação que combina técnicas de iluminação direta, frontal e em campo escuro. [1]

3.1.1.2 Outras técnicas de iluminação

Para além das técnicas de iluminação já referidas, é possível ainda identificar outras que podem ser usadas na implementação de sistemas de iluminação. Por exemplo, as técnicas que se baseiam nas propriedades da composição espectral da luz são também alvo de interesse pela sua aplicabilidade num vasto conjunto de sistemas de iluminação. [1]

Estas técnicas assentam na premissa de que objetos coloridos apenas refletem parte do espectro de luz, absorvendo a restante parte. Esta particularidade pode ser utilizada quando se pretendem evidenciar certas partes de um objeto de uma dada cor. Para o efeito, deverá escolher-se a iluminação de tal maneira que as partes a evidenciar a reflitam e as partes do objeto que não têm interesse a absorvam. [1]

3.1.1.3 Exemplos de aplicação

A implementação de um sistema de iluminação é na generalidade dos casos uma combinação das técnicas dos critérios acima mencionados. A seleção das mesmas deve ser feita de forma cuidada de acordo com o objetivo pretendido em cada aplicação.

Deste modo, quando se pretendem eliminar sombras, reduzir reflexões especulares ou facilitar a segmentação de objetos, são normalmente utilizados sistemas de iluminação difusa, frontal e em campo claro. Já os sistemas de iluminação direta, frontal e em campo claro, revelam-se úteis em aplicações em que se deseja evidenciar detalhes na superfície dos objetos. [1] Na figura 3.8 está ilustrada uma comparação entre os dois últimos sistemas de iluminação referidos.

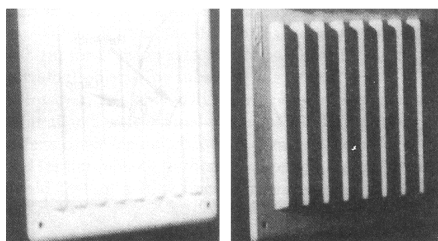


Figura 3.8: Exemplo de aplicação de um sistema de iluminação em campo claro, frontal: e difusa, à esquerda; e direta, à direita. [1]

Para aplicações em que o objetivo é o de realçar saliências ou concavidades nos objetos, são geralmente escolhidos sistemas de iluminação frontal, direta e em campo escuro, como o exemplo ilustrado na figura 2.10. [1] Na figura 3.9 observam-se dois exemplos de aplicabilidade deste tipo de sistema de iluminação.

Em situações em que interessa evidenciar a silhueta de objetos opacos é uma boa escolha a combinação de técnicas de iluminação difusa, em contraluz e em campo claro.[1] Exemplos de aplicação deste sistema de iluminação estão evidenciados na figura 3.10.

De notar que com o sistema de iluminação anterior, a utilização de iluminação difusa faz com que apareçam reflexões no objeto do lado da câmara. Para evitar este fenómeno pode utilizar-se

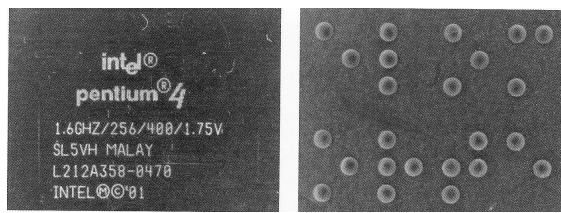


Figura 3.9: Exemplo de aplicação de um sistema de iluminação frontal, direta e em campo escuro. À esquerda, o realce de letras e à direita, o realce de uma impressão em *Braille*. [1]

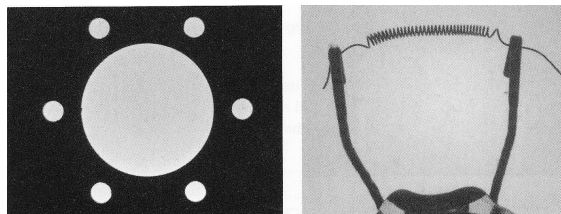


Figura 3.10: Exemplos de aplicação de um sistema de iluminação difusa, em contraluz e em campo claro. [1]

a iluminação direta, implementando-se assim um sistema de iluminação direta, em contraluz e em campo claro. Na figura 3.11 pode-se constatar a diferença induzida pela mudança da técnica baseada na direção da luz, sendo que com a iluminação direta as reflexões do lado da câmara são eliminadas, tendo-se a visualização bem vincada da silhueta do objeto.[1]

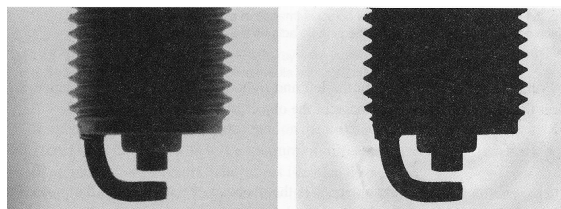


Figura 3.11: Exemplificação da diferença entre um sistema de iluminação em contraluz e em campo claro com luz difusa, à esquerda, e luz direta, à direita. [1]

Por sua vez, quando a intenção é a de facilitar a visualização de contornos de objetos transparentes, podem utilizar-se sistemas de iluminação difusa, em contraluz e em campo escuro.[1] Na figura 3.12 observa-se uma comparação entre a utilização da técnica de campo claro e campo escuro para o objetivo referido.

Finalizando, tendo ainda em conta as técnicas de iluminação baseadas na composição espectral da luz, exemplifica-se a aplicação em que se deseja evidenciar um objeto vermelho sobre um fundo verde. Neste caso, ao se utilizar uma luz vermelha o objeto aparecerá mais claro que o fundo, facilitando a sua segmentação.[1] Na figura 3.13, pode-se constatar a influência da cor da iluminação em imagens adquiridas de uma PCB¹.

¹PCB - Printed Circuit Board

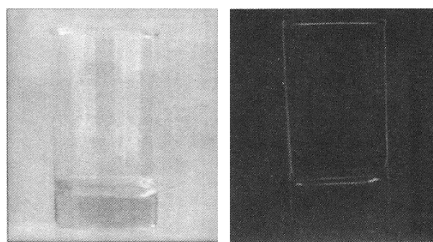


Figura 3.12: Comparação entre a utilização de um sistema de iluminação difusa, em contraluz e em campo claro (à esquerda) e de iluminação difusa, em contraluz e em campo escuro (à direita), para evidenciar o contorno de um copo de vidro.[1]

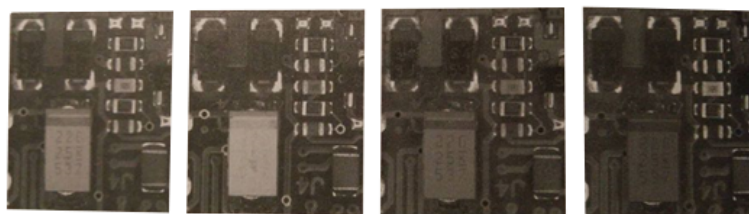


Figura 3.13: Iluminação de uma PCB com luz branca, vermelha, verde e azul, da esquerda para a direita, respetivamente, evidenciando-se o efeito da cor da luz nas imagens adquiridas. [1]

3.1.2 Software de aplicação

Tendo em conta os objetivos desta dissertação, prevê-se o desenvolvimento de um *software* de aplicação para o sistema de inspeção dos moldes da parede dos pneus. Com efeito, neste subcapítulo são abordados alguns algoritmos cujo estudo se revela interessante para atingir o objetivo referido.

3.1.2.1 OCR

O software de aplicação do AVI *system* a desenvolver neste tema de dissertação, tem como um dos objetivos dotar o sistema da capacidade de identificação e reconhecimento de caracteres alfanuméricos nas imagens adquiridas. Esta capacidade torna-se necessária para, por exemplo, a leitura de números de referências de determinadas peças. Com efeito, um estudo de algoritmos de OCR² torna-se necessário para uma compreensão deste assunto.

Em [2] está descrito o uso de um algoritmo de OCR para reconhecimento de caracteres alfanuméricos presentes em documentos digitalizados. O algoritmo apresentado pressupõe as etapas presentes no diagrama de blocos da figura 3.14.

Desta maneira, em primeiro lugar, as imagens a cores adquiridas são convertidas para imagens em tons de cinza (pixels com valores entre 0 e 255, sendo o 0 correspondente ao preto e 255 ao branco) e é-lhes aplicado um filtro para remoção de ruído. Segundo a informação em [2] o

²OCR - *Optical Character Recognition*

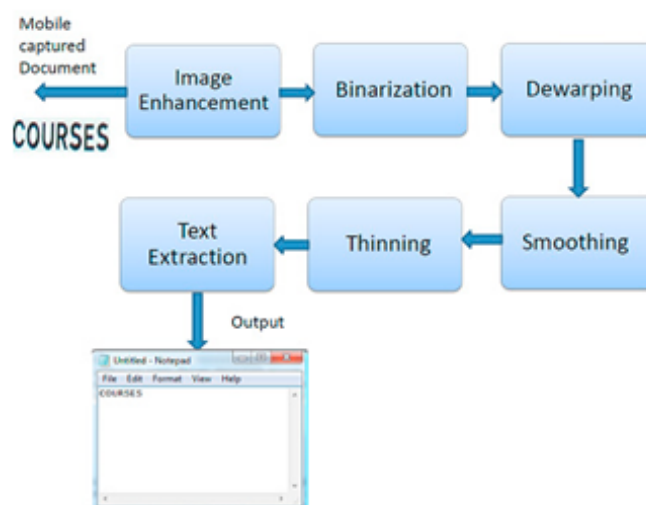


Figura 3.14: Diagrama de blocos do algoritmo de OCR proposto em [2].

filtro aplicado é um filtro de mediana, uma vez que o objetivo é o de remover ruído do tipo sal e pimenta. A esta operação segue-se a de equalização do histograma das imagens, de forma a aumentar o contraste entre os diversos elementos da imagem, sendo, neste caso, aumentado o contraste entre os caracteres alfanuméricos e o fundo.

Em segundo lugar, as imagens são submetidas ao processo de binarização, no qual o valor de cada pixel é convertido para 1 ou 0 em função do valor de *threshold* escolhido. Isto é, se o *threshold* for aplicado à imagem total, os pixels com valor acima do valor de *threshold* serão branco e os restantes serão preto. Em situações em que a influência da iluminação não é uniforme em toda a imagem, deve-se ter em conta a utilização de método de *threshold* adaptativo já que este traz melhores resultados. Na figura 3.15 pode-se observar um exemplo de uma imagem após ter sido binarizada.

A2Y3V

Figura 3.15: Exemplo de uma imagem binarizada. [2]

Em terceiro lugar, como se verifica na figura 3.15, uma vez que os caracteres alfanuméricos estão com um efeito ondulado, é aplicado um algoritmo de transformação geométrica para endireitar o texto. Este tipo de operação é relevante para corrigir distorções frequentemente presentes nas imagens adquiridas, resultantes, por exemplo, da perspetiva de aquisição de imagem. Desta transformação resulta a imagem da figura 3.16.

Em quarto lugar, uma vez que apenas interessa a forma dos caracteres para reconhecer as suas *features*, é aplicada uma máscara à imagem para efetuar a erosão dos caracteres e preservar o seu


 The image shows the text "A2Y3V" in a bold, black, sans-serif font, centered on a white background.

Figura 3.16: Resultado da transformação geométrica para endireitar a imagem. [2]

esqueleto. Na figura 3.17 observa-se o resultado desta operação aplicada à imagem da figura 2.18.


 The image shows the text "A2Y3V" in a thin, black, sans-serif font, centered on a white background. The characters are significantly thinner than in the previous figure, representing a skeletonized version of the text.

Figura 3.17: Resultado da operação de erosão da imagem. [2]

Por último, procede-se à extração do texto presente na imagem. Esta etapa é conseguida, primeiramente, pela extração de cada carácter em separado. Na figura 3.18 observa-se o resultado da aplicação de um algoritmo de identificação dos caracteres, onde se coloca um retângulo a contornar cada caractere.


 The image shows the text "A2Y3V" in a bold, black, sans-serif font, centered on a white background. Each character is enclosed in a thin green rectangular border, representing the bounding boxes for each character.

Figura 3.18: Identificação e contorno de cada caractere. [2]

Seguidamente, tendo-se cada caractere em separado, o reconhecimento do seu valor é feito através de um algoritmo de *template matching*, que é uma técnica de processamento de imagem com o objetivo de encontrar numa imagem partes de outra, que funciona como *template*. Deste modo, é necessário ter uma biblioteca com imagens de letras de A a Z e números de 0 a 9, para comparação com o caractere a reconhecer. Com efeito, o processo de comparação é feito tendo em conta várias posições da imagem *template* sobre a imagem a reconhecer. Deste processo é retornada a métrica de correlação, que traduz o quão a imagem *template* corresponde à imagem a reconhecer. Quando se identificar uma correspondência, o caractere é guardado e o processo prossegue com a imagem a reconhecer seguinte. De notar que cada imagem a reconhecer é comparada com todas as imagens presentes na biblioteca de *template*, podendo ser um processo relativamente demorado.

Em [3], está descrito um algoritmo em que, após binarização da imagem é aplicada uma operação de deteção de orlas. Deste resultado, apenas as orlas do lado direito de cada contorno dos caracteres são mantidas, sendo as outras eliminadas. Este resultado pode ser visto na figura 3.19.

De seguida, as imagens resultantes da deteção de orlas são segmentadas. Esta operação traduz-se na identificação de segmentos nas imagens, que são definidos em [3] como sendo um segmento contínuo de pixels pretos. Na etapa seguinte, estes segmentos são categorizados em diferentes



Figura 3.19: Imagem resultante da binarização, à esquerda, e imagem resultante da detecção de orlas do lado direito, à direita. [3]

tipos, tais como: curtos, longos, lineares, curvos, entre outros. Desta maneira, chega-se ao pressuposto deste algoritmo: definir cada caractere alfanumérico pelo conjunto de tipos de segmentos que o constituem, para posterior reconhecimento.

Em [4], por sua vez, o algoritmo para reconhecimento de caracteres tem como diferença o pressuposto de que são usados 23 pontos de cada imagem para comparação com a biblioteca de *templates*. Deste modo, o caractere é reconhecido como sendo aquele para o qual o *template* apresenta um maior número de pontos correspondentes. Na figura 3.20 pode-se observar o diagrama do algoritmo e os 23 pontos da imagem utilizados.

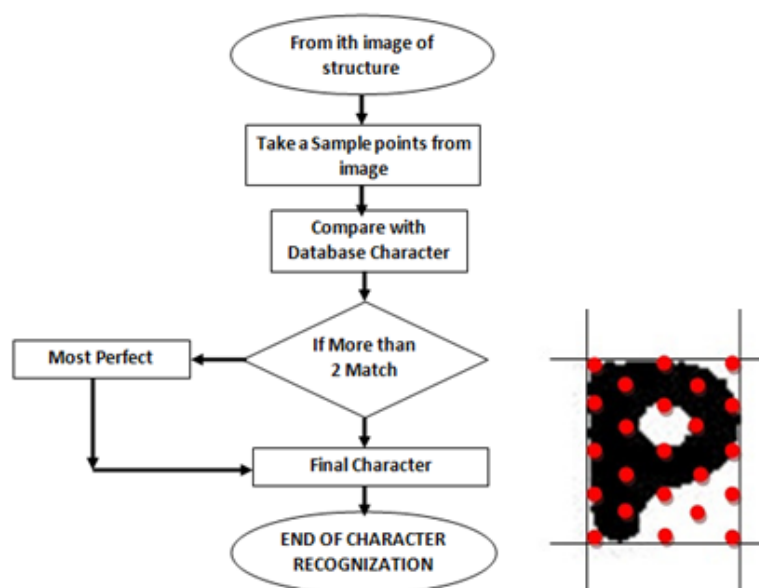


Figura 3.20: Diagrama do algoritmo de OCR, à esquerda, e a seleção dos 23 pontos para comparação da imagem, à direita. [4]

3.1.2.2 *Template e feature matching*

Template matching é um dos assuntos mais populares na área dos sistemas de visão e tem grande relevo no desenvolvimento deste tema de dissertação. Em situações em que o componente a inspecionar possui grafismos, como por exemplo o logótipo da sua marca, será necessário recorrer

a um algoritmo deste tipo para efetuar a sua validação em conformidade com as especificações pré-definidas.

Entende-se então por *template matching* como uma técnica para localizar numa imagem um elemento que a constitui, que é semelhante a um *template* [5]. Para o efeito, o método mais comumente utilizado é o da janela deslizante, segundo o qual a imagem *template* é deslizada sobre a imagem na qual se pretende identificar o elemento, ocupando sequencialmente todas as posições possíveis sobre a mesma. Na figura 3.21 observa-se um exemplo da aplicação do método da janela deslizante.

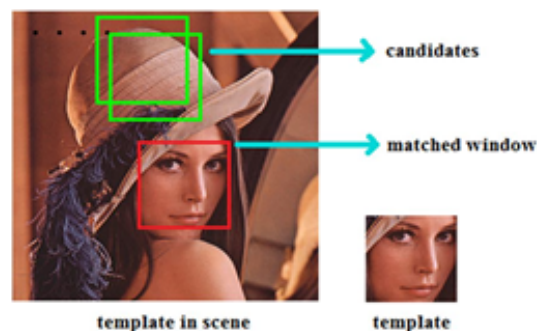


Figura 3.21: Exemplo de aplicação do método da janela deslizante. A verde estão identificadas duas posições possíveis do *template* sobre a imagem. A vermelho está identificada a posição para a qual foi encontrada uma correspondência do *template*. [5]

Há na literatura uma série de algoritmos para determinar a correspondência do *template* a uma posição (ou várias) de uma imagem [5, 20, 6, 21, 22]. Os algoritmos baseados na localização de *features* são os mais documentados e alvo de crescente investigação e melhoramento. Este tipo de algoritmos baseiam-se nas *features* de elementos de imagens tais como cantos [23] e orlas [24] para determinar a correspondência [5].

Em [25], *Paul Viola e Michael Jones* propuseram as *Haar-like features* para reconhecimento de elementos de uma imagem. Cada *Haar-like feature* é dividida em retângulos claros e escuros, sendo o valor de cada *feature* dado pela diferença da soma dos pixels escuros com a soma dos pixels claros. Na figura 3.22 encontram-se alguns exemplos de *Haar-like features*.



Figura 3.22: Exemplos de *Haar-like features*. [5]

Deste modo, uma vez que poderão haver *features* numa imagem com o mesmo valor, várias *Haar-like features* são combinadas para melhorar a sua eficácia. Um exemplo de um algoritmo de *template matching* que utiliza as *Haar-like features* é descrito em [5].

Contudo, vários autores defendem que para uma deteção mais precisa de um elemento numa imagem é necessário utilizar algoritmos com várias etapas de deteção de *features*. Em [5] é proposto um algoritmo com 3 etapas em que os pontos da imagem candidatos à identificação do *template* diminuem a cada etapa, fruto do reajuste de parâmetros. Este é um facto interessante na medida em que os pontos identificados como possíveis correspondências do *template* funcionam como região de interesse (ROI³) na etapa seguinte, não sendo testados todos os pontos da imagem a cada etapa. Na mesma ordem de ideias, em [6] é proposto um algoritmo que também contempla várias etapas, sendo que a resolução do *template* é alterada entre etapas. Deste modo, é primeiramente utilizado um *template* com baixa resolução de forma a reduzir a ROI da imagem para a próxima etapa. Com efeito, é de seguida utilizado o *template* com a resolução original sobre os pontos onde foi encontrada correspondência com o *template* de baixa resolução. Na figura 3.23 encontra-se um exemplo de uma imagem *template* com resolução original e com baixa resolução, notando que, neste exemplo, a imagem de baixa resolução se trata de uma imagem binária.



Figura 3.23: Exemplos de imagens *templates*: *template* original, à esquerda, e *template* de baixa resolução para redução da ROI, à direita. [6]

3.2 Formato CAD e sua integração numa aplicação de inspeção visual

O formato CAD é atualmente muito usado na indústria da manufatura pelo motivo de possibilitar o desenvolvimento de modelos geométricos para os mais diversos objetos [26]. Com efeito, consegue-se uma representação altamente precisa que se traduz numa descrição matemática da forma de cada objeto. Além disso, o formato CAD pode ainda ter informação sobre algumas especificações, tais como a cor, o tipo de material ou o tipo de superfície [7] dos objetos. Por conseguinte, é possível identificar uma vasta gama de aplicações [26, 27, 10, 28, 29, 30, 31, 32, 33] no qual é utilizado o referido formato.

Não obstante, entre todas as aplicações que tiram partido do formato CAD, aquelas em que o mesmo é utilizado a par de um sistema de visão têm particular interesse para o desenvolvimento deste tema de dissertação. Segundo [26], este conjunto é designado por aplicações de *CAD-based vision* (CBV), ou seja, aplicações de visão baseada em CAD.

Ainda mais concretamente, as aplicações de CBV em sistemas AVI são aquelas que traduzem exatamente o foco desta dissertação. Em [7], lê-se que sistemas AVI desenvolvidos para utilizarem

³ROI - *Region Of Interest*

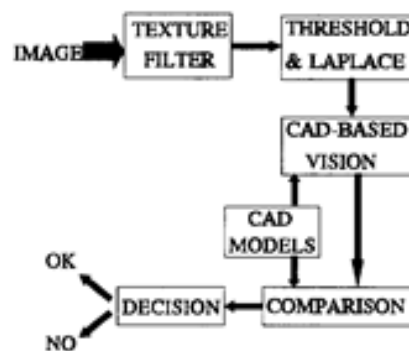


Figura 3.25: Diagrama do sistema proposto em [8].

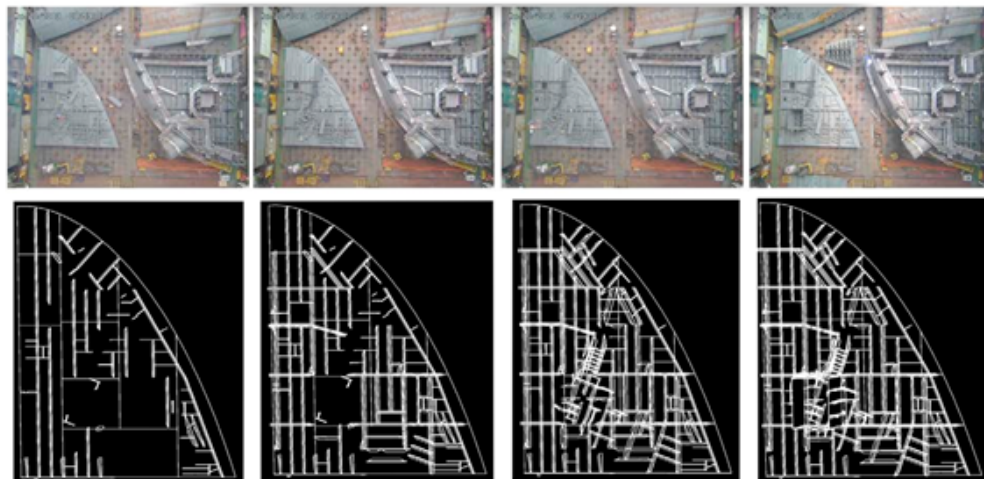


Figura 3.26: Comparação entre imagens adquiridas do topo de um componente (em cima) e as geradas a partir do seu modelo CAD (em baixo) em diversas fases da sua produção. [9]

Por fim, é ainda importante referir a aplicação documentada em [10] de um sistema AVI de CBV em peças de construções na indústria aeronáutica. Nesta aplicação, são comparadas as imagens das peças adquiridas através de uma câmara PTZ⁴ com o seu modelo CAD.

À semelhança de [9], esta comparação também é feita tendo em conta imagens adquiridas das peças e imagens geradas a partir dos modelos CAD respetivos. Não obstante, segundo [10], torna-se clara a forma como o modelo CAD tem um papel fundamental neste tipo de aplicações uma vez que as ditas imagens geradas são denominadas por “imagens teóricas”.

Este desígnio advém do facto de que estas imagens são originadas através de uma representação artificial de cada peça. Com efeito, são consideradas como uma representação perfeita, i.e. a configuração desejada, de uma peça segundo um dado ângulo de visão. Desta maneira, a ideia é a de efetuar a comparação entre uma imagem real adquirida e a “imagem teórica” com a qual a peça

⁴PTZ - Pan-Tilt-Zoom

terá de coincidir, novamente, segundo o mesmo ângulo de visão.

Conclui-se então, que como resultado irá ter-se a identificação de diferenças entre as duas imagens intervenientes na comparação e, conseqüentemente, a identificação de não conformidades da peça real relativamente à configuração desejada.

Na figura 3.27 observam-se exemplos de imagens adquiridas das peças e as “imagens teóricas” geradas a partir do respetivo modelo CAD.

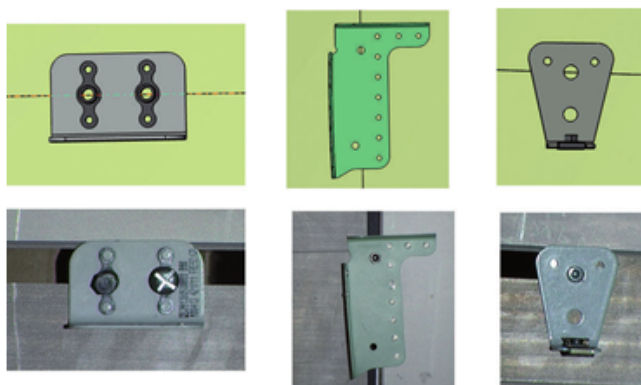


Figura 3.27: Exemplos de imagens geradas a partir de modelos CAD (em cima) usados para comparação com as imagens adquiridas das peças (em baixo). [10]

Capítulo 4

O Sistema de Inspeção

O presente capítulo serve de apresentação ao sistema envolvido no processo de inspeção dos moldes. De forma a expor os principais assuntos este encontra-se dividido em duas secções.

Na secção 4.1, introduz-se a arquitetura do sistema de inspeção. Com efeito, clarifica-se o papel do *software* de aplicação desenvolvido nesta dissertação enquanto elemento do referido sistema.

Na secção 4.2, apresenta-se o *software* mencionado e as etapas da sua estrutura de funcionamento. Estas funcionam como ponto estrutural deste documento, uma vez que é posteriormente dedicado um capítulo à descrição de cada uma dessas etapas. Assim, serve esta subsecção apenas para explicar a sequência e função das mesmas.

4.1 Arquitetura do sistema

O sistema envolvido na inspeção de um molde segue a arquitetura apresentada no esquema da figura 4.1. Este sistema tem em conta que a produção de um molde termina após o mesmo ter sido inspecionado.

A **Inspeção do Molde** é efetuada através de um **Sistema de Inspeção Visual Automática** que efetua a comparação entre uma imagem do molde e o seu modelo CAD. Em primeira instância, este sistema é constituído pelos **Sistema de Aquisição de Imagem** e **Sistema de Iluminação** responsáveis pela aquisição da imagem do molde, cujo princípio de funcionamento se encontra descrito no anexo B, página 109. Pelas razões apresentadas nesse anexo, um **Software de Aquisição** é necessário para receber as imagens adquiridas com diversas iluminações e processá-las de forma a reconstruir uma imagem do molde com melhores propriedades para o seu posterior processamento. Para efeitos de nomenclatura, denomina-se daqui em diante a imagem reconstruída como imagem adquirida. A par do modelo CAD do molde, esta imagem é também uma entrada do **Software de Aplicação**, elemento do sistema responsável por concretizar a tarefa de comparação anteriormente mencionada. Relativamente à sua saída, os resultados são apresentados a um utilizador através de uma **GUI**¹, podendo ser distinguidos dois tipos de informação. Por um lado,

¹GUI - *Graphical User Interface*

é exibida a identificação automática das não conformidades entre o molde e o seu modelo CAD. Por outro lado, é apresentada a informação dita não comparável, referida na subsecção 2.4.11, que necessita de uma intervenção de verificação manual.

Identifica-se seguidamente a presença de um **MES**² enquanto sistema que faz a monitorização e controlo da produção dos moldes. Concretamente, é o sistema responsável por gerar a ordem de produção de um molde específico e que no final recebe o resultado da inspeção vinda da intervenção manual de validação/rejeição do molde tendo em conta a informação apresentada na interface gráfica.

Como se pode observar no esquema da arquitetura, após ser concluído o **Projeto do Modelo CAD** do molde, este é armazenado numa **Base de Dados de Modelos CAD**, podendo ser guardado em diversos formatos de CAD. Por exemplo, no decorrer desta dissertação foram fornecidos os formatos DXF e IGS.

Como foi referido, esta dissertação apenas se destina ao desenvolvimento de uma parte do sistema de inspeção. Inclui-se desta maneira uma **Aplicação de Interface** na arquitetura do sistema como elemento com a função de receber as ordens de produção vindas do **MES** e estabelecer a comunicação com a base de dados, de forma a obter o modelo CAD necessário e disponibilizá-lo para a **Produção do Molde** e ao **Software de Aplicação**. A função desta aplicação é imprescindível ao funcionamento do sistema e como o seu desenvolvimento não é parte dos objetivos desta dissertação, a sua denominação e inclusão nesta arquitetura advém justamente do facto de servir de interface com o *software* alvo de estudo.

Em suma, a arquitetura apresentada representa o sistema de inspeção da produção de um molde de um ponto de vista estrutural. Entre os vários elementos apresentados, apenas o desenvolvimento do **Software de Aplicação** é objetivo desta dissertação e, naturalmente, apenas este se aborda a nível de modo de funcionamento. Todavia, é importante uma compreensão da arquitetura onde o mesmo se insere.

²MES - *Manufacturing Execution System*

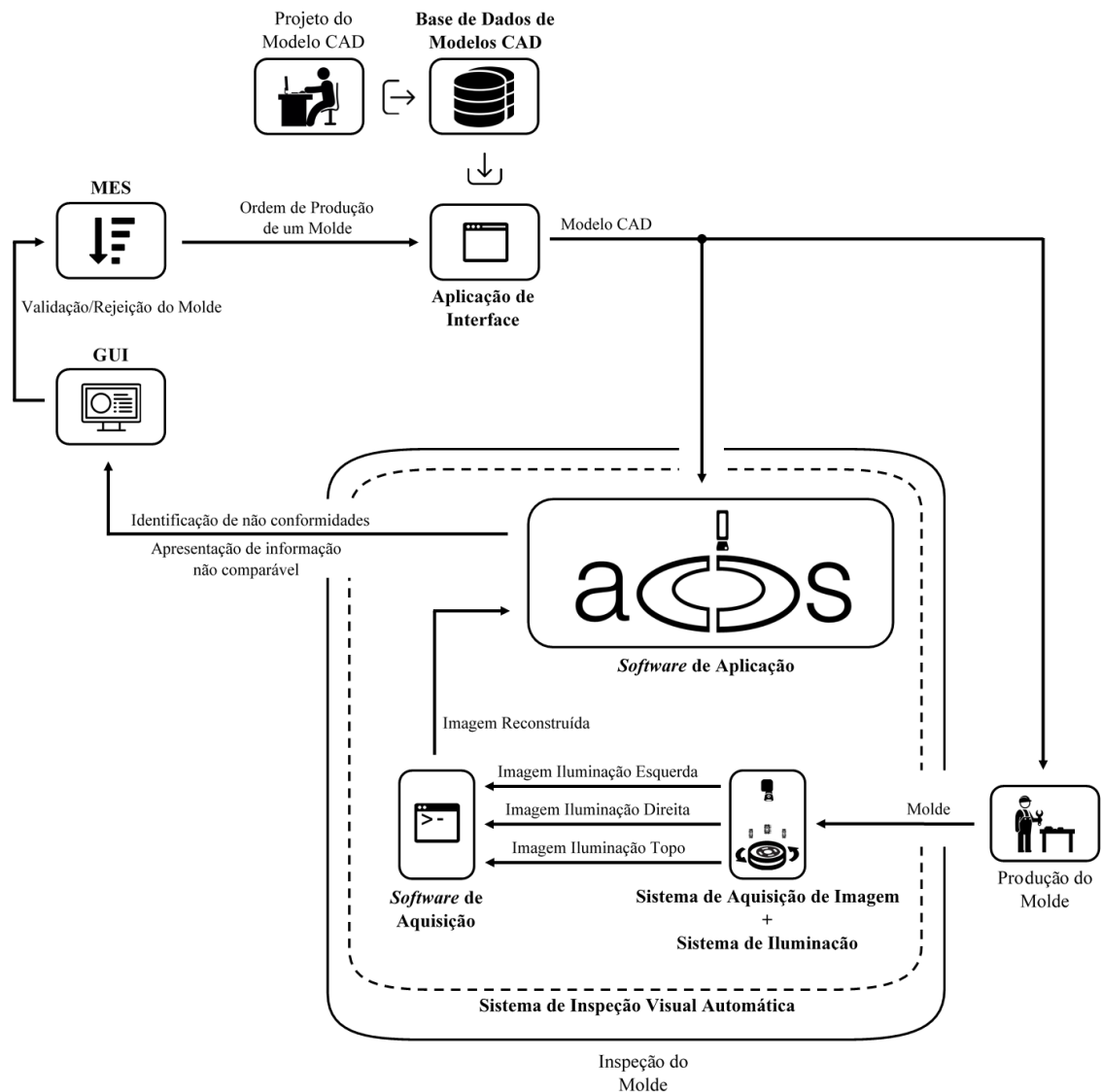


Figura 4.1: Arquitetura do sistema de inspeção.

4.2 O software de aplicação

O *software* de aplicação desenvolvido nesta dissertação para o objetivo do sistema de inspeção visual automática envolvido, está implementado em linguagem Python e é denominado por *acbs*. O seu logótipo pode ser visto no bloco correspondente da figura 4.1, sendo evidenciado na figura 4.2.

O *acbs* tem como entradas uma imagem do molde e o seu modelo CAD, como já foi referido anteriormente. Entre os formatos de ficheiro de CAD disponibilizados, foi escolhido o DXF para utilizar no sistema pela razão de que a sua interpretação é mais simples. No capítulo seguinte, dedica-se a secção 5.1 à apresentação deste formato de CAD.

A nível estrutural a primeira etapa do *acbs* é a **Geração de Imagem a Partir do Modelo CAD**

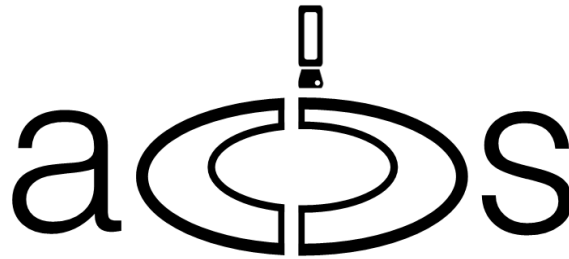


Figura 4.2: Logótipo do *software* de aplicação - *acbs*.

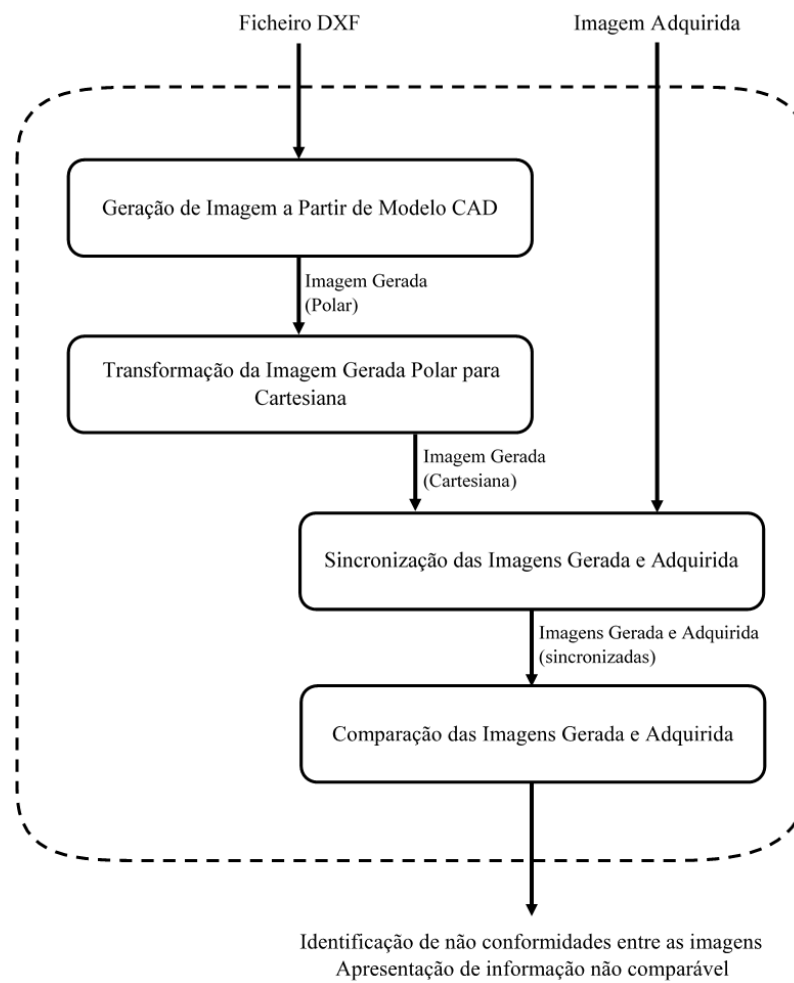
(capítulo 5). A importância desta etapa prende-se com o facto de que para efetuar a comparação da imagem do molde com o seu modelo CAD de forma automática, a melhor maneira de extrair informação do referido modelo é a de gerar uma representação do mesmo em formato de imagem. Assim, a tarefa de comparação é feita entre duas imagens. Denomina-se daqui em diante essa representação do modelo CAD por imagem gerada, para efeito de nomenclatura.

Tendo em conta a forma como a aquisição de imagem do molde é feita (descrita no anexo B, página 109), a imagem adquirida apresenta uma forma retangular. Por sua vez, a imagem gerada apresenta a forma anelar da parede do pneu. Com efeito, a segunda etapa do *acbs* é a **Transformação da Imagem Gerada Polar para Cartesiana** (capítulo 6) de forma a uniformizar as imagens. Entenda-se imagem gerada polar como a de forma anelar e a cartesiana como a imagem retangular resultante da conversão.

Posteriormente, é efetuada a **Sincronização das Imagens Gerada e Adquirida** (capítulo 7). Apesar de a este ponto se terem as imagens com a mesma forma, é ainda necessária uma intervenção na dimensão e conteúdo das mesmas de forma a igualar a disposição da informação.

Por último, efetua-se a **Comparação das Imagens Gerada e Adquirida** (capítulo 8). Aqui são identificadas as não conformidades entre as imagens e é apresentada a informação não comparável.

Para finalizar, o funcionamento descrito apresenta-se esquematizado na figura 4.3.

Figura 4.3: Esquema de funcionamento do *acbs*.

Capítulo 5

Geração de Imagem a partir de Modelo CAD

De forma a concretizar a comparação entre a imagem de um molde e o seu modelo CAD é necessário extrair informação do modelo para possibilitar esse objetivo. A solução encontrada foi a de gerar uma representação do modelo CAD em formato de imagem .

Os vários tipos de *software* utilizados para desenvolver modelos CAD, como por exemplo o AutoCAD ou o SolidWorks, geralmente permitem guardar os referidos modelos em diferentes formatos de imagem. Mesmo quando se tratam de modelos CAD em 3D, é possível gerar uma imagem 2D segundo uma determinada perspectiva. Além disso, existem ainda outros exemplos de *software*, como o Adobe Illustrator, que conseguem importar alguns formatos de ficheiros CAD e convertê-los para formatos de imagem. No entanto, a utilização de todo o *software* mencionado implica um processo manual de geração da imagem do modelo CAD, impossibilitando a implementação de um sistema de inspeção dos moldes automático.

Por outro lado, existem ainda os inconvenientes de que parte desse *software* não realiza a conversão de forma correta, especialmente aqueles sob licença gratuita, como o QCAD ou LibreCAD. Isto acontece fruto de uma importação deficiente do ficheiro CAD devido a problemas de interpretação do formato. Outro ponto de extrema importância é que ao efetuar a conversão desta forma, não existe muito controlo no que toca à informação presente no modelo que realmente se deseja que apareça na imagem gerada.

Esta dissertação prevê o desenvolvimento de um *software* que permita a geração de imagem a partir do modelo CAD de um molde, fornecido em formato DXF. Desta maneira, a geração da referida imagem é feita de forma automática e é feito um controlo sobre a informação que interessa extrair do referido modelo para cumprir o objetivo de comparação com a imagem adquirida do molde.

Estrutura-se este capítulo segundo cinco secções. Na secção 5.1, apresenta-se o formato DXF. Seguidamente, dedica-se a secção 5.2 à explicitação da forma como foi validada uma maneira de interpretar o referido formato. Na secção 5.3, é abordado o método específico de representação de uma imagem a partir do ficheiro com o modelo CAD. A secção 5.4 inclui os resultados obtidos

relativos às imagens geradas. Por fim, a secção 5.5 refere algumas conclusões relevantes relativas aos assuntos expostos no presente capítulo.

5.1 O formato DXF

Os modelos CAD dos moldes utilizados para desenvolvimento desta dissertação foram fornecidos através de ficheiros em formato DXF. Este formato foi desenvolvido pela Autodesk, tendo sido introduzido, em 1982, com o objetivo de tornar possível a interoperabilidade entre o seu *software* AutoCAD e programas de outros fabricantes.

Este formato representa a informação contida no formato nativo do AutoCAD, o DWG, por meio de *tags*. Isto é, um ficheiro DXF é essencialmente constituído por conjuntos de códigos e valores. Esses códigos, definidos como *group codes*, traduzem o tipo ou significado do valor que se lhe segue.

A nível estrutural, encontra-se uma organização do formato dividida por secções. Com efeito, e exemplificando o que foi referido, cada secção é definida pelo código 0, seguido pelo valor *SECTION*. Estas podem ser de vários tipos. Assim, depois da informação código e valor referidos, aparece o código 2 identificativo do tipo de secção em causa. Estas serão: *HEADER*, *CLASSES*, *TABLES*, *BLOCKS*, *ENTITIES*, *OBJECTS* e *THUMBNAILIMAGE*, sendo apenas a última de carácter opcional. Seguem-se então os códigos e valores de cada tipo de secção indicada que traduzem a informação contida no modelo CAD. Cada secção termina novamente com o código 0, mas desta vez com o valor *ENDSEC*.

Ainda que para a definição e interpretação do formato DXF todas as secções referidas sejam necessárias, é a secção *ENTITIES* que tem particular interesse para o objetivo desta dissertação. É imperioso perceber que neste formato os modelos desenhados são definidos através das suas linhas. Isto é, cada modelo fica definido como uma agregação de elementos que, segundo a nomenclatura utilizada neste formato, se designam por entidades. Para clarificar esta ideia de definição segundo entidades, por exemplo, um quadrado poderia ser definido como a agregação de quatro linhas, cada uma sendo um lado.

Existem vários tipos de entidades que podem fazer parte de um modelo, porém, aquelas presentes nos casos estudados foram (segundo a terminologia própria do formato): *ARC*, *CIRCLE*, *ELLIPSE*, *INSERT*, *LINE*, *POINT* e *SPLINE*. Como se pode concluir facilmente, estão definidas na secção *ENTITIES* acima referida, juntamente com os seus atributos. De notar ainda que poderão haver entidades que aparecem agrupadas. Desta maneira, é possível formar blocos de entidades que se definem na secção *BLOCKS*.

Por fim, tendo em conta o supracitado, conclui-se que a maneira como foi conseguido este formato se reflete numa forma de fácil interpretação. Esta foi a principal razão que motivou a sua utilização para o objetivo pretendido.

5.2 Interpretação de um modelo CAD em DXF

Existem várias bibliotecas disponíveis para interpretar e extrair informação dos ficheiros CAD em DXF. A biblioteca escolhida foi a *dxgrabber*, desenvolvida para linguagem Python. Para além de ter uma boa documentação de suporte, esta biblioteca é compatível com todas as versões de ficheiros DXF.

De forma a perceber o funcionamento e a validar a utilização da referida biblioteca no desenvolvimento deste projeto, foi elaborado um modelo CAD simples em DXF para testar a sua interpretação. Na figura 5.1 encontra-se uma imagem do modelo mencionado, na qual foi inserido o referencial de desenho utilizado e um número identificativo de cada entidade do modelo.

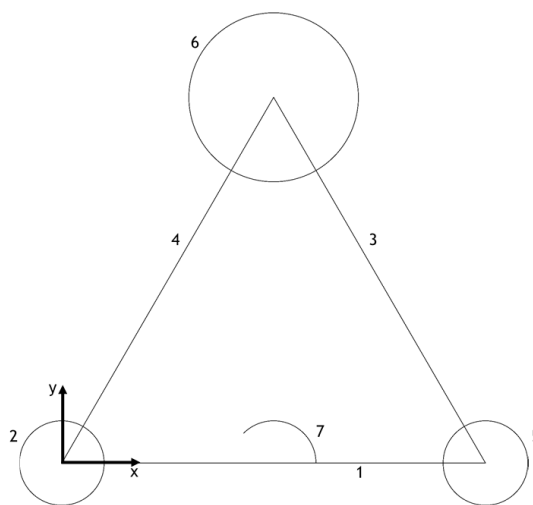


Figura 5.1: Ilustração do modelo CAD desenvolvido para teste da biblioteca com identificação das entidades e referencial de desenho.

Este modelo foi projetado de forma a se terem dimensões e pontos conhecidos, para ser compararem com a informação retornada pela biblioteca. Desta maneira, o modelo é constituído por três linhas (1, 3 e 4 na figura 5.1), três circunferências (2, 5 e 6 na figura 5.1) e um arco de circunferência (7 na figura 5.1). As linhas têm comprimento de 50mm cada uma, definindo assim um triângulo equilátero. Centradas nos vértices desse triângulo localizam-se as circunferências. A circunferência 2 está centrada na origem e tem raio de 5mm. Com igual valor de raio, a circunferência 5, centra-se no ponto de abcissa 50mm e ordenada nula. Por sua vez, a circunferência 6 foi desenhada com raio igual a 10mm e o seu centro situa-se no ponto de abcissa igual a 25mm e ordenada, dada pelo Teorema de Pitágoras, com valor $\sqrt{50^2 - 25^2} \approx 43.3mm$. Por fim, o arco 7, com 135° de amplitude, centra-se no ponto médio da linha 1, tendo valor de abcissa igual a 25mm e ordenada nula.

Foi então desenvolvido um pequeno programa que extraísse a informação do formato DXF do modelo CAD do desenho da figura 5.1 e apresentasse a informação relativa às entidades. Esta

informação está disponível segundo uma série de atributos que caracterizam cada entidade. Na tabela 5.1 observa-se o aspeto do resultado da informação extraída para o arco 7.

Tabela 5.1: Informação extraída da entidade do tipo *ARC* pertencente ao modelo da figura 5.1.

center	(25.00001406018236, 4.2963119949e-06, 0.0)
color	256
dxftype	ARC
endangle	135
extrusion	(0.0, 0.0, 1.0)
invisible	0
layer	0
linetype	None
ltscale	1.0
paperspace	False
radius	4.999999999999996
shadow_mode	None
startangle	5.15e-14
thickness	0.0
transparency	None
true_color	None

Como se pode observar, os valores do centro e raio do arco estão de acordo com o que foi projetado. Além disso, também a amplitude do arco, definida pela diferença entre o valor de *endangle* e *startangle*, resulta nos 135° acima referidos. Disponível no anexo C, na página 117, encontram-se os resultados da informação relativa a cada uma das entidades constituintes do modelo da figura 5.1. Efetuando uma análise aos resultados é possível concluir que a biblioteca extrai de forma correta a informação relativa às entidades e, portanto, valida-se assim a sua utilização.

De notar que há atributos que não foram tidos em conta ao efetuar a validação da biblioteca. Estes tratam-se de atributos comuns a todas as entidades, cuja informação caracteriza principalmente o aspeto de cada entidade do modelo CAD enquanto desenho. Apresentam-se esses atributos, juntamente com o seu significado, na tabela D.1 do anexo D da página 121.

5.3 Imagem gerada

Tendo-se percebido a estrutura do formato DXF, a sua definição por entidades e de que forma se obtém informação das mesmas, dedica-se este ponto a explicitar a maneira específica de como foi gerada a imagem do modelo CAD do molde.

Assim, pensou-se precisamente em tirar partido do facto de que um modelo CAD no formato DXF é definido como uma junção de vários elementos. Dessa maneira, efetuando uma representação de cada entidade, seria possível atingir uma imagem do modelo CAD completo presente no ficheiro. Tendo em conta que as mesmas são definidas sob forma de elementos parametrizáveis, o método de geração de imagem proposto nesta dissertação é o de representar conjuntamente cada

entidade do modelo. Como resultado da combinação da representação de cada entidade, tem-se uma imagem que coincide com o modelo CAD.

Apesar de parecer uma ideia de fácil implementação, alguns problemas surgem na representação de algumas entidades que complicam a sua concretização. Entre eles, uma questão que requereu um tempo considerável de estudo é a influência do vetor de extrusão na representação das entidades. Com efeito, dedica-se a subsecção 5.3.1 à explicitação dos conceitos associados ao referido vetor e a subsecção 5.3.2 à representação das entidades em concreto.

5.3.1 Vetor extrusão e conversão de coordenadas

Os atributos relativos às coordenadas dos pontos das entidades estão definidos tendo em conta um referencial próprio. Este referencial denomina-se por sistema de coordenadas do objeto (OCS¹). Com efeito, a relação entre o OCS e o sistema de coordenadas gerais, denominado por sistema de coordenadas do mundo (WCS²), é dada pelo vetor de extrusão de cada entidade. Este é justamente o vetor unitário que representa o eixo Z do OCS da entidade correspondente. De notar que o referencial WCS é sempre definido pelos vetores unitários segundo x, y, z com valores (1, 0, 0), (0, 1, 0) e (0, 0, 1), respetivamente.

As entidades ditas não planares, como as linhas e os pontos, têm o OCS sempre igual ao WCS, apresentando o vetor de extrusão igual a (0, 0, 1). No entanto, para todas aquelas cujo vetor apresenta um valor diferente é preciso efetuar uma conversão de coordenadas antes de se efetuar a sua representação gráfica. Esta é necessária uma vez que se devem ter todas as coordenadas definidas segundo o WCS.

Tendo em conta um eixo Z qualquer, há uma infinidade de referenciais que se podem definir que o incluam. Basta para isso efetuar uma translação no espaço e/ou rodar os eixos X e Y em torno do referido eixo Z. Todavia, no presente caso de estudo, para cada eixo Z (vetor extrusão) existe apenas um referencial que define univocamente o OCS de cada entidade. Isto advém do facto de que, em primeira instância, a origem de cada um desses referenciais unívocos coincide sempre com a origem do WCS e, em segunda instância, a direção dos eixos X e Y é sempre definida de forma consistente.

O primeiro passo necessário à conversão de coordenadas é a definição do referencial do OCS a partir do seu eixo Z, começando esta pela determinação do eixo X. Desta maneira, o pressuposto é o de analisar o eixo Z do OCS e avaliar se este é aproximadamente igual ao eixo Z (positivo ou negativo) do WCS. Em caso afirmativo, o eixo X do OCS é determinado pelo produto vetorial entre o eixo Y do WCS e o eixo Z do OCS. Em caso negativo, determina-se o eixo X do OCS fazendo o produto vetorial entre os eixos Z do WCS e do OCS. No final desta operação, ter-se-á determinado o eixo X do OCS. Efetua-se posteriormente uma normalização do vetor obtido para que fique com norma unitária. Por fim, o eixo Y do OCS é dado pelo produto vetorial entre os eixos Z e X do OCS.

¹OCS - *Object Coordinate System*

²WCS - *World Coordinate System*

Têm-se agora definidos os vetores unitários que definem os referenciais WCS e OCS. Para efeito da explicação seguinte, passam-se a representar os referenciais mencionados pelos sistemas de coordenadas $o_0x_0y_0z_0$ e $o_1x_1y_1z_1$, respectivamente. A conversão de coordenadas em questão trata-se apenas de uma rotação de referencial, uma vez que os dois referenciais estão centrados no mesmo ponto. Com efeito, este tipo de transformação é feita utilizando uma matriz de rotação. Exemplificando, tendo em conta um ponto p^1 , definido segundo o sistema de coordenadas 1, as suas coordenadas segundo o referencial 0 obtêm-se multiplicando-o à direita pela matriz de rotação R_1^0 .

Equacionando vem :

$$p^0 = R_1^0 p^1 \quad (5.1)$$

Sendo a matriz de rotação em causa definida por:

$$R_1^0 = \begin{bmatrix} x_1 \cdot x_0 & y_1 \cdot x_0 & z_1 \cdot x_0 \\ x_1 \cdot y_0 & y_1 \cdot y_0 & z_1 \cdot y_0 \\ x_1 \cdot z_0 & y_1 \cdot z_0 & z_1 \cdot z_0 \end{bmatrix} \quad (5.2)$$

Para o caso concreto, tendo em conta os valores de x_0 , y_0 e z_0 referidos na subsecção 5.3.1, ao efetuar as operações de produto escalar obtêm-se a seguinte matriz de rotação:

$$R_1^0 = \begin{bmatrix} x_1 & y_1 & z_1 \end{bmatrix} \quad (5.3)$$

Concluindo, para se obterem as coordenadas dos pontos segundo o WCS de uma entidade com vetor de extrusão diferente de (0, 0, 1), apenas é necessário efetuar dois passos. Em primeiro lugar terão de se determinar os vetores representativos do OCS, a partir do vetor de extrusão, conforme explicitado. Por último, multiplica-se à direita a matriz coluna, definida pelas coordenadas do ponto a transformar segundo o OCS, pela matriz de rotação, cujas colunas coincidem com os vetores unitários que definem o OCS.

5.3.2 Representação das entidades

A representação das entidades foi conseguida utilizando a biblioteca *matplotlib*. Esta, desenvolvida para linguagem Python, é utilizada para efetuar representações gráficas em 2D, revelando-se bastante completa e de simples utilização.

É importante notar que as entidades apresentam informação em 3D, isto é, além dos valores de abcissa e ordenada também é incluída a cota nas coordenadas dos pontos correspondentes. Todavia, esta não é tida em conta. Isto é feito pelo motivo de que interessa apenas a vista de topo do modelo CAD dos moldes, pelo que apenas importa efetuar a representação 2D dessa vista.

Desta maneira, dedicam-se os pontos seguintes ao processo de representação de cada tipo de entidade presente nos casos estudados.

5.3.2.1 *CIRCLE*

Uma entidade do tipo *CIRCLE* representa uma circunferência. Esta é simplesmente parametrizável tendo apenas em conta os valores das coordenadas do seu centro e do seu raio, dados respetivamente pelos atributos *center* e *radius* desta entidade. No entanto, é necessário efetuar a conversão das coordenadas do centro para o WCS nos casos em que o vetor de extrusão não apresenta o valor (0, 0, 1). Os atributos deste tipo de entidade podem ser vistos no exemplo do anexo C.2 da página 118.

Ilustra-se na figura 5.2 a representação das entidades do tipo *CIRCLE* do modelo CAD de um molde.



Figura 5.2: Exemplo de representação das entidades do tipo *CIRCLE* do modelo CAD de um molde.

5.3.2.2 *LINE*

O tipo de entidade *LINE* é utilizado para representação de segmentos de linha reta. A sua parametrização é conseguida unicamente com os seus pontos inicial e final, correspondendo aos atributos *start* e *end*, respetivamente. Uma vez que segmentos de reta são entidades não planares, o valor do vetor de extrusão não influencia a representação deste tipo de entidades. O seu valor será assim sempre igual a (0, 0, 1). No anexo C.1 da página 117, pode ser visto um exemplo do conjunto completo de atributos deste tipo de entidade.

Na figura 5.3 ilustra-se a adição das entidades do tipo *LINE*, a verde, ao mesmo modelo CAD da figura 5.2.

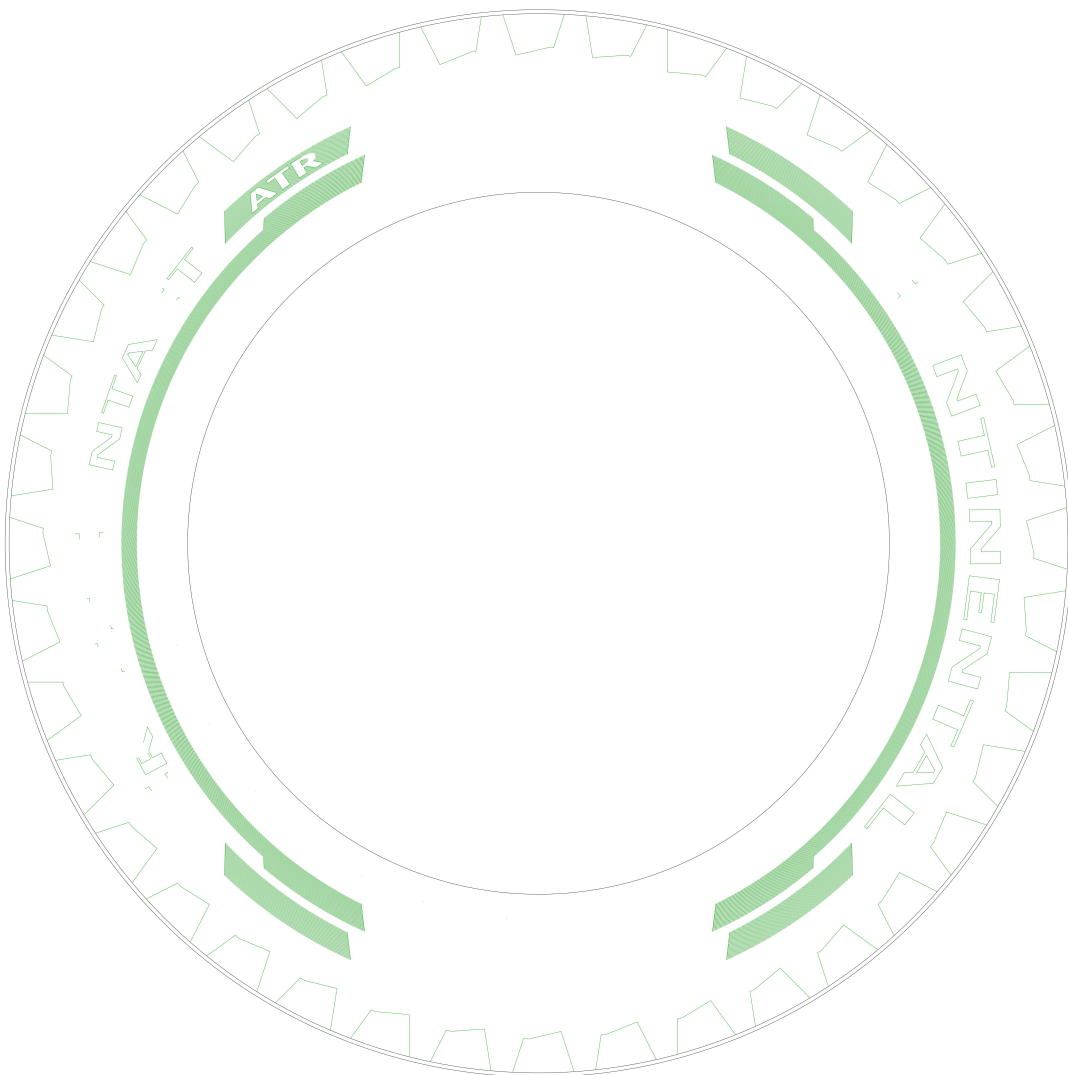


Figura 5.3: Exemplo de representação das entidades do tipo *CIRCLE*, a preto, e *LINE*, a verde, do modelo CAD de um molde.

5.3.2.3 ARC

Entidades do tipo *ARC* traduzem arcos de circunferência. Estes são parametrizados a partir das coordenadas do seu centro, do valor do seu raio e dos seus ângulos inicial e final, tratando-se respetivamente dos atributos *center*, *radius*, *startangle* e *endangle*. Os atributos completos podem ser vistos no exemplo do anexo C.7 da página 120.

No caso deste tipo de entidade, o valor do vetor de extrusão não só deve ser tido em conta na conversão das coordenadas do centro, mas também na conversão dos ângulos referidos. Para apoio da explicação seguinte, ilustra-se na figura 5.4 um exemplo de um arco de circunferência. Como se pode observar, exprimem-se as coordenadas do seu centro por (c_x, c_y) , o seu raio por r , e os seus ângulos inicial e final por α e β , respetivamente.

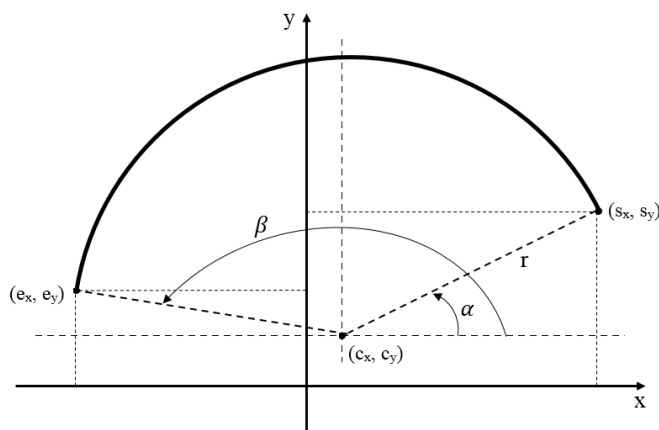


Figura 5.4: Exemplo de representação de um arco de circunferência.

As coordenadas do centro no WCS, representadas por c_x^{WCS} e c_y^{WCS} , são obtidas através do método apresentado na subsecção 5.3.1. Por outro lado, a conversão dos ângulos segue um processo diferente. Em primeiro lugar são determinados os pontos inicial e final de cada arco. Estes estão indicados na figura 5.4 pelos pontos de coordenadas (s_x, s_y) e (e_x, e_y) . Com efeito, os valores das abcissas e ordenadas desses pontos no OCS são dados pelas fórmulas 5.4 a 5.7.

$$s_x^{OCS} = c_x^{OCS} + r \cdot \cos \alpha^{OCS} \quad (5.4)$$

$$s_y^{OCS} = c_y^{OCS} + r \cdot \sin \alpha^{OCS} \quad (5.5)$$

$$e_x^{OCS} = c_x^{OCS} + r \cdot \cos \beta^{OCS} \quad (5.6)$$

$$e_y^{OCS} = c_y^{OCS} + r \cdot \sin \beta^{OCS} \quad (5.7)$$

O segundo passo é então o de efetuar a conversão das coordenadas desses pontos para o WCS, obtendo-se os novos pontos que se passam a representar por (s_x^{WCS}, s_y^{WCS}) e (e_x^{WCS}, e_y^{WCS}) . Estes são necessários para o objetivo de cálculo do novo valor dos ângulos, por sua vez representados por α^{WCS} e β^{WCS} . Utilizando a função *atan2*, pela razão de que esta retorna os ângulos entre $-\pi$ e π , o seu valor é dado por:

$$\alpha^{WCS} = \text{atan2}(s_y^{WCS} - c_y^{WCS}, s_x^{WCS} - c_x^{WCS}) \quad (5.8)$$

$$\beta^{WCS} = \text{atan2}(e_y^{WCS} - c_y^{WCS}, e_x^{WCS} - c_x^{WCS}) \quad (5.9)$$

Seguidamente, efetua-se uma normalização do valor de todos os ângulos (no OCS e WCS) de forma a obter os seus valores entre 0° e 360° . Esta normalização é conseguida fazendo a conversão para dos ângulos para graus e somando 360 a todos os de valor negativo. Determina-se posteriormente o valor da amplitude de cada arco no OCS e no WCS, fazendo a diferença entre os ângulos final e inicial. Do mesmo modo atrás referido, normalizam-se esses valores obtidos das amplitudes.

A necessidade dos cálculos e normalizações referidas prende-se com o facto de que é necessário comparar os valores de amplitude de cada arco no dois referenciais. Apesar da representação gráfica de um arco pela função da *matplotlib* ser sempre feita no sentido anti-horário (sentido positivo), há casos em que deve ser feita no sentido horário (sentido negativo), fruto da conversão de referencial de coordenadas. Exemplifica-se esta questão na figura 5.5, na qual se verifica que o arco do gráfico da esquerda, com vetor de extrusão de $(0, 0, -1)$, após conversão seria representado pela linha contínua do gráfico da direita.

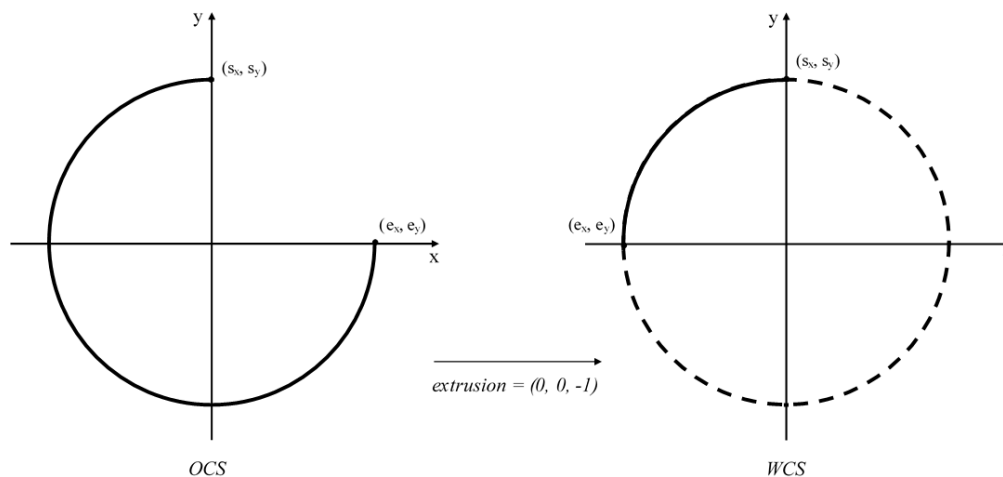


Figura 5.5: Ilustração do efeito da conversão de referencial tendo em conta um vetor de extrusão igual a $(0, 0, -1)$ na representação de um arco de circunferência.

Conclui-se facilmente que para que o arco após conversão tenha a mesma amplitude, a sua representação deve acontecer em sentido horário (sentido negativo) de maneira a ficar igual à

linha tracejada. Como tal não se verifica, é então necessário averiguar antes da representação de cada arco se a amplitude no WCS (dada por $\beta^{WCS} - \alpha^{WCS}$) é aproximadamente igual à amplitude no OCS (dada por $\beta^{OCS} - \alpha^{OCS}$). Em caso afirmativo a sua representação é efetuada normalmente. No caso contrário, invertem-se os ângulos inicial e final para que a representação possa ser feita no sentido correto.

Na figura 5.6 exemplifica-se a representação no sentido anti-horário (sentido da função) após inversão dos pontos inicial e final do arco da figura 5.4.

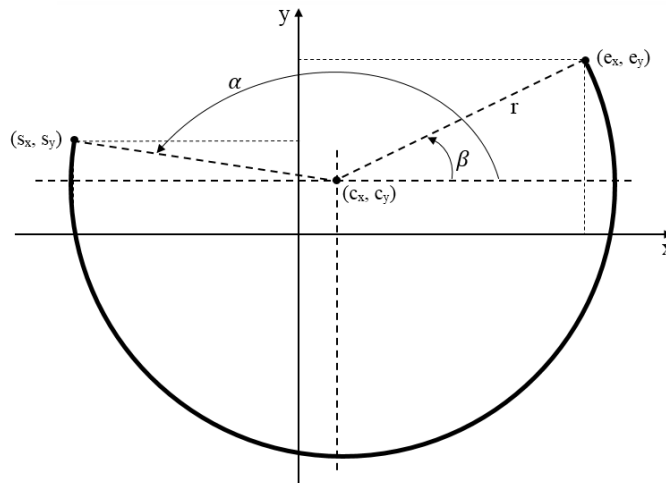


Figura 5.6: Exemplo da inversão dos pontos inicial e final e consequente representação do arco da figura 5.4.

Tendo-se percebido o processo subjacente à representação correta de entidades do tipo *ARC*, ilustra-se na figura 5.7 a adição destas entidades, a azul, ao mesmo modelo CAD da figura 5.3.

5.3.2.4 ELLIPSE

Este tipo de entidade é utilizada para representar elipses. A parametrização de um *ELLIPSE* é feita tendo em conta as coordenadas do seu centro, o ponto de extremidade do seu eixo maior, o rácio entre os eixos menor e maior e, por fim, os ângulos inicial e final. Segundo os atributos desta entidade, a informação referida é dada por *center*, *majoraxis*, *ratio*, *startparam* e *endparam*. Um exemplo da informação completa que é possível extrair deste tipo de entidade pode ser vista no anexo E.1 da página 123.

De maneira a acompanhar a explicação seguinte, ilustra-se na figura 5.8 um exemplo de uma elipse. Representam-se as coordenadas do seu centro por (s_x, s_y) , os seus ângulos inicial e final por α e β , os seus eixos menor e maior por m e M , e as coordenadas da extremidade do eixo maior por (a_x, a_y) .

Deste modo, o primeiro passo é a determinação da medida dos seus eixos. O eixo maior é conseguido fazendo o cálculo da distância entre o ponto da sua extremidade e o centro da elipse.

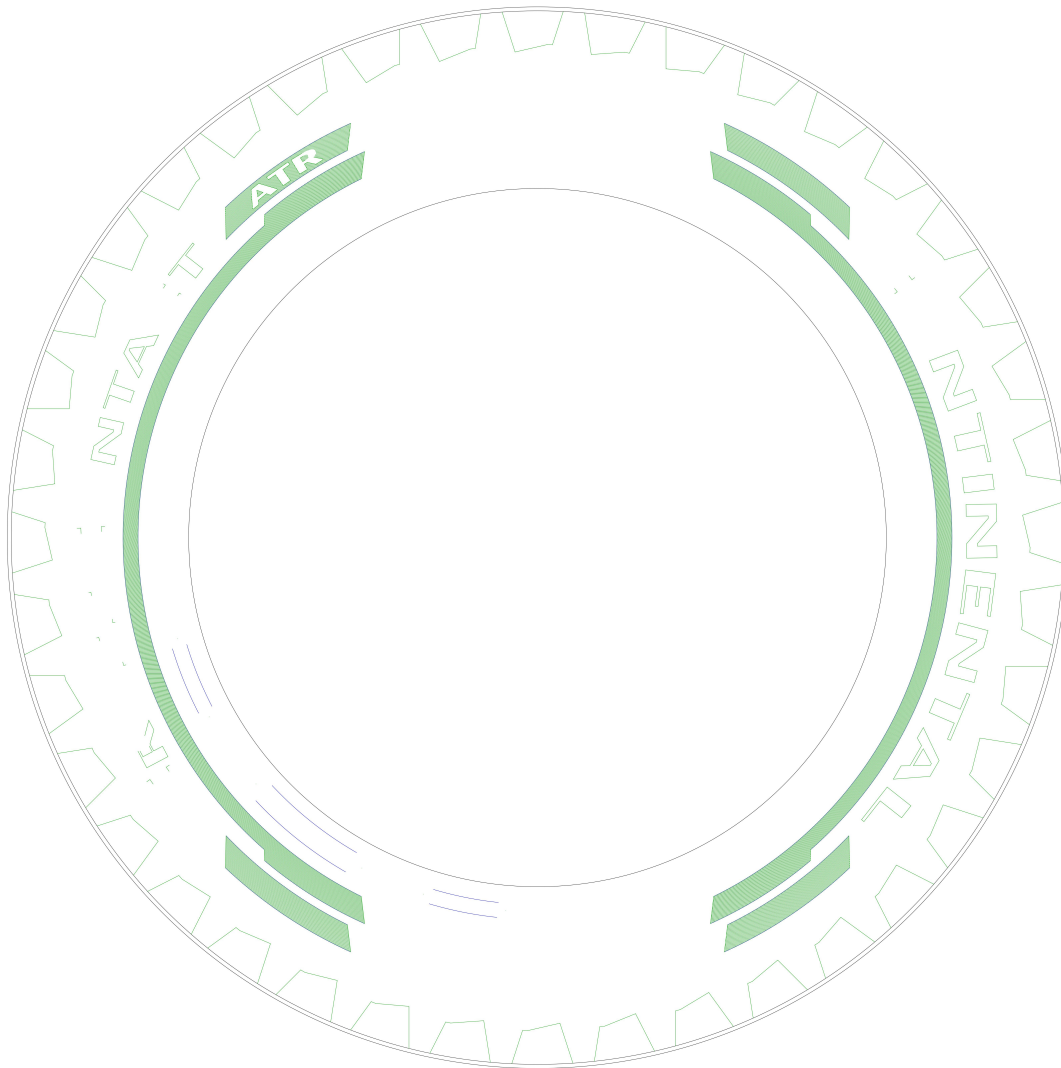


Figura 5.7: Exemplo de representação das entidades do tipo *CIRCLE*, a **preto**, *LINE*, a **verde**, e *ARC*, a **azul**, do modelo CAD de um molde.

Multiplicando o valor obtido pelo rácio entre eixos, determina-se o eixo menor. Representando o rácio por r e a medida dos eixos maior e menor por \bar{M} e \bar{m} , equacionando vem:

$$\bar{M} = \sqrt{(a_x - c_x)^2 + (a_y - c_y)^2} \quad (5.10)$$

$$\bar{m} = r \cdot \bar{M} \quad (5.11)$$

Uma particularidade a ter em conta na representação de uma entidade deste tipo é que os valores dos ângulos inicial e final são definidos a partir do ponto extremo do seu eixo maior. Ou seja, é necessário calcular o ângulo entre o eixo maior da elipse e o eixo das abcissas, definindo-se

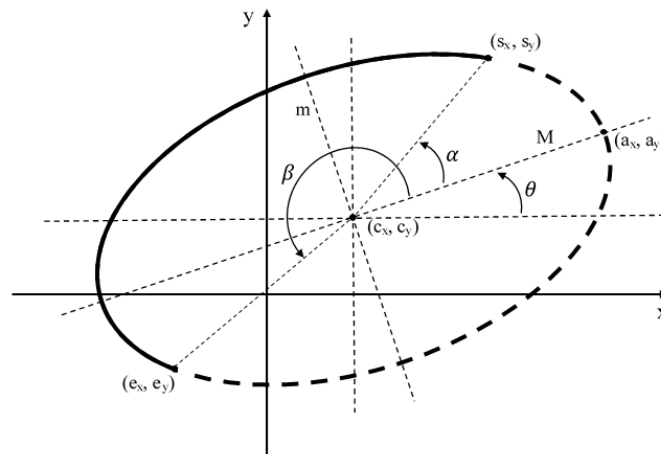


Figura 5.8: Exemplo de representação de uma elipse.

um ângulo base. Este é o ângulo representado na figura 5.8 por θ e é dado por:

$$\theta = \text{atan2}(a_y - c_y, a_x - c_x) \quad (5.12)$$

De seguida, é ainda necessário efetuar uma normalização do valor dos ângulos da mesma forma descrita para entidades *ARC*, de maneira a obter valores entre 0° e 360° . Determinam-se então os valores dos ângulos inicial e final somando-lhe o ângulo base determinado.

A função utilizada na representação desta entidade necessita dos parâmetros de largura e altura de cada elipse. Com efeito, uma última verificação é necessária já que tendo em conta a orientação da elipse os seus valores de altura e largura variam. No caso do ângulo base estar entre 315° e 45° ou estar entre 135° e 225° (áreas a cinzento na figura 5.9) a largura é dada pelo eixo maior e a altura pelo eixo menor. Pelo contrário, no caso do ângulo base estar entre 45° e 135° ou entre 225° e 315° (área a branco do gráfico da figura 5.9), a largura é dada pelo eixo menor e a altura pelo eixo maior.

Ilustra-se finalmente na figura 5.10 o resultado da adição de entidades *ELLIPSE*, a vermelho, ao modelo CAD da figura 5.7.

Em conclusão, ainda que o vetor extrusão tenha provavelmente influência nos cálculos descritos, nos casos de estudo disponíveis nenhuma elipse apresenta extrusão diferente de $(0, 0, 1)$. Desta maneira, não é possível comprovar a referida influência pelo que não se inclui na representação desta entidade.

5.3.2.5 *SPLINE*

Entidades do tipo *SPLINE* traduzem linhas paramétricas definidas através de dois ou mais pontos de controlo. Na aplicação em estudo, estas são utilizadas para desenho de linhas contínuas. Para a sua representação gráfica apenas é necessário ter em conta a sua ordem e os referidos pontos de controlo respetivos. Esta informação é dada pelos atributos *degree* e *controlpoints*,

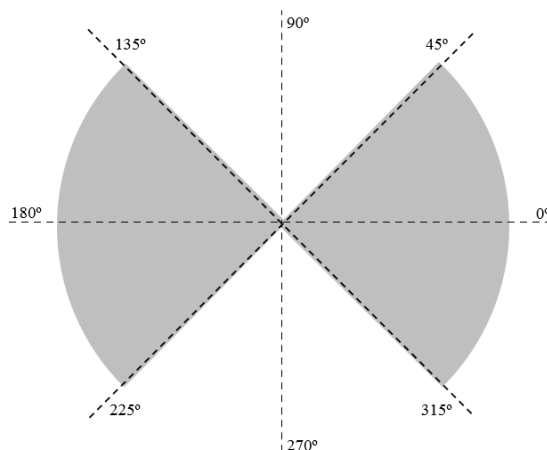


Figura 5.9: Representação dos ângulos utilizados na decisão do valor de largura e altura de cada elipse.

respetivamente. Desta forma, cada linha resulta de uma interpolação dos referidos pontos, de ordem igual à referida na informação da entidade. Um exemplo do conjunto da referida informação pode ser vista no anexo E.2 da página 124.

No conjunto de casos de estudo disponíveis, apenas aparecem *SPLINEs* de primeira e terceira ordem. Para o primeiro caso efetua-se uma interpolação linear dos pontos de controlo, cujo resultado é a união desses pontos com segmentos de linha reta. Para o segundo caso, é necessário efetuar uma interpolação cúbica afim de se obter a equação a representar. Na figura 5.11 ilustra-se um exemplo de *SPLINEs* resultantes de uma interpolação dos pontos cinzentos, de primeira ordem, pelo tracejado, e de terceira ordem, pela linha contínua.

Os cálculos subjacentes às interpolações referidas foram efetuados utilizando funções da biblioteca *scipy*³ destinadas ao efeito.

Por fim, ilustra-se na figura 5.12 o resultado da adição de entidades *SPLINE*, a laranja, ao modelo CAD da figura 5.10.

5.3.2.6 POINT

Entre o conjunto de entidades estudadas, o tipo *POINT* é o de mais simples interpretação. Efetivamente, apenas representa um ponto que se define unicamente pelas coordenadas do seu atributo *point*. Um exemplo do conjunto de atributos que é possível extrair pode ser visto no anexo E.3 da página 125.

Por fim, ainda que se tenha analisado este tipo de entidade, uma vez que constava de alguns modelos CAD do conjunto de casos de estudo, não são incluídos na imagem gerada. A razão deste facto é de que os pontos não são perceptíveis após serem representados.

³Biblioteca para linguagem Python utilizada para operações de cálculo matemático.

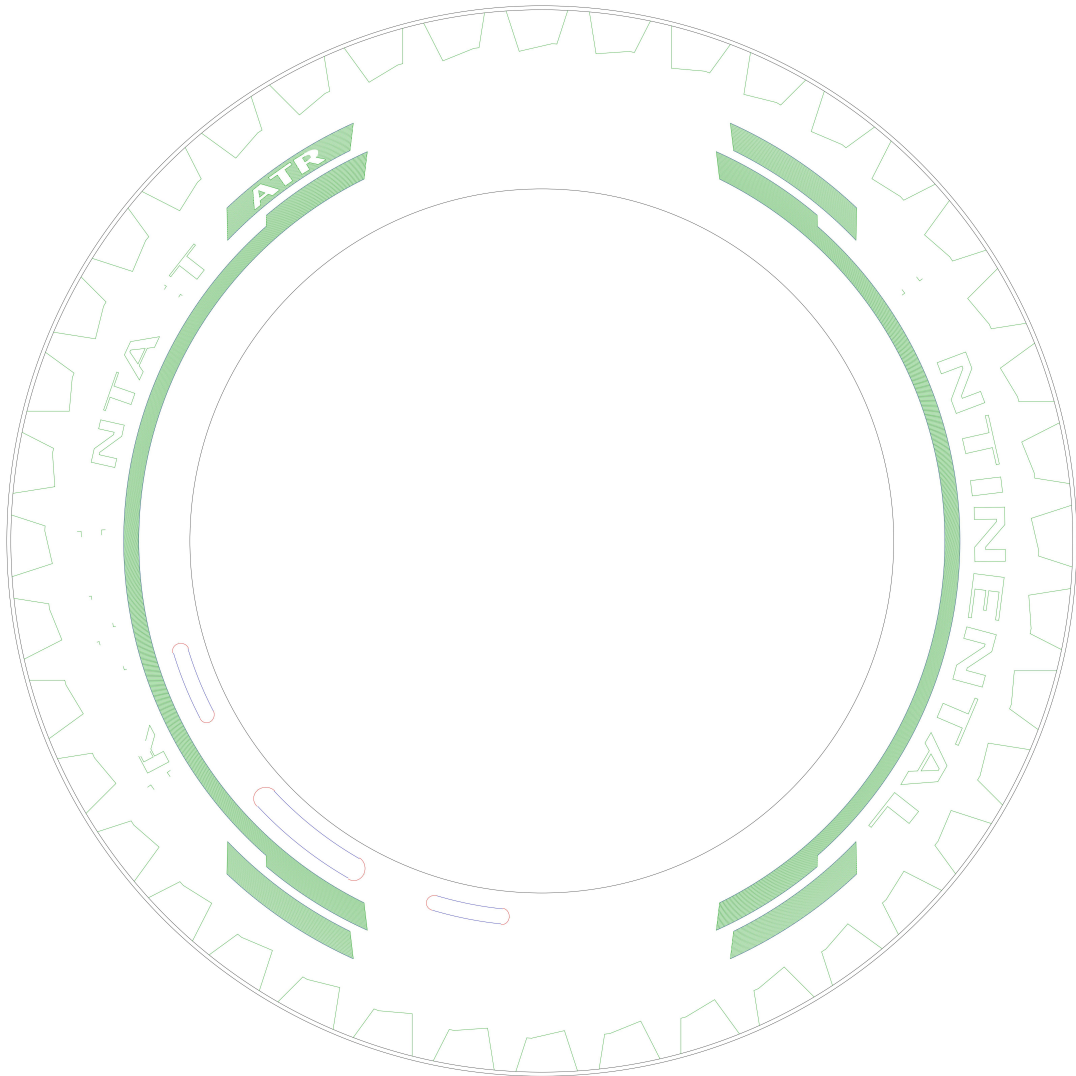


Figura 5.10: Exemplo de representação das entidades do tipo *CIRCLE*, a **preto**, *LINE*, a **verde**, *ARC*, a **azul** e *ELLIPSE*, a **vermelho**, do modelo CAD de um molde.

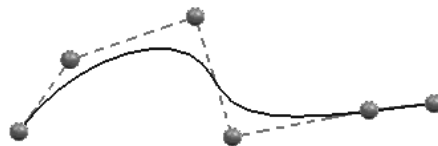


Figura 5.11: Exemplo de *SPLINEs* resultantes de uma interpolação de primeira ordem, pelo tracejado, e de terceira ordem, pela linha contínua.

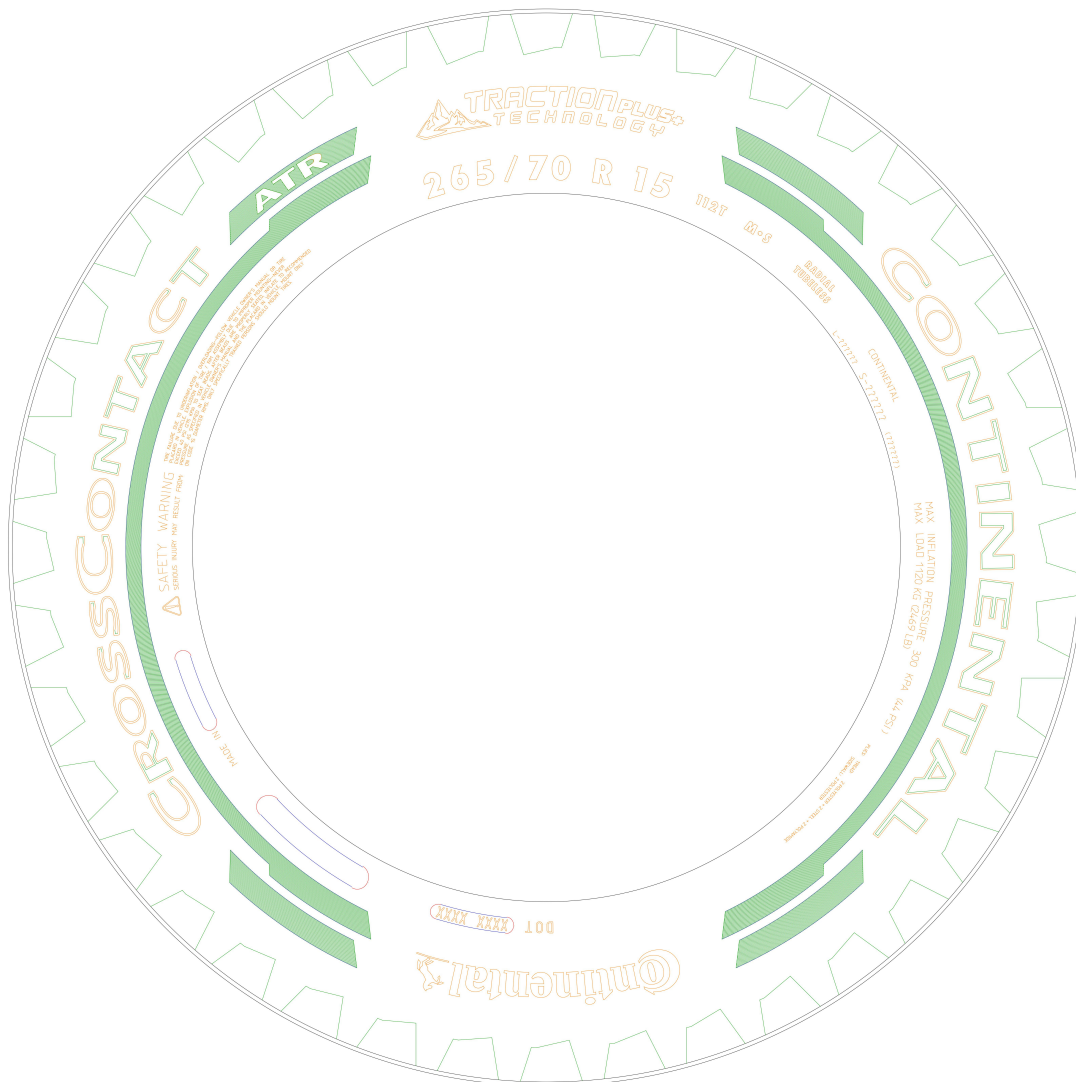


Figura 5.12: Exemplo de representação das entidades do tipo *CIRCLE*, a **preto**, *LINE*, a **verde**, *ARC*, a **azul**, *ELLIPSE*, a **vermelho**, e *SPLINE*, a cor de **laranja**, do modelo CAD de um molde.

5.3.2.7 *INSERT*

Uma entidade do tipo *INSERT* representa um caso especial, pois esta não exprime concretamente um elemento graficamente representável. Na verdade, esta entidade traduz uma referência a um bloco de entidades dos tipos referidos nas subsecções precedentes. Desta maneira, é necessário apenas ter em conta o atributo *name*, que corresponde ao nome do bloco referenciado, e representar todas as entidades que lhe estão associadas. Um exemplo da informação que é possível extrair deste tipo de entidade pode ser vista no anexo E.4 da página 125.

Para finalizar, ilustra-se na figura 5.13 o resultado da adição de entidades *INSERT*, a roxo, ao modelo CAD da figura 5.12, obtendo-se assim a imagem gerada completa.

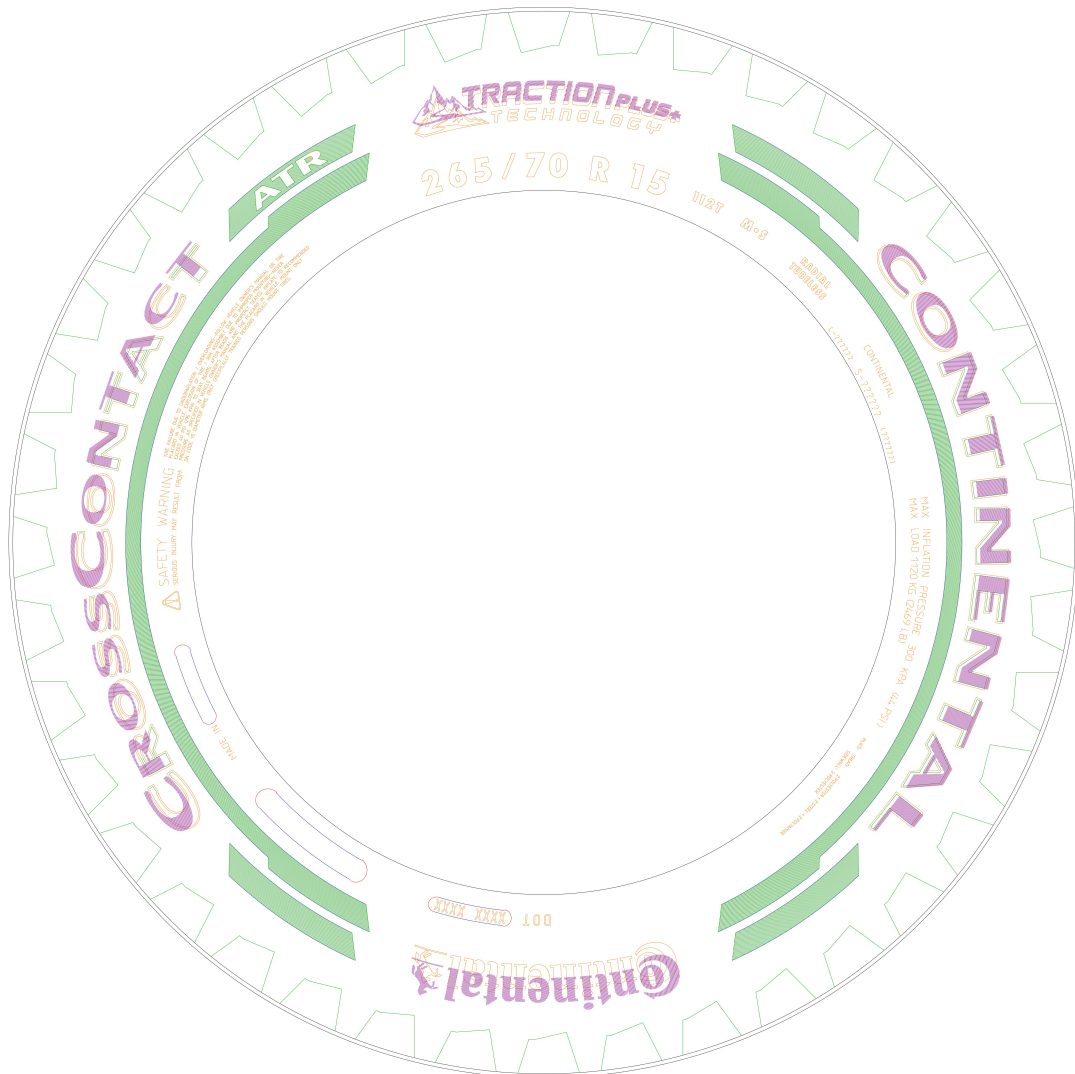


Figura 5.13: Resultado da imagem gerada de um modelo CAD que contém 86 entidades *CIRCLE*, a preto, 3363 *LINE*, a verde, 22 *ARC*, a azul, 12 *ELLIPSE*, a vermelho, 1343 *SPLINE*, a cor de laranja, e 49 *INSERT* contendo 3154 *LINE*, a roxo, do modelo CAD de um molde.

5.4 Resultados

Depois de percebido o método de representação gráfica de cada entidade do conjunto estudado, foram desenvolvidas funções para efetuar a transformação de referenciais e para representar cada tipo de entidade utilizando a biblioteca referida em 5.3.2.

No que diz respeito à representação da imagem como um todo existem alguns pontos a ressaltar. Em primeiro lugar, a janela da imagem é definida pelas dimensões da maior entidade do tipo *CIRCLE* encontrada. Isto deve-se ao simples facto de que todo e qualquer modelo CAD de um molde é sempre delimitado por uma circunferência, não havendo qualquer informação fora da

mesma. Em segundo lugar, é importante perceber que nem todas as entidades se incluem na representação da imagem gerada. Em primeira instância, todas as que apresentam o valor 1 no atributo que respeita a visibilidade, atributo *invisible*, não são tidas em conta. Entidades com este valor no atributo referido referem-se geralmente a curvas de auxílio no projeto do modelo CAD, como por exemplo linhas de guia. Em segunda instância, apenas as entidades cujo atributo referente ao tipo de linha, *linetype*, tem o valor de linha sólida (*Continuous*) são representadas. Este facto surge como conclusão da análise efetuada às imagens resultantes com este método, tendo-se percebido que outros tipos de linhas não acrescentam informação relevante.

De maneira a comprovar a correta geração de imagem com o método proposto, na figura 5.14 observa-se um *print screen* do AutoCAD no qual se verifica um excerto do modelo da figura 5.13.

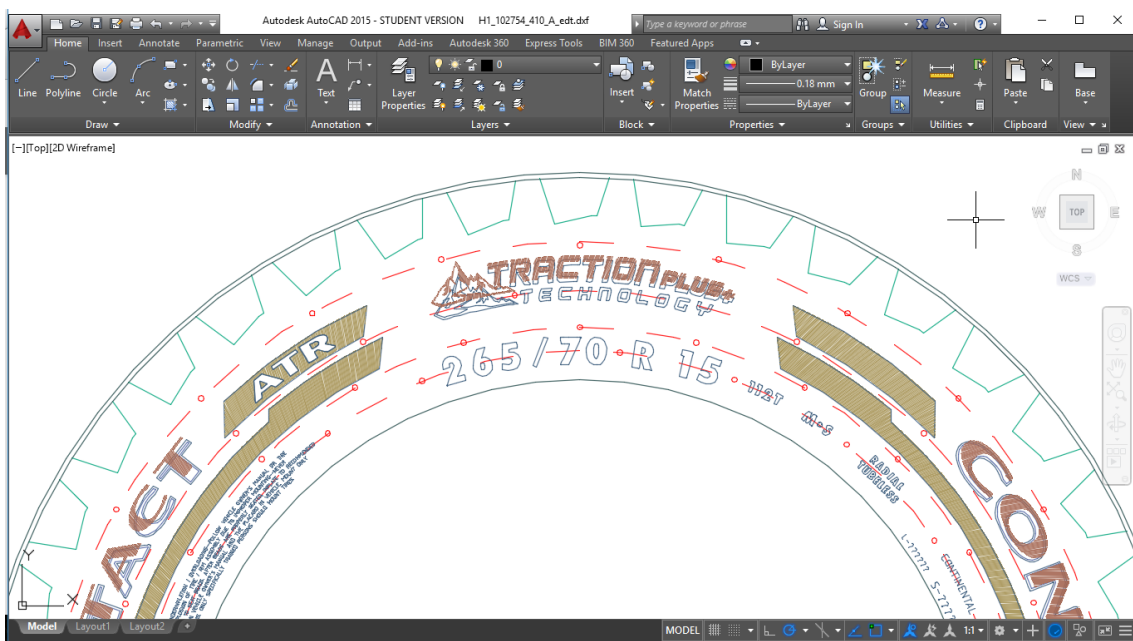


Figura 5.14: *Print screen* do AutoCAD para comprovar a correta geração de imagem do modelo da figura 5.13.

Por fim, ilustra-se nas figuras 5.15, 5.16 e 5.17 as imagens geradas resultantes da interpretação de modelos CAD de outros moldes.

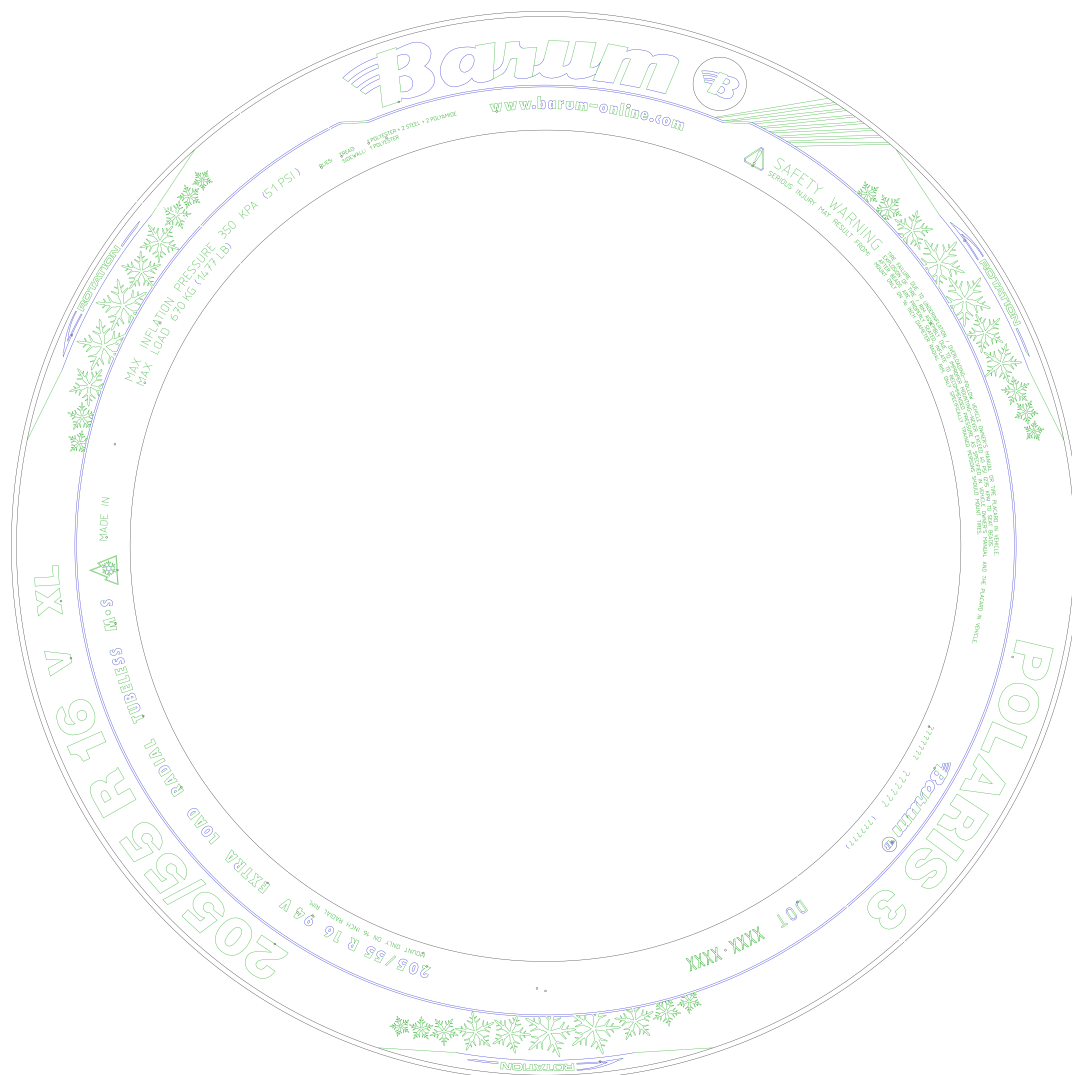


Figura 5.15: Resultado da imagem gerada de um modelo CAD que contém 60 entidades *CIRCLE*, a preto, 10372 *LINE*, a verde, e 708 *ARC*, a azul.

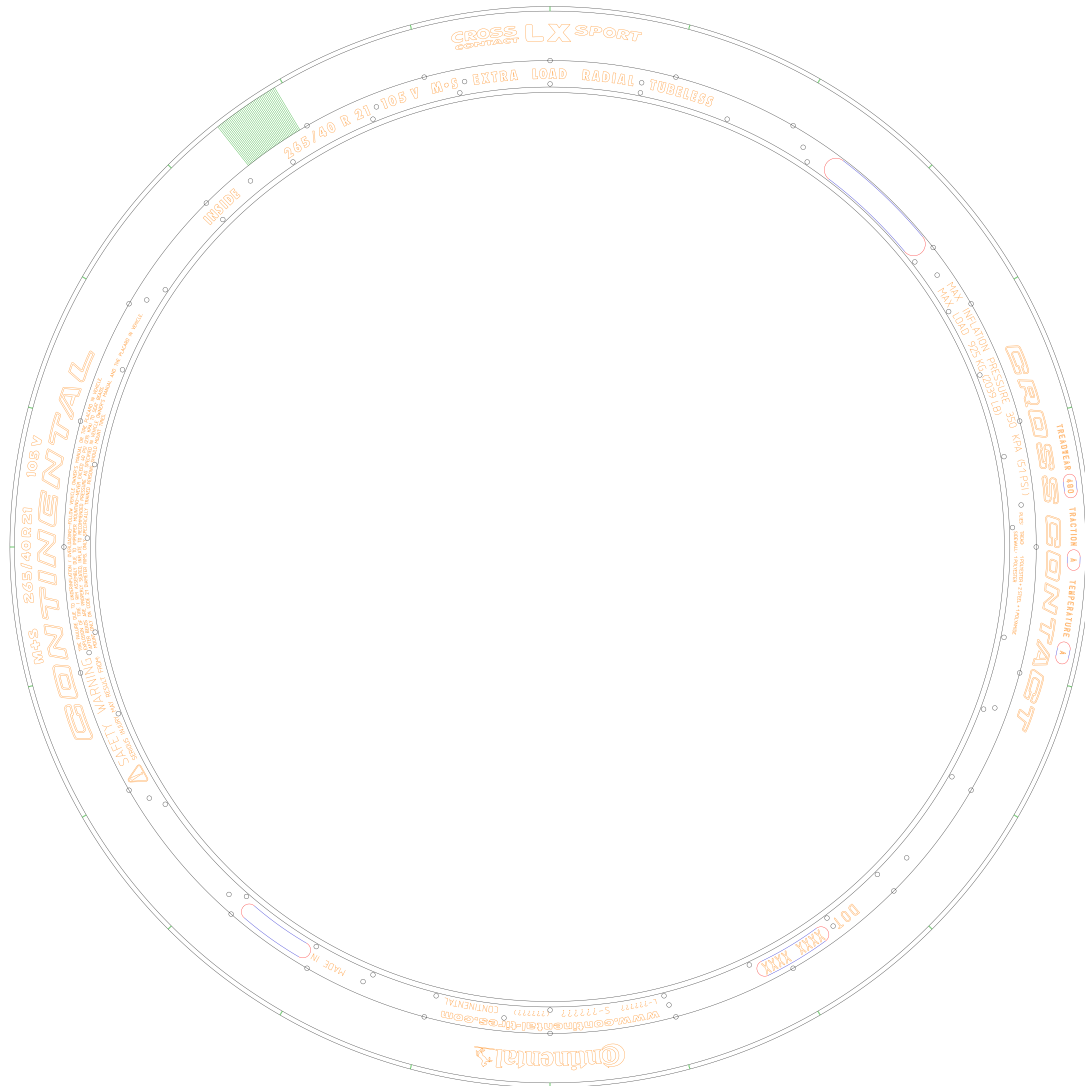


Figura 5.16: Resultado da imagem gerada de um modelo CAD que contém 81 entidades *CIRCLE*, a preto, 103 *LINE*, a verde, 14 *ARC*, a azul, 24 *ELLIPSE*, a vermelho, e 1242 *SPLINE*, a cor de laranja.



Figura 5.17: Resultado da imagem gerada de um modelo CAD que contém 42 entidades *CIRCLE*, a preto, 1523 *LINE*, a verde, 36 *ARC*, a azul, 20 *ELLIPSE*, a vermelho, e 1402 *SPLINE*, a cor de laranja.

5.5 Conclusões

O presente capítulo serviu para apresentar o método proposto nesta dissertação para gerar uma imagem a partir dos modelos CAD dos moldes fornecidos.

Em primeiro lugar importa referir que este ponto ocupou um tempo considerável de estudo. Isto aconteceu principalmente pelo facto de que a documentação relativa ao formato DXF não é muito clara no que toca a certas informações, como a questão da extrusão, tendo sido um entrave ao desenvolvimento deste capítulo de trabalho.

Em segundo lugar, relativamente ao método proposto, é preciso ter em conta que será necessário atualizar o *software* desenvolvido no que concerne a introdução de novos tipos de entidades. Não obstante, a ideia a seguir será sempre a mesma: avaliar os atributos que indicam informação relativa à parametrização da entidade e proceder à sua representação em conjunto com todas as outras entidades. De notar que há atributos das entidades que nos casos de estudo não tiveram influência na sua representação mas que o poderão ter noutros casos. Efetivamente, foram considerados os atributos cuja influência pudesse ser validada tendo em conta os casos disponíveis. Não obstante, as imagens geradas traduzem corretamente a representação dos modelos CAD.

Em terceiro lugar, é importante referir que o sucesso do método desenvolvido depende da forma como o modelo CAD é projetado. Para a representação das entidades da maneira que foi abordada neste capítulo, apenas interessa o modelo do molde concretamente. Mais ainda, apenas interessa uma vista superior de um modelo 2D do mesmo. Todavia, alguns modelos CAD fornecidos no decorrer deste projeto apresentam informação além da necessária para gerar a imagem. Esta informação é na sua maioria uma legenda referente a dimensões e posições a ocupar de certas entidades. O problema desta questão prende-se com o facto de que a informação referida está definida para ser visível, sendo que, conseqüentemente, aparece representada na imagem gerada. Como se pode concluir, compromete-se a utilização da referida imagem nas próximas fases deste projeto.

Importa por fim salientar que o método desenvolvido funcionou com sucesso num número razoável de casos de teste, validando-se a sua utilização. Este sucesso é fruto principalmente de uma boa compreensão do formato DXF e das entidades que segundo ele se definem.

Capítulo 6

Transformação da Imagem Gerada Polar para Cartesiana

No anexo B da página 109 está descrito o método utilizado para efetuar a aquisição da imagem dos moldes. Como se pode verificar na figura B.11 desse anexo, com o referido método as imagens resultantes da aquisição têm forma retangular. Desta maneira, uma vez que as imagens geradas a partir do modelo CAD apresentam a forma anelar do molde, é necessário um processamento para a uniformização das imagens gerada e adquirida. A solução proposta é a de efetuar a transformação da imagem gerada anelar para uma de forma retangular. Para efeito de nomenclatura, denominam-se por imagens polar e cartesiana as imagens anelar e retangular (resultante da transformação), respetivamente. Esta designação advém do referencial de coordenadas escolhido para definir cada uma das imagens, como é apresentado seguidamente.

Estrutura-se assim este capítulo segundo cinco secções. Na secção 6.1 aborda-se o problema inicial da definição da região de interesse da imagem gerada. Segue-se a secção 6.2, na qual se explicita o método utilizado para determinar a largura da secção anelar, ou seja, a altura do pneu. A transformação da forma da imagem em concreto, aborda-se na secção 6.3. Por fim incluem-se os resultados obtidos na secção 6.4, aos quais se seguem as principais conclusões da temática deste capítulo, na secção 6.5.

6.1 Definição da região de interesse

A importância da definição da região de interesse (ROI¹) da imagem gerada prende-se com a necessidade da determinação das coordenadas do centro e dos valores dos raios interno e externo da secção anelar definida pelo molde. Estes valores são necessários, em pixels, para a concretização do objetivo de transformação da forma da referida imagem.

Como foi explicado na secção 5.4, a imagem gerada apresenta sempre uma linha de circunferência delimitadora. Desta forma, a ROI é definida pelo quadrado que envolve a referida circunferência, conceito conhecido, em inglês, por *bounding box*.

¹ROI - *Region Of Interest*

O método proposto para obtenção da ROI segue então uma ideia muito simples. É feita uma iteração ao longo dos pixels da imagem segundo os seus quatro lados, até se localizar um pixel que não é branco. Isto é, um pixel com valor de intensidade menor que 255. Nesse ponto ter-se-á atingido a circunferência exterior e define-se o respetivo lado da *bounding box* dado pela linha tangente à circunferência no referido ponto encontrado. Na figura 6.1 ilustra-se a representação do quadrado envolvente após determinação da ROI da imagem. Efetuando-se um corte na imagem segundo as quatro linhas representadas obtém-se a imagem que contém apenas a ROI.

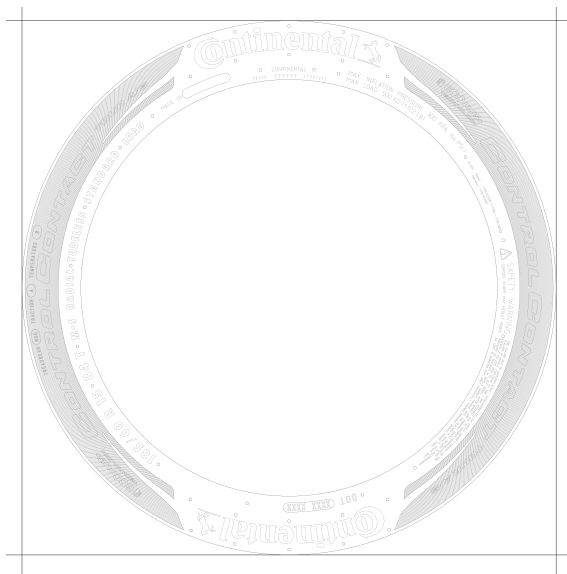


Figura 6.1: Exemplo de representação da *bounding box* da ROI da imagem gerada do modelo CAD de um molde.

6.2 Determinação da espessura da secção anelar

Depois de definida a ROI, a determinação da espessura, em pixels, da secção anelar que representa a parede do pneu é conseguida com uma operação de fácil implementação. Primeiramente determinam-se as coordenadas do centro da secção, sendo a abcissa dada por metade do valor da largura da imagem e a ordenada por metade da altura. Uma vez que se trata de uma imagem quadrada, os valores determinados são iguais. Seguidamente, itera-se a coluna de pixels com início no centro da secção anelar segundo a direção do eixo y. No ponto em que se detetar um pixel não branco estará a circunferência interior que define a secção. Isto acontece uma vez que a área da imagem interior a essa circunferência será sempre branca. Para clarificar esta ideia, identifica-se a referida coluna de pixels na imagem da figura 6.2.

Tendo sido obtido o raio interno da secção, o passo seguinte é a determinação do valor do raio externo. Novamente, uma vez que se trata de uma imagem quadrada que envolve a circunferência exterior da secção anelar, o exterior é dado por metade do valor da largura (ou altura) da imagem. Ainda que possa haver um erro associado aos referidos cálculos, este considera-se desprezável

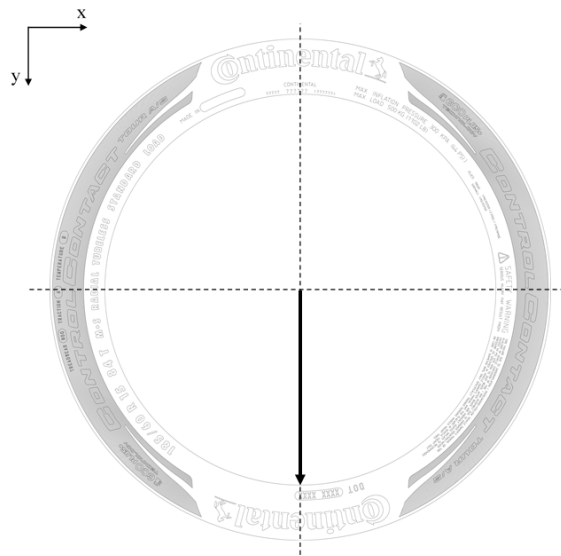


Figura 6.2: Identificação da coluna de pixels e direção de iteração da imagem para determinação do raio interior da secção anelar.

uma vez que não afeta o objetivo pretendido. Na figura 6.3 representam-se os limites interiores e exteriores da secção.

Por fim, o valor da espessura da secção anelar é dado pela diferença entre os raios externo e interno.

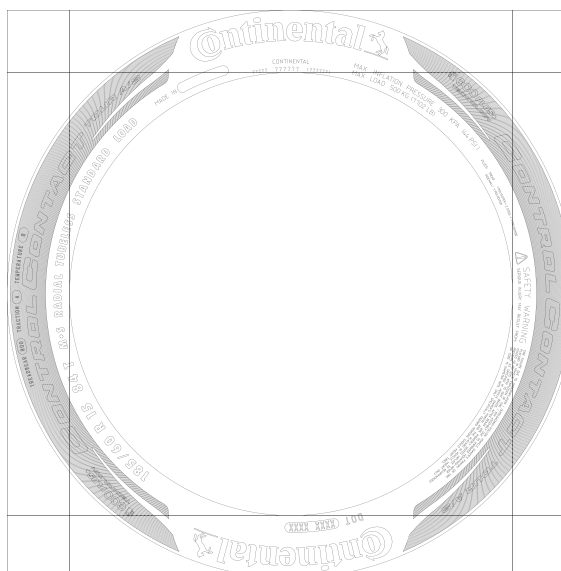


Figura 6.3: Representação dos limites interiores e exteriores da secção anelar, paralelos aos lados da imagem.

6.3 Transformação do referencial de imagem

A transformação da forma da imagem gerada é conseguida através de uma transformação no seu sistema de coordenadas. Assim, identifica-se primeiramente a imagem polar cuja designação advém da sua definição segundo um referencial de coordenadas polares com origem no centro da secção anelar. Esta imagem e respetivo referencial estão representados na figura 6.4.

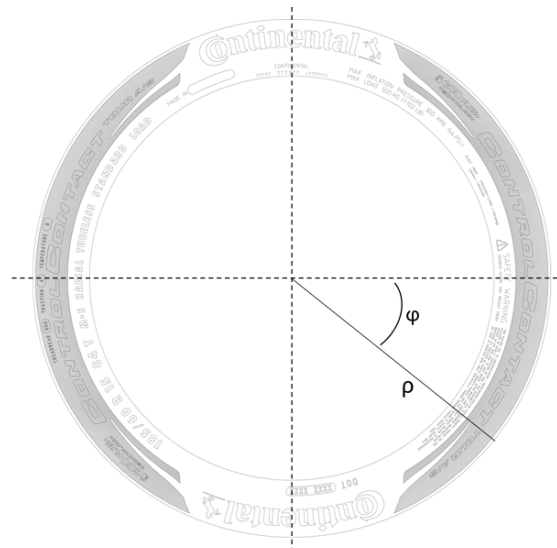


Figura 6.4: Ilustração da definição da imagem gerada através de um sistema de coordenadas polares.

O pressuposto da transformação referida é o de efetuar um varrimento dos pixeis pertencentes à secção anelar e converter as suas coordenadas polares para cartesianas. Para o efeito, é necessário definir um passo de varrimento segundo ρ e φ , cujas iterações correspondem a uma nova linha e coluna (eixos y e x cartesianos) de pixeis, respetivamente, da imagem cartesiana. Isto é, o número de passos efetuados segundo ρ e φ traduzem a altura e a largura da referida imagem.

A definição dos passos referidos é então o ponto chave do processo de transformação. O valor de cada passo tem de ser atribuído de forma a não se perder informação presente na imagem polar e a minimizar o efeito de distorção. Desta maneira, o passo angular (segundo φ) é determinado tendo em conta o seu menor valor possível. Este é dado pelo ângulo equivalente ao deslocamento de apenas um pixel para o lado na circunferência exterior, ou seja, nos pixeis mais afastados do centro. Na figura 6.5 ilustra-se este cenário para melhor compreensão. Observando a figura, o passo angular, designado por φ' , é dado por:

$$\varphi' = \arctan \frac{1}{r} \quad (6.1)$$

Sendo r o raio exterior determinado em 6.2.

Por sua vez, o valor definido passo segundo ρ define-se de maneira a que a altura da imagem cartesiana seja igual à espessura da secção anelar. Percebe-se facilmente o valor a atribuir tendo

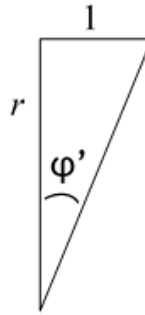


Figura 6.5: Ilustração da definição do passo angular de varrimento.

em conta as situações em que o varrimento está a ser efetuado nas linhas/colunas de pixels coincidentes com os eixos representados a tracejado na figura 6.4, ou seja, nas situações para as quais o varrimento tem a direção exata das linhas/colunas de pixels da imagem. Conclui-se então que para que haja um número de iterações radiais igual ao valor da espessura referida, o referido passo toma valor unitário.

Por fim, efetua-se o varrimento da imagem que se inicia no semieixo horizontal da direita a tracejado da figura 6.4 e é feito em sentido horário. Cada iteração deste processo começa com a determinação das coordenadas de imagem de cada pixel da secção anelar, de forma a se ter acesso ao seu valor de intensidade. Estas coordenadas, identificadas por (x_{img}, y_{img}) , são dadas pelas seguintes equações:

$$x_{img} = c_x + \rho \cdot \cos \varphi \quad (6.2)$$

$$y_{img} = c_y + \rho \cdot \sin \varphi \quad (6.3)$$

Em que (c_x, c_y) traduzem as coordenadas do centro da secção anelar.

A atribuição do valor de intensidade de cada pixel da imagem cartesiana em cada iteração do varrimento pode ser efetuado utilizando vários métodos. Primeiramente foi atribuído o valor a cada pixel pelo método do vizinho mais próximo, conhecido em inglês por *nearest neighbor*. Este método consiste em atribuir o valor de intensidade do pixel mais próximo do ponto dado por (x_{img}, y_{img}) , ou seja, efetua-se um arredondamento às unidades dos valores das coordenadas. Na figura 6.6 ilustra-se um exemplo da atribuição do valor de um pixel segundo este método.

Uma vez que como se pode observar na secção de resultados deste capítulo as imagens geradas com este método apresentam linhas com ruído, optou-se por implementar o método bilinear. Este pressupõe a atribuição do valor de intensidade a cada pixel com o cálculo da interpolação dos valores de intensidade dos quatro pixels mais próximos. Representa-se na figura 6.7 o caso da figura 6.6 na qual se identificam as coordenadas dos pixels próximos e as distâncias Δu e Δv do ponto ao pixel base (u, v) .

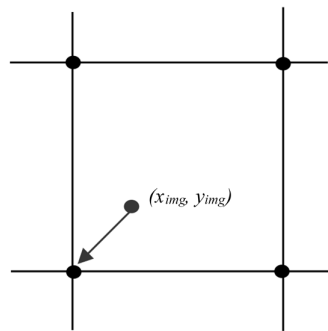


Figura 6.6: Atribuição do valor de intensidade a um pixel pelo método do vizinho mais próximo.

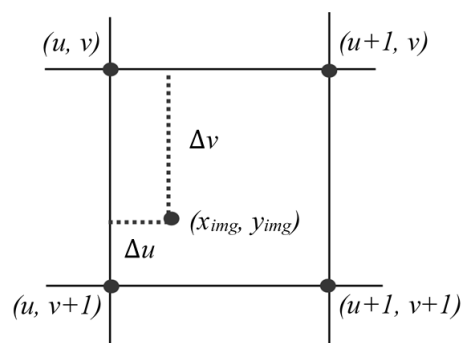


Figura 6.7: Atribuição do valor de intensidade a um pixel pelo método bilinear.

Concluindo, com este método o valor de intensidade de cada ponto é dado através da expressão 6.4.

$$\begin{aligned}
 I(x_{img}, y_{img}) = & (1 - \Delta u) \cdot (1 - \Delta v) \cdot I(u, v) \\
 & + \Delta u \cdot (1 - \Delta v) \cdot I(u + 1, v) \\
 & + (1 - \Delta u) \cdot \Delta v \cdot I(u, v + 1) \\
 & + \Delta u \cdot \Delta v \cdot I(u + 1, v + 1)
 \end{aligned} \tag{6.4}$$

6.4 Resultados

Após se ter explicitado o processo associado à transformação da forma da imagem gerada, serve esta secção para apresentação de alguns dos resultados obtidos.

Em primeiro lugar, ilustram-se as diferenças no resultado da atribuição do valor de intensidade de cada pixel da imagem cartesiana em função do método utilizado. No caso da figura 6.8 exemplifica-se o efeito no elemento textual *SERIOUS*, presente no campo de informação relativo ao aviso de segurança (subsecção 2.4.5, página 12), em cima pelo método do vizinho mais próximo e, em baixo, pela interpolação bilinear. Por sua vez, na figura 6.9 observa-se novamente a

influência do método do vizinho mais próximo, em cima, e da interpolação bilinear, em baixo, no logótipo presente no modelo CAD da figura 5.15 (página 57).



Figura 6.8: Diferença entre os métodos de atribuição de intensidade a cada pixel num elemento textual: em cima, vizinho mais próximo e, em baixo, bilinear.

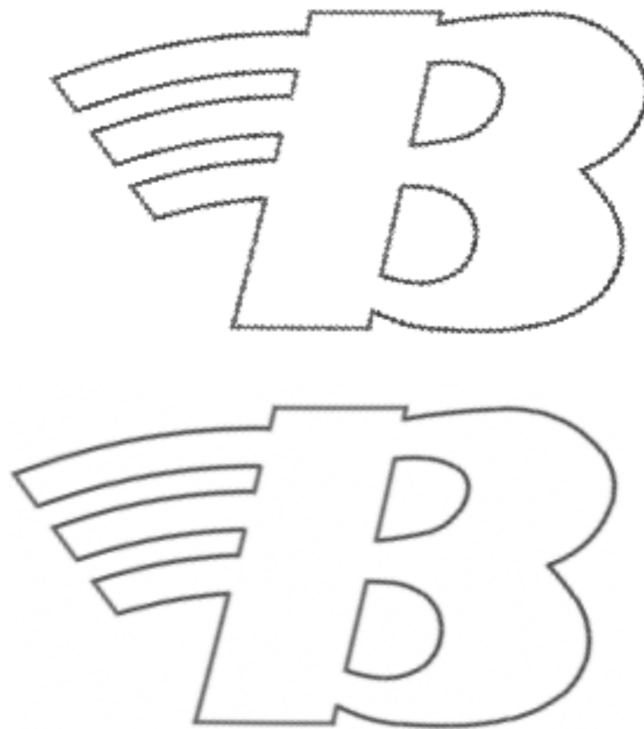


Figura 6.9: Diferença entre os métodos de atribuição de intensidade a cada pixel num logótipo: em cima, vizinho mais próximo e, em baixo, bilinear.

Por fim, observam-se na figura 6.10 excertos das imagens cartesianas resultantes da transformação das imagens polares dos modelos CAD das figuras 5.13, 5.15, 5.16 e 5.17.

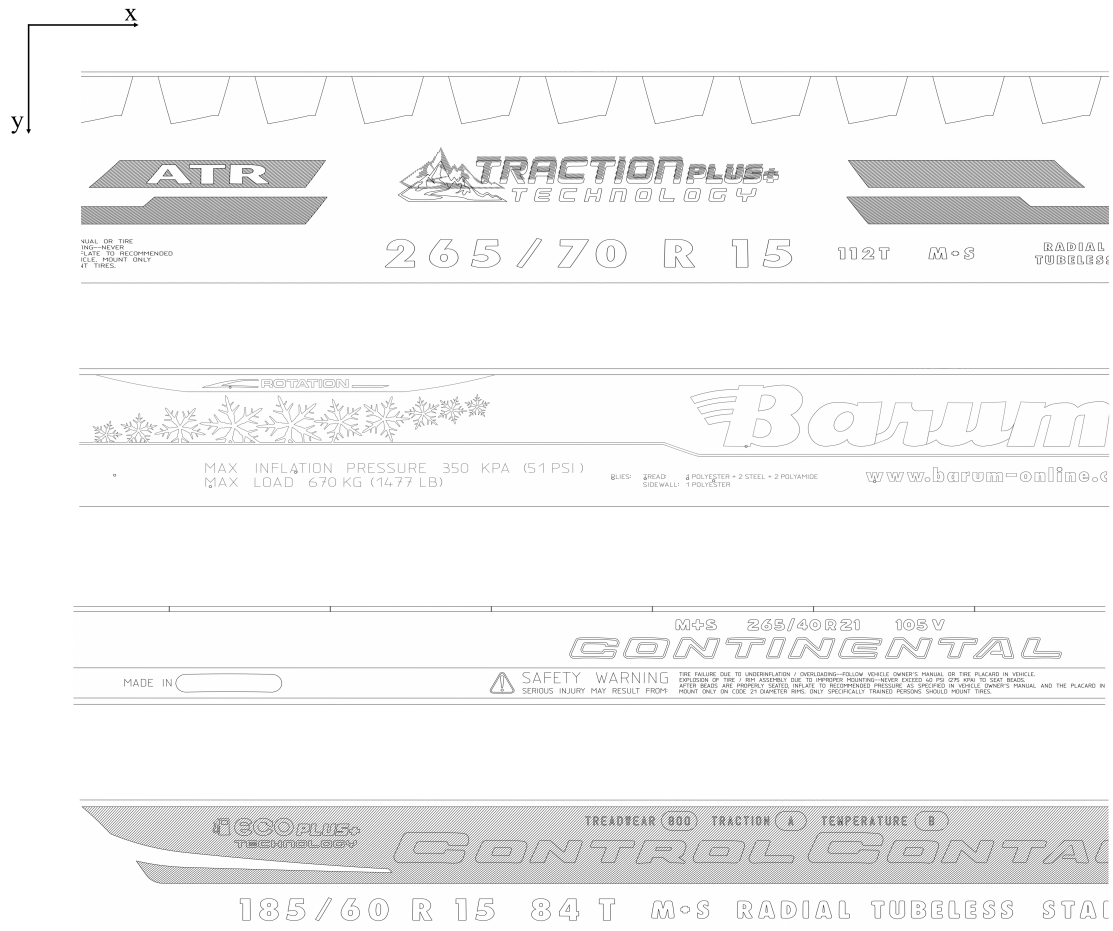


Figura 6.10: Excerto das imagens cartesianas resultantes dos modelos CAD das figuras 5.13, 5.15, 5.16 e 5.17, respectivamente de cima para baixo.

6.5 Conclusões

Como principal conclusão retirada do processo de transformação de imagem explicitado neste capítulo, é importante referir o efeito dos passos de varrimento no resultado final.

Importa perceber que ao se efetuar a transformação da imagem polar para cartesiana, parte da imagem será esticada ou encolhida. Exemplifica-se esta questão com as circunferências externa e interna que delimitam a secção anelar e que têm, conseqüentemente, diferentes perímetros. Depois da transformação mencionada, estas são as linhas que delimitam a imagem retangular em cima e em baixo e que têm, portanto, o mesmo comprimento. Desta maneira, uma vez que há sempre um erro associado a este tipo de transformação, os passos foram escolhidos de forma a que este fosse pequeno.

A título ilustrativo, na figura 6.7, observa-se o efeito da variação do passo angular num excerto de uma imagem cartesiana final.

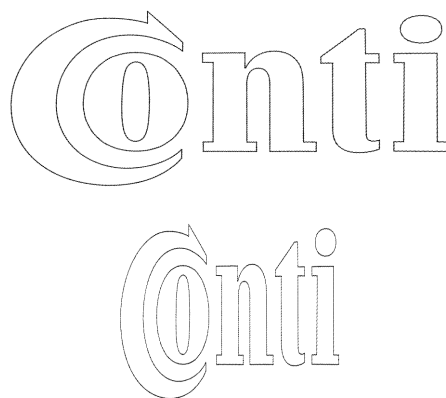


Figura 6.11: Efeito do passo angular na imagem retangular resultante.

Outro ponto ainda a referir é o método utilizado na atribuição do valor de intensidade a cada pixel da imagem cartesiana. Efetivamente, como se observa nas figuras 6.8 e 6.9 utilizando o método de interpolação bilinear conseguem-se imagens com linhas mais bem definidas.

Por fim, depois de obtida cada imagem cartesiana há ainda uma questão a ter em conta. Na transformação de forma de uma imagem, como foi referido, o varrimento é efetuado em sentido horário com início no semieixo horizontal direito que passa no centro da secção anelar. Conseqüentemente, tomando como exemplo a imagem obtida a partir da figura 5.17 da página 59, tem-se uma quebra da informação textual situada na zona de início de varrimento. De forma a evitar este tipo de quebras, acrescenta-se no início e no fim de cada imagem um excerto correspondente da mesma. Desta maneira, garante-se que todos os campos de informação aparecem sempre completos.

Capítulo 7

Sincronização das Imagens Gerada e Adquirida

Chegada esta etapa do *software acbs*, têm-se as imagens resultantes da aquisição do molde (anexo B) e da geração a partir do modelo CAD (capítulo 5), ambas em forma retangular (capítulo 6). Não obstante, para efetuar a comparação das referidas imagens é ainda necessário realizar um processo de sincronização das mesmas. Para o efeito, definem-se uma série de passos a seguir.

Em primeiro lugar, é fundamental efetuar o redimensionamento das imagens de maneira a que estas apresentem uma escala o mais semelhante possível. Em segundo lugar, importa definir a região de interesse da imagem adquirida. A necessidade deste passo justifica-se com duas razões: em primeira instância, uma vez que a referida imagem não apresenta apenas a secção do molde relativa à parede do pneu, consequência da área abrangida pelo campo de visão da camara; em segunda instância, pelo motivo de que é difícil definir os pontos inicial e final da aquisição, resultando uma imagem que não corresponde apenas aos 360° da secção anelar do molde, havendo assim zonas repetidas a excluir. Em terceiro lugar, a nível de conteúdo das imagens, pretende-se que o início e o final seja o mesmo em ambas. Torna-se assim imperativo o rearranjo de uma das imagens, sendo este efetuado na imagem gerada.

A realização deste processo de sincronização tendo em conta os passos mencionados implica a identificação da diferença da escala e do desfasamento das imagens gerada e adquirida. A solução proposta para esta questão assenta no princípio de localizar os campos de informação referentes à informação de fabrico (subsecção 2.4.3) e ao aviso de segurança (subsecção 2.4.5) em ambas as imagens. Com efeito, localizando-se as áreas correspondentes a essas informações e tendo em conta as dimensões de cada uma, torna-se possível identificar a informação pretendida. Foram escolhidos estes campos de informação pela razão de que aparecem em todo e qualquer molde e sempre com o mesmo aspeto.

Apresentam-se no seguimento deste capítulo todos procedimentos associados ao processo de sincronização, numa estrutura de seis secções. Na secção 7.1 aborda-se a definição das designadas zonas de sincronismo, com a localização dos referidos campos de informação nas imagens. Segue-se a secção 7.2, na qual se explica o método utilizado no ajuste da escala das referidas

imagens. A seleção da região de interesse da imagem adquirida inclui-se na secção 7.3. Por outro lado, a secção 7.4 dedica-se ao rearranjo da imagem gerada, de forma a concretizar a sincronização. Finaliza-se com a apresentação de resultados obtidos na secção 7.5 e com algumas notas conclusivas na secção 7.6.

7.1 Definição das zonas de sincronismo

O processo de sincronização das imagens gerada e adquirida inicia-se com a localização de campos de informação enquanto elementos comuns entre as mesmas. Para o efeito, pensou-se primeiramente em aplicar a técnica de *template matching* uma vez que se trata de um método utilizado para localizar uma imagem *template* numa imagem maior. Desta maneira, criando *templates* para cada campo de informação seria possível obter a sua localização nas imagens.

A eficácia da técnica de *template matching* não é todavia invariante à escala. Isto é, o seu funcionamento correto pressupõe que o *template* apareça na imagem com uma escala muito próxima daquela que o mesmo apresenta. De forma a solucionar esta questão, a referida técnica é aplicada tendo em conta várias escalas da imagem, ou seja, aplica-se o *template matching* num processo iterativo sendo em cada iteração redimensionada a imagem para uma escala ligeiramente diferente.

Por outro lado, esta técnica pressupõe ainda que a zona a localizar apresente uma orientação bastante semelhante à do *template*. Ainda que nas imagens geradas a orientação seja sempre a mesma, não havendo qualquer problema nesse aspeto, relativamente às imagens adquiridas verifica-se que as zonas a localizar podem aparecer inclinadas, colocando em causa o sucesso da aplicação desta técnica. Para superar esta questão, a solução proposta é a de complementar a utilização da técnica de localização baseada num *template* com uma outra técnica baseada em *features*, designada por *feature matching*. Com efeito, o processo de localização em cada imagem adquirida passa por obter zonas de localização candidatas com *template matching* e, posteriormente, avaliar cada uma dessas zonas obtidas e validar a correta com *feature matching*, garantindo-se a correta localização dos *templates*.

De forma a denotar os conceitos envolvidos no método proposto, dedica-se uma subsecção a cada uma das técnicas referidas, bem como à necessidade e processo de seleção da região de interesse de cada zona após a sua localização.

7.1.1 *Template Matching*

O princípio de funcionamento da técnica de *template matching* baseia-se em deslizar a imagem *template* sobre a imagem maior, comparando-a com todas as posições possíveis ocupadas sobre a última. No presente caso de aplicação, as imagens *template* são obtidas efetuando o corte das zonas correspondentes aos campos da informação de fabrico e de aviso de segurança de uma imagem gerada e de uma adquirida, criando assim *templates* para as respetivas imagens. Observam-se na figura 7.1 os *templates* obtidos.



Figura 7.1: Imagens *template* utilizadas para as imagens gerada e adquirida, de cima para baixo, e para o campo da informação de fabrico e do aviso de segurança, da esquerda para a direita.

Uma vez obtidas as imagens *template*, a deteção da sua localização é conseguida através da utilização da função *matchTemplate* da biblioteca *OpenCV*. Esta função tem como parâmetros de entrada: o *template*, a imagem na qual se pretende localizá-lo e o método de comparação das imagens. Entre os vários métodos existentes, foi utilizado o *TM_CCOEFF_NORMED*. A sua fórmula é dada por:

$$R(x, y) = \frac{\sum_{x', y'} (T'(x', y') \cdot I'(x + x', y + y'))}{\sqrt{\sum_{x', y'} T'(x', y')^2 \cdot \sum_{x', y'} I'(x + x', y + y')^2}} \quad (7.1)$$

Sendo:

$$T'(x', y') = T(x', y') - \frac{\sum_{x'', y''} T(x'', y'')}{w \cdot h} \quad (7.2)$$

$$I'(x + x', y + y') = I(x + x', y + y') - \frac{\sum_{x'', y''} I(x + x'', y + y'')}{w \cdot h} \quad (7.3)$$

Em que I corresponde à imagem e T , w e h ao *template* e à sua largura e altura, respetivamente. Desta maneira, através do cálculo do valor resultante da equação 7.1 para cada posição do *template* sobre a imagem, define-se a matriz R , que é a saída da função mencionada. Cada elemento desta matriz traduz assim a métrica de correspondência entre as imagens em comparação para cada posição diferente. Estabelecendo as dimensões da imagem maior como (W, H) e as do *template* como (w, h) , esta matriz terá como dimensões $(W-w+1, H-h+1)$. Aplica-se posteriormente a função *minMaxLoc*, igualmente da biblioteca referida, de forma a extrair a posição da melhor correspondência entre as imagens. Para o método de comparação utilizado, essa posição ocorre para o valor máximo da matriz R .

Nas imagens geradas, a localização de cada *template* é dada pelo valor máximo entre os máximos de R para cada escala. Todavia, em relação às imagens adquiridas esse processo é insuficiente. Nessas imagens a aplicação desta técnica destina-se a, como mencionado anteriormente, determinar zonas de localização candidatas do *template*. Com efeito, a nível de algoritmo implementado, é guardada cada posição correspondente ao valor máximo de R para cada dimensão diferente da

imagem maior. Desta maneira, averigua-se que a zona de localização esperada é corretamente detetada como candidata em todos os casos de teste. O redimensionamento das imagens tem como valor máximo 1.5 vezes o seu tamanho original e metade do mesmo como mínimo, sendo avaliadas 50 escalas igualmente espaçadas desse intervalo.

Por fim, observa-se nas figuras 7.2 e 7.3 um exemplo da identificação dos *templates* da figura 7.1, desenhando-se um retângulo a azul que define a zona identificada como localização dos mesmos na imagem gerada, e a vermelho as localizações candidatas na imagem adquirida.



Figura 7.2: Localização, a azul, dos *templates* numa imagem gerada.

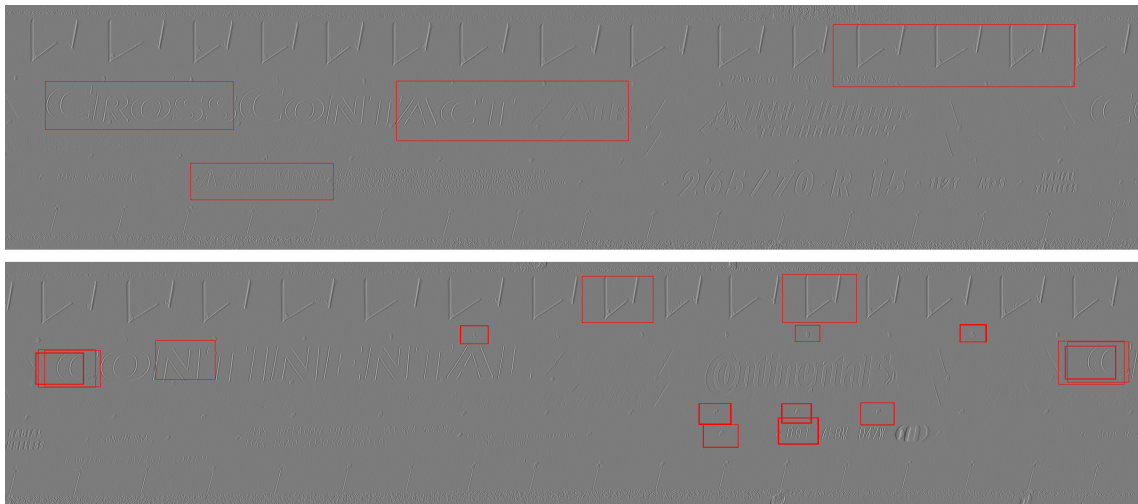


Figura 7.3: Localizações candidatas, a vermelho, dos *templates* numa imagem adquirida.

7.1.2 Feature Matching

Feature matching foi a técnica escolhida para efetuar a validação da zona de localização do *template* numa imagem adquirida, após se terem definido as zonas candidatas. Como foi referido anteriormente, o facto de a imagem adquirida poder aparecer com alguma rotação relativamente ao *template* traduz-se em resultados errados com a aplicação de *template matching*. Por sua vez, a presente técnica baseada em *features* é invariante a fatores de escala e rotação, não sendo suscetível aos problemas da técnica anterior.

Desta maneira, são primeiramente detetadas as *features* na imagem *template* e na zona candidata em análise. Esta deteção é conseguida através da função *detectAndCompute*, novamente da biblioteca *OpenCV*, tendo sido utilizados descritores do tipo *AKAZE*. Ilustram-se na figura 7.4 as *features* detetadas em cada *template* e zona candidata correta correspondente.

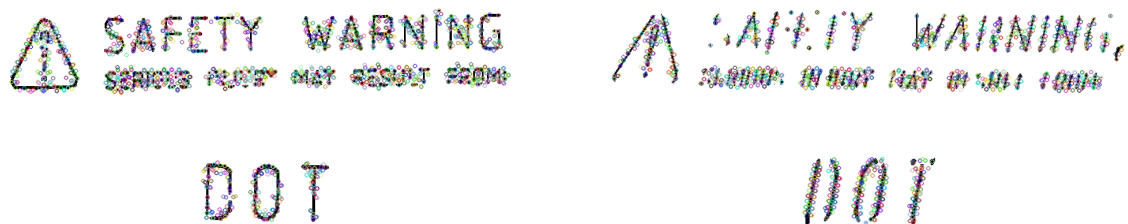


Figura 7.4: *Features* detetadas em cada *template*, à esquerda, e na zona candidata correta respetiva, à direita, com descritores *AKAZE*.

Determinam-se seguidamente para cada zona, as correspondências entre as *features* do *template* e da zona candidata, utilizando o *BFMatcher*¹. Este método avalia cada *feature* do *template* e determina a melhor correspondência entre todas as *features* da imagem. Além disso, o referido

¹*BFMatcher* - *Brute-Force Matcher*

método é utilizado com *crossCheck*, ou seja, são retornadas apenas as correspondências cuja melhor correspondência do *template* para a imagem em análise é também a melhor dessa imagem para o *template*. Isto é, são selecionadas as *features* das duas imagens que sejam reciprocamente correspondentes.

Não obstante, algumas das correspondências encontradas poderão não traduzir uma correspondência correta, pondo em causa a veracidade do resultado. Para solucionar esta questão é utilizada a função *findHomography*, igualmente pertencente à *OpenCV*, que retorna a matriz de transformação do conjunto de pontos das *features* do *template* para o das *features* da imagem em análise. Para o efeito, esta função utiliza o algoritmo *RANSAC*² que permite assim remover as correspondências que não representam boas estimações (*outliers*), mantendo as correspondências bem estimadas (*inliers*). Observam-se na figura 7.5 as correspondências entre as imagens da figura 7.4, após serem submetidas ao processo descrito.

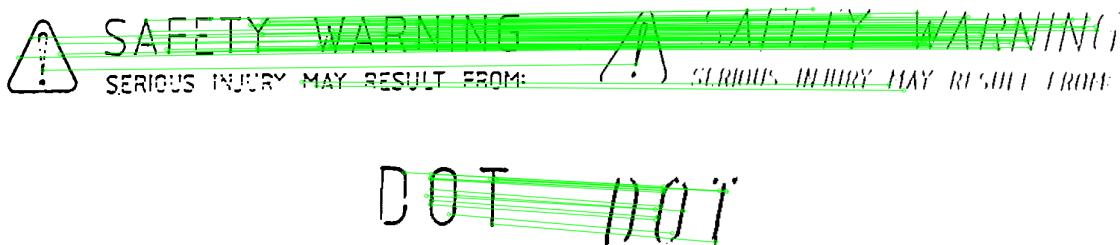


Figura 7.5: Aplicação da técnica de *feature matching* às imagens da figura 7.4.

Finalmente, a validação da zona candidata correta é feita tendo em conta aquela para qual é máximo o número de correspondências. Na figura 7.6 definem-se, a azul, as localizações validadas com o método descrito.

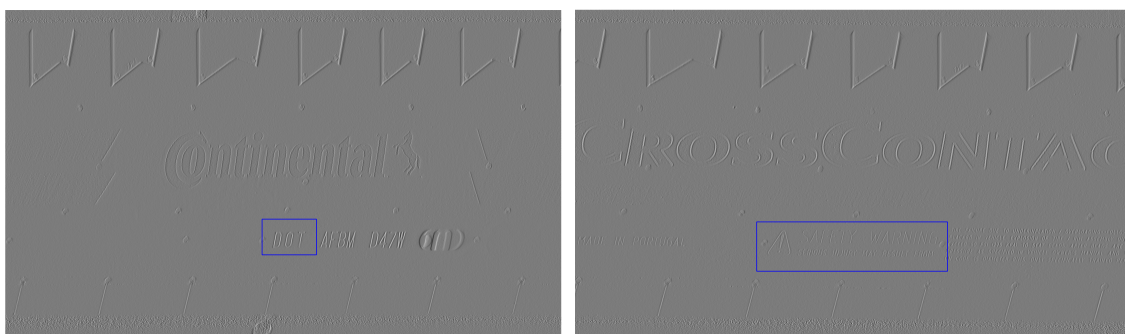


Figura 7.6: Identificação, a azul, da zona validada da localização dos *templates* numa imagem adquirida com a técnica de *feature matching*.

²RANSAC - *Random Sample Consensus*

7.1.3 Definição da região de interesse

Como se pode observar nas figuras 7.2 e 7.6 a zona detetada nas imagens para localização do *template* engloba uma área que não é exclusivamente a informação presente em cada um. Ainda que cada *template* correspondesse apenas à informação, isto é, não tivesse margens, esta questão continuaria a não ser controlável. A área detetada é, efetivamente, resultado dos pontos retornados pela função referida em 7.1.1.

É importante contornar esta questão uma vez que é imperioso ter bem definida a zona onde a informação de cada *template* ocorre na imagem. Como se verá de seguida, o desenvolvimento das próximas etapas de sincronização das imagens gerada e adquirida depende disso. A solução proposta é então a de definir a região de interesse das imagens dadas pelo retângulo de deteção. Isto é, a ideia apresentada é a de definir uma *bounding box* à volta da informação em cada imagem resultante da deteção.

Apresentam-se nos pontos seguintes os métodos implementados para definir a ROI da informação de fabrico, em 7.1.3.1 e do aviso de segurança, em 7.1.3.2. No final, após aplicar os métodos implementados, obtêm-se as novas zonas de deteção ilustradas nas imagens da figura 7.7.

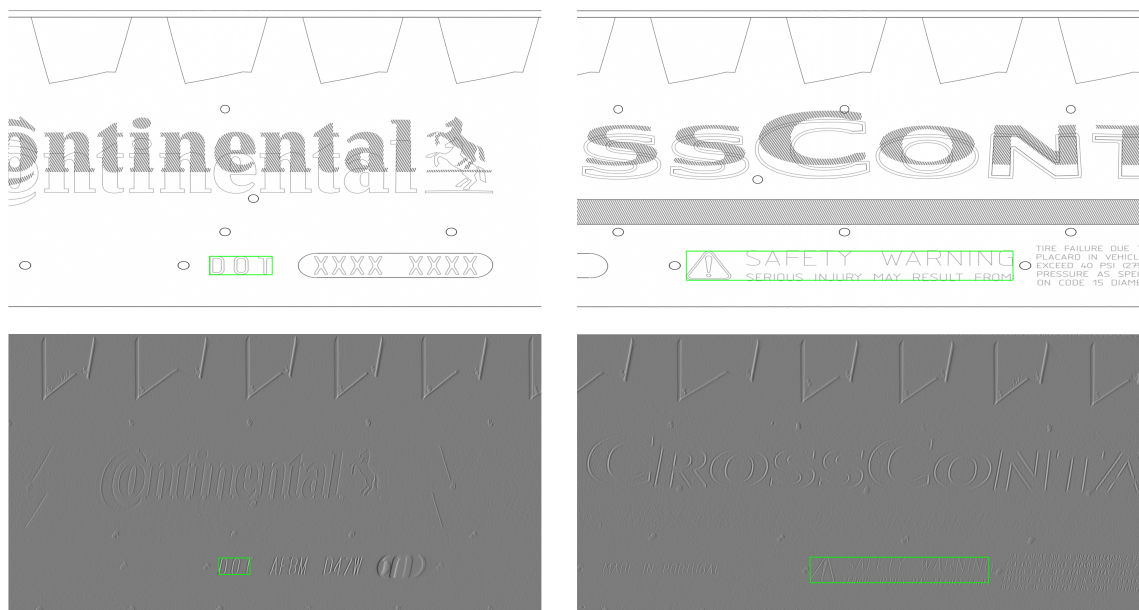


Figura 7.7: Identificação, a verde, da localização dos *templates* após definição da região de interesse.

7.1.3.1 Região de interesse na localização da informação de fabrico

A ideia subjacente à definição da ROI da informação de fabrico é a de determinar a *bounding box* da sigla DOT.

Em primeiro lugar, efetua-se a binarização da imagem. Como resultado, tem-se a branco a informação a detetar, passível de algum ruído, e o fundo a preto. Para as imagens resultantes da

deteção em imagens adquiridas é ainda realizada uma operação de fecho, de maneira a aumentar a parte branca da imagem e compensar o resultado da binarização.

Posteriormente, utilizando a função *findContours* da *OpenCV*, determinam-se os contornos da imagem binária. Do conjunto de contornos encontrados, determina-se o raio máximo entre as circunferências envolventes dos mesmos, obtidas pela função *minEnclosingCircle* da mesma biblioteca.

De maneira a eliminar possível ruído, que assumidamente terá pequenas dimensões comparativamente à sigla a detetar, consideram-se apenas os contornos cujo raio da circunferência envolvente é maior que metade do raio máximo encontrado. Seguidamente, refina-se ainda o conjunto encontrado, tendo apenas em conta os contornos que aparecem relativamente próximos.

Por fim, com o conjunto final, definem-se as abcissas inicial e final da ROI com os contornos mais à esquerda e direita, respetivamente. Desses contornos, aquele que apresentar uma maior altura, define o intervalo das ordenadas da região a definir.

Ilustram-se na figura 7.8 os passos seguidos na determinação da região de interesse pretendida.

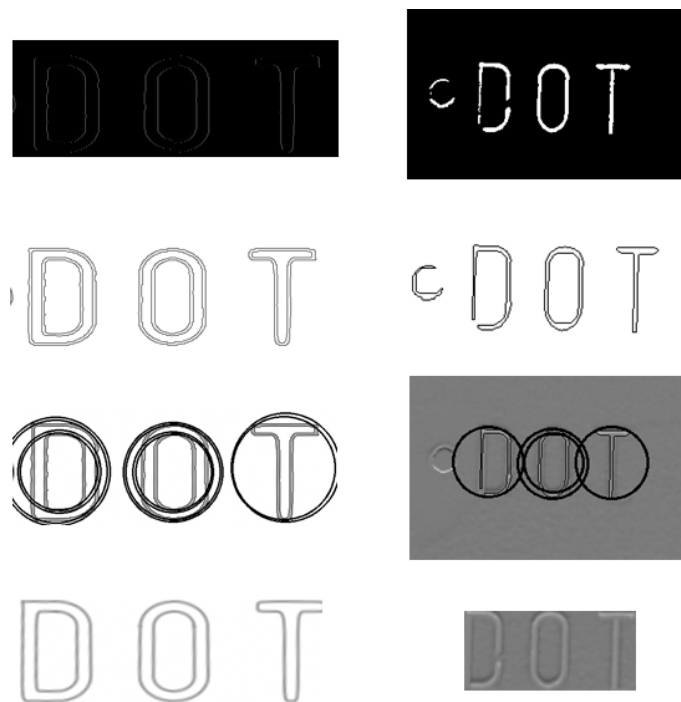


Figura 7.8: Passos seguidos na determinação da região de interesse da sigla DOT da imagem gerada, à esquerda, e da adquirida, à direita. De cima para baixo: imagens binárias; contornos detetados; circunferências envolventes dos contornos maiores; e, por fim, região de interesse obtida.

7.1.3.2 Região de interesse na localização do aviso de segurança

A definição da região de interesse do aviso de segurança, enquanto área que engloba apenas a informação contida no *template* correspondente, segue um processo semelhante ao do ponto anterior.

Com efeito, começa-se igualmente por binarizar a imagem, detetar os contornos na imagem resultante e determinar as circunferências envolventes. Analisando a informação presente nestas imagens, uma vez que aparece sempre o triângulo com o ponto de exclamação, é correto considerar que os contornos a ele relativos correspondem às maiores circunferências determinadas.

Define-se então o conjunto de contornos cuja circunferência envolvente tenha um diâmetro maior que metade do diâmetro máximo. Neste conjunto, o contorno mais à esquerda, (relativo ao triângulo) define a abcissa inicial e ordenadas inicial e final da ROI. Por sua vez, o contorno mais à direita define a abcissa final da ROI.

Na figura 7.9 observam-se os passos seguidos na determinação da região de interesse apresentada.



Figura 7.9: Passos seguidos na determinação da região de interesse do aviso de segurança da imagem gerada, à esquerda, e da adquirida, à direita. De cima para baixo: imagens binárias; contornos detetados; circunferência envolvente do maior contorno à esquerda; circunferências envolventes dos contornos maiores; e, por fim, região de interesse obtida.

7.2 Escalamento das imagens

Após se terem localizado as imagens *template* e se ter bem definida a informação nas imagens gerada e adquirida, o passo seguinte do processo de sincronização é o escalamento das mesmas. Esta tarefa divide-se em duas partes: o escalamento horizontal e vertical. Todavia, é necessária uma intervenção prévia na imagem gerada.

Como foi referido na secção 6.5 da página 69, de forma a evitar quebras de informação são acrescentadas partes à imagem gerada repetido-se as partes inicial e final. A imagem obtida não compreende assim apenas os 360° da secção anelar do molde, havendo necessidade de seleccionar

a zona a ter em conta. Desta maneira, tendo em conta cada excerto acrescentado anteriormente, elimina-se o correspondente a dois excertos no final da imagem. Por fim, verifica-se ainda se há algum *template* localizado na área eliminada. Em caso afirmativo, uma vez que se trata de informação repetida, antecipa-se a localização da zona detetada para o correspondente à largura da imagem exclusiva dos 360°, já que nessa área haverá outra ocorrência desse mesmo *template*.

Relativamente ao escalamento horizontal a ideia proposta é a de determinar a distância entre a localização dos *templates* e efetuar o redimensionamento da imagem com maior valor, de tal maneira que essa distância fique igual em ambas. Há no entanto um fator a ter em conta. Uma vez que não se tem conhecimento relativo à área da imagem adquirida que aparece repetida, a distância entre os *templates* terá de ser sempre determinada pela ordem em que aparecem na referida imagem. Apenas desta forma se garante que não se está a ter em conta área repetida. Finalmente, uma vez determinadas as distâncias em ambas as imagens, o fator de escalamento horizontal é dado pela razão entre esses valores obtidos.

O escalamento vertical assenta por sua vez no princípio de obter o valor da altura da zona da localização do aviso de segurança e redimensionar a imagem cujo valor é maior, de forma a igualar essa altura em ambas as imagens. Assim, o fator de escalamento vertical é dado pela razão entre as alturas das zonas mencionadas.

Concluindo, ilustra-se na figura 7.10 um exemplo de uma imagem gerada e uma adquirida nas quais se identifica a distância entre os *templates* (setas horizontais) e a altura da zona do aviso de segurança (setas verticais). No final deste processo, tendo-se igualado a medida das setas em ambas as imagens, obtêm-se as imagens com uma escala satisfatoriamente aproximada.

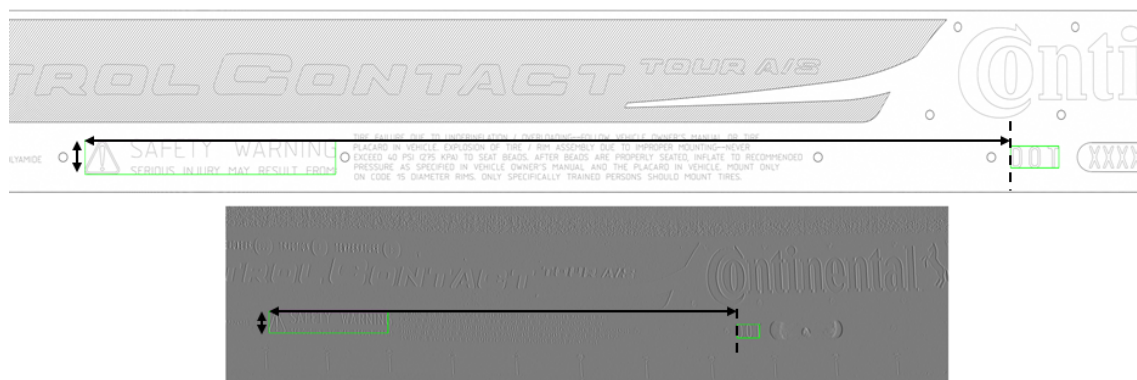


Figura 7.10: Ilustração das medidas utilizadas no escalamento das imagens: distância entre o aviso de segurança e a informação de fabrico, dada pelas setas horizontais; e altura do aviso de segurança, pelas setas verticais.

7.3 Seleção da região de interesse da imagem adquirida

Como foi referido no início do presente capítulo, ao contrário da imagem gerada, a imagem adquirida não contém apenas a parede do pneu em termos de altura. Além disso, também já

mencionada, a imagem adquirida não inclui apenas os 360° do molde a nível de largura, isto é, há uma área da imagem que aparece repetida. Com efeito, dedica-se este ponto à exposição da forma como é feita a selecção da secção da imagem adquirida correspondente apenas à parede do pneu.

Uma vez que a este ponto de desenvolvimento as imagens apresentam a mesma escala, a altura da secção a extrair da imagem adquirida terá de ser igual à altura da imagem gerada. Desta maneira, tendo em conta a localização do aviso de segurança na imagem gerada, determinam-se as alturas das áreas da imagem que há acima e abaixo dessa zona. A ROI vertical é então seleccionada de tal maneira que acima e abaixo dessa localização na imagem adquirida seja igual. Ilustra-se na figura 7.11 a ideia de selecção da área da imagem relativa à parede do pneu.



Figura 7.11: Ilustração do processo de selecção da região de interesse vertical da imagem adquirida.

Por fim, partindo novamente do facto de que as imagens têm a mesma escala, a selecção da ROI horizontal é um processo trivial. Efetivamente, basta seleccionar a área da imagem adquirida de forma a que esta fique com a mesma largura da imagem gerada, ficando assim com informação referente apenas a uma repetição do molde.

7.4 Rearranjo da imagem gerada

De maneira a concretizar o processo de sincronização é necessário ainda intervir a nível do conteúdo das imagens. Assim, serve este ponto para descrever como é efetuada a sincronização horizontal das mesmas.

À semelhança do ponto anterior, este passo é conseguido tendo em conta a localização do aviso de segurança. Primeiramente, tendo conhecimento das coordenadas da referida zona determina-se a distância entre as abcissas da ocorrência desse *template* na imagem gerada e adquirida.

Efetua-se seguidamente um deslocamento na imagem gerada tendo em conta o valor da diferença mencionada. Se essa diferença for positiva, significa que a zona de *template* na imagem gerada ocorre mais à direita que na imagem adquirida. Neste caso, deslocam-se os pixels da imagem gerada para a esquerda tantos pixels quanto a diferença mencionada. Logicamente, em caso da mesma ter um valor negativo, a ocorrência da zona de *template* será mais à esquerda, fazendo-se o mesmo deslocamento dos pixels da imagem para a direita.

Finalmente, para terminar este processo resta efetuar a compensação da imagem para não se perder parte da mesma com o deslocamento efetuado. No primeiro caso identificado, em que o

deslocamento é feito para a esquerda, os pixels iniciais de abcissa até ao valor do número de pixels de deslocamento, serão anexados à imagem no seu final. Seguindo a mesma ideia, no segundo caso, em que o deslocamento é feito para a direita, o número (igual ao deslocamento) de pixels finais da imagem serão anexados ao seu início.

7.5 Resultados

Após serem aplicados às imagens os processos descritos, estas apresentam a mesma escala com a informação segundo a mesma disposição. Apresentam-se nas imagens seguintes, 7.12 a 7.15, excertos dos resultados da sincronização. Como se pode observar, comprova-se que a informação nas imagens é a mesma, apresentando uma escala bastante próxima.



Figura 7.12: Excerto do resultado do processo de sincronização evidenciando a zona da informação de fabrico.



Figura 7.13: Excerto do resultado do processo de sincronização evidenciando a zona da informação das dimensões do pneu.

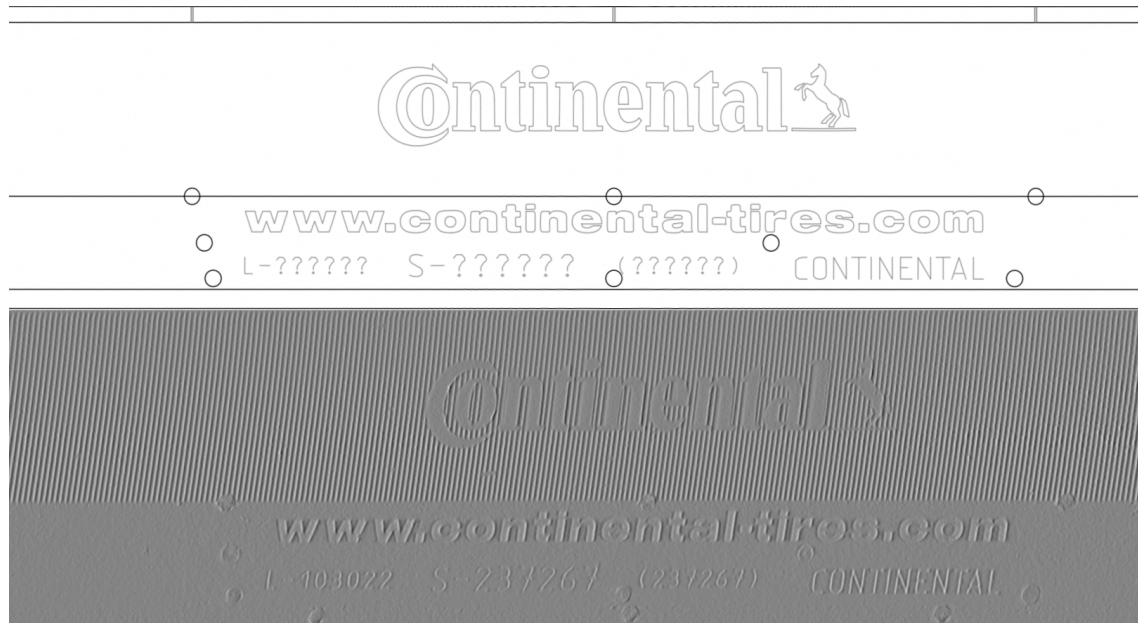


Figura 7.14: Excerto do resultado do processo de sincronização evidenciando a zona do logótipo da marca e do número identificativo do pneu.



Figura 7.15: Excerto do resultado do processo de sincronização evidenciando a zona do aviso de segurança.

7.6 Conclusões

A sincronização das imagens gerada e adquirida segundo os processos descritos possibilita a obtenção de bons resultados. Há, contudo, alguns pontos a assinalar.

Em primeiro lugar, algumas das imagens adquiridas que fazem parte do conjunto em estudo foram obtidas com um sistema de aquisição que apenas contemplava iluminação segundo uma direção aproximadamente perpendicular ao eixo *y*. Como resultando, obtêm-se imagens com alguma informação suprimida ao longo do eixo do *x*, como se exemplifica na figura 7.16. Esta questão, em primeira instância, dificulta a tarefa de deteção baseada em *templates* e, em segunda instância, dificulta a posterior determinação da ROI das zonas de *template*, uma vez que os contornos ficam mal definidos e, pequenos contornos resultantes da quebra das linhas das letras, são considerados ruído.



Figura 7.16: Ilustração comparativa do efeito da iluminação segundo duas direções, em cima, e apenas uma direção, em baixo.

Por outro lado, também a questão da presença de alguns elementos nas imagens adquiridas próximos das zonas a localizar aumentam a dificuldade de definição da sua ROI. Estes elementos não acrescentam qualquer informação à imagem, sendo considerados ruído. Na figura 7.17 apresenta-se um exemplo de uma zona relativa ao aviso de segurança na qual é possível observar a presença de ruído.



Figura 7.17: Ilustração da presença de elementos de ruído próximos da zona do aviso de segurança.

Outra questão a assinalar ainda é a presença de distorção e inclinação em algumas imagens adquiridas. Este facto não só dificulta o sucesso da deteção dos *templates* nessas imagens, como também limita a qualidade dos resultados obtidos com a sincronização. O último acontece porque ao haver distorção, à medida que se consideram áreas da imagem mais distantes das zonas de *template*, o sincronismo perde-se. Sendo o escalamento efetuado de maneira uniforme em toda a área da imagem, é indiferente à ocorrência de zonas na imagens mais esticadas e encolhidas. Na

figura 7.18 ilustra-se um exemplo da perda de sincronismo, conseguindo-se identificar a presença de desfasamento entre as imagens gerada e adquirida.



Figura 7.18: Ilustração da perda de sincronismo entre as imagens gerada, em cima, e adquirida, em baixo.

Por fim, tendo em conta as questões referidas, torna-se necessária uma operação final de ajuste da ROI da imagem adquirida, definida na secção 7.3. Verifica-se que há perda de informação ao ser seleccionada uma área da imagem mencionada igual à da imagem gerada correspondente. Desta forma, são contempladas tolerancias na largura e altura, ficando assim a imagem adquirida com dimensões ligeiramente maiores que a gerada, garantindo que não há qualquer perda de conteúdo.

Apesar dos entraves referidos, são conseguidos resultados satisfatórios com o método proposto para sincronização das imagens. Este foi testado e validado em todo o conjunto de imagens disponível para estudo.

Capítulo 8

Comparação das Imagens Gerada e Adquirida

O presente capítulo dedica-se à etapa que encerra a estrutura de funcionamento do *software acbs*. As imagens gerada e adquirida apresentam, a este ponto, a mesma forma, dimensão semelhante e igual disposição de conteúdo. Consequentemente, a sua comparação de forma manual é assim um processo consideravelmente mais simples que as soluções atuais. Não obstante, centra-se ainda esta etapa na automatização do processo de comparação de maneira a concretizar o objetivo de verificação da conformidade do molde em relação ao seu modelo CAD.

Como foi anteriormente mencionado, a informação presente no molde pode ser dividida em dois tipos. Por um lado, identifica-se a informação de carácter geral, que está presente no modelo CAD e que é diretamente comparável com a informação correspondente no molde. Por outro lado, há informação mais específica que não está incluída no referido modelo. Concretamente, os campos relativos à informação de fabrico (subsecção 2.4.3), ao país de fabrico (subsecção 2.4.4) e aos números identificativos do pneu (subsecção 2.4.10), sendo característicos da fábrica e da produção em que os moldes são utilizados, não estão presentes no modelo CAD.

O primeiro tipo de informação começa por ser interpretado no modelo CAD através da deteção de contornos. Desta maneira, é possível selecionar e localizar a informação para posteriormente averiguar a sua presença na imagem adquirida, seguindo um processo semelhante ao descrito no capítulo anterior para localização das zonas de sincronismo. Por sua vez, relativamente ao segundo tipo de informação, a solução proposta é a de utilizar um algoritmo de OCR de maneira a efetuar a leitura dos caracteres alfanuméricos presentes para posterior verificação manual. Esta verificação torna-se necessária uma vez que não há acesso de forma automática à informação presumivelmente correta.

A exposição dos assuntos relativos à comparação das imagens gerada e adquirida segue uma estrutura de quatro secções. A secção 8.1 trata da solução proposta para leitura da informação não diretamente comparável. Por sua vez, na secção 8.2, aborda-se o processo da verificação de que a informação dita comparável está presente no molde. Finaliza-se com a apresentação dos resultados obtidos, na secção 8.3, aos quais se seguem algumas conclusões gerais, na secção 8.4.

8.1 Informação Lida

O processo de leitura da informação relativa aos campos que não estão presentes no modelo CAD, inicia-se com a deteção da sua localização na imagem adquirida. Uma vez que essa imagem se encontra sincronizada com a imagem gerada, ao se localizarem os referidos campos na última, é possível determinar a sua localização na primeira.

A solução proposta é então a de utilizar novamente a técnica de *template matching* para determinar as localizações pretendidas. Observando o campo de informação referente ao país de fabrico, conclui-se que, ainda que sem informação específica, a expressão *MADE IN* está presente no modelo CAD. De forma semelhante, a localização do campo referente aos números identificativos do pneu, aparece na imagem gerada substituída com pontos de interrogação. Na figura 8.1 ilustra-se um exemplo das zonas mencionadas numa imagem gerada.



Figura 8.1: Zonas relativas aos campos de informação do país de fabrico, em cima, e dos números identificativos do pneu, em baixo, numa imagem gerada.

Com efeito, a ideia é a de criar *templates* para a localização dessa informação. Isto é, localizando-se a expressão mencionada e os pontos de interrogação, que só aparecerem nestes campos, é possível concretizar o objetivo de localização. Na figura 8.2 observam-se os *templates* criados para cada campo. De referir que, o campo relativo à informação de fabrico, uma vez que foi utilizado como zona de sincronismo (capítulo anterior), já é de localização conhecida.



Figura 8.2: *Templates* criados para localização dos campos de informação do país de fabrico, em cima, e dos números identificativos do pneu, em baixo, numa imagem gerada.

De seguida, a localização dos *templates* é feita tendo em conta o mesmo processo descrito na secção 7.1.1 do capítulo anterior, para determinação da localização das zonas de sincronismo na imagem gerada. Primeiramente, os *templates* são escalados tendo em conta o mesmo redimensionamento a que a imagem gerada foi submetida no capítulo anterior. Este passo é de extrema importância uma vez que, como já foi referido, a técnica de *template matching* não é invariante à escala. Desta maneira, no caso dos *templates* apresentarem a escala da imagem não redimensionada, o sucesso do método é posto em causa. Posteriormente, após se ter obtido a localização de cada campo de informação, é extraída a área correspondente na imagem adquirida. No caso dos campos de informação e de país de fabrico, a área é selecionada para a direita de cada *template*. Por sua vez, o *template* para os números identificativos do pneu é feito tendo em conta os pontos de interrogação entre parênteses que aparecem no final do campo. Isto é feito pelo motivo de que é possível diferenciá-lo dos outros conjuntos, identificando-o univocamente. Naturalmente, a seleção da zona correspondente a este campo na imagem adquirida é feita tendo em conta uma área para a esquerda da localização deste *template*. De notar que, o processo de seleção da área correspondente na imagem adquirida, partindo da zona localizada na imagem gerada contempla uma tolerância, de modo a compensar as pequenas imperfeições de sincronização. Na figura 8.3 encontram-se ilustradas as localizações dos três campos de informação a ler, bem como a sua área correspondente na imagem adquirida.

De maneira a concretizar a leitura da informação presente na área obtida para cada campo, é necessário proceder a um pré-processamento de cada uma dessas pequenas imagens de forma a obter apenas a informação a ler. Tomando como exemplo de explicação o campo relativo aos números identificativos do pneu, definem-se uma série de etapas aplicadas à imagem relativa ao mesmo (última imagem da figura 8.3).

Em primeiro lugar, a imagem é filtrada através de um filtro bilateral e é posteriormente submetida a um aumento de contraste. Observa-se na figura 8.4 o resultado deste passo. Em segundo lugar, a imagem é binarizada e de seguida é alvo de um processo de dilatação, de forma a melhorar a definição dos caracteres presentes. Ilustra-se o resultado desta etapa na figura 8.5. Em terceiro lugar, com a utilização da função *findContours*, anteriormente mencionada, são determinados os contornos da imagem resultante da dilatação. Visualiza-se na figura 8.6 a representação dos contornos detetados. Em quarto lugar, de maneira a determinar a ROI na qual a informação a ler se encontra, é selecionada a área horizontal da imagem dada pela altura da *bounding box* de maior área dos contornos. Deste passo resulta a imagem presente na figura 8.7. Por fim, de maneira a eliminar ruído ainda presente na imagem, são novamente detetados os contornos da imagem e são determinadas as mínimas circunferências que os envolvem, com a utilização da função *minEnclosingCircle*, para eliminar os elementos da imagem relativos às circunferências mais pequenas. Ilustra-se o resultado deste processo na figura 8.8.

Finalizado o pré-processamento definido pelas etapas descritas, a informação presente na imagem está pronta a ser lida. Esta tarefa foi concretizada com a utilização do *tesseract*, um *software* de OCR desenvolvido pela Google, disponível sob licença gratuita.

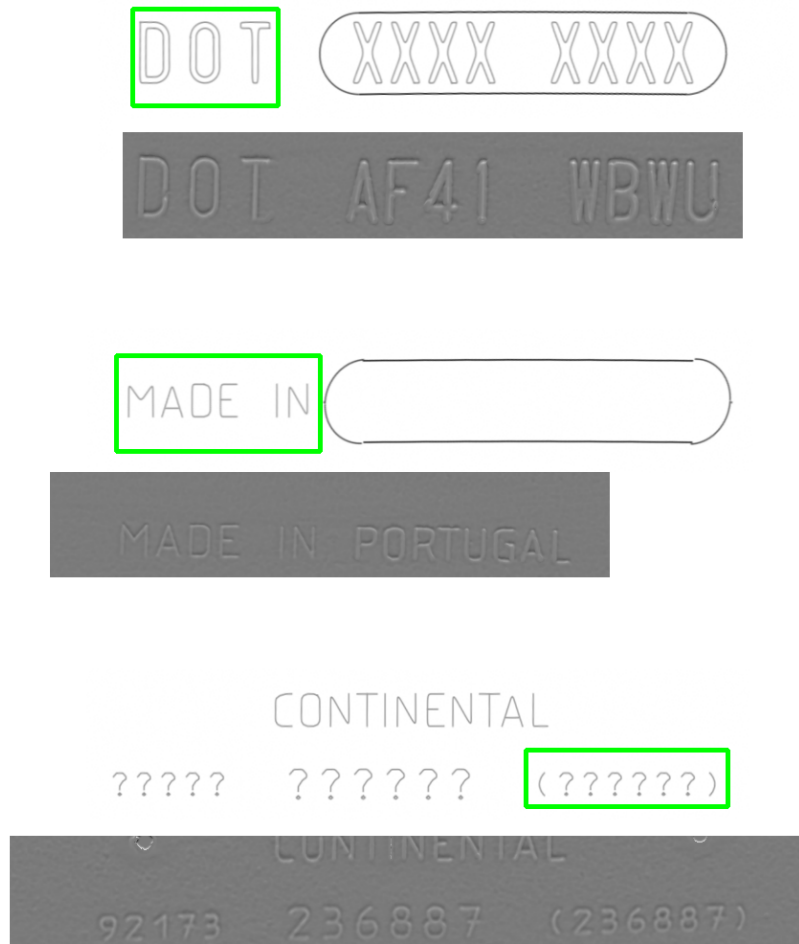


Figura 8.3: Resultados da detecção de cada *template* na imagem gerada e respetiva área correspondente na imagem adquirida. De cima para baixo: informação de fabrico, país de fabrico e números identificativos do pneu.



Figura 8.4: Imagem resultante do processo de aumento de contraste.

Por fim, ilustram-se nas figuras 8.9 e 8.10 os mesmos passos seguidos para definir a imagem final para leitura dos campos de informação e de país de fabrico, respetivamente.



Figura 8.5: Imagem resultante do processo de binarização, em cima, e da sua dilatação, em baixo.



Figura 8.6: Imagem resultante da representação dos contornos detetados na imagem dilatada.



Figura 8.7: Imagem resultante do processo de definição da região de interesse.

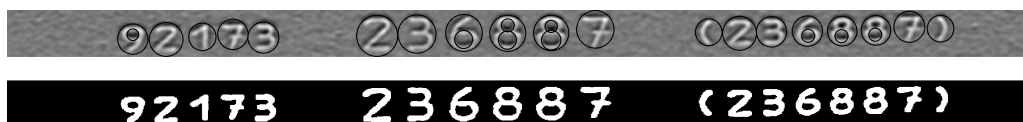


Figura 8.8: Imagem resultante da representação das mínimas circunferências envolventes dos contornos, em cima, e da eliminação do ruído, em baixo.



Figura 8.9: Imagens relativas aos passos seguidos na definição da imagem para leitura da informação de fabrico. De cima para baixo: aumento de contraste, binarização, dilatação, contornos detetados, região de interesse, mínimas circunferências envolventes dos contornos, e eliminação do ruído.



Figura 8.10: Imagens relativas aos passos seguidos na definição da imagem para leitura do país de fabrico. De cima para baixo: aumento de contraste, binarização, dilatação, contornos detetados, região de interesse, mínimas circunferências envolventes dos contornos, e eliminação do ruído.

8.2 Informação Comparável

O processo de tratamento da informação comparável inicia-se com a sua identificação. Com efeito, o primeiro passo é o de eliminar a informação anteriormente lida da imagem gerada, uma vez que não interessa efetuar a sua comparação. Tendo-se a sua localização pelo método descrito na secção anterior, a eliminação destes campos é conseguida substituindo a área da sua localização por píxeis brancos. Novamente, a seleção da área a eliminar é feita para a direita, nos campos de informação e país de fabrico, e para a esquerda, no campo dos números identificativos do pneu.

Seguidamente, a identificação da informação a comparar é feita através da determinação dos contornos dos elementos presentes na imagem gerada. Através deste método, é possível localização cada elemento, sendo uma premissa imprescindível à concretização da tarefa de comparação. Além disso, uma vez que uma imagem gerada é dada por linhas pretas em fundo branco, a determinação dos contornos de forma correta é um processo obtido facilmente com sucesso. Não obstante, a comparação de cada elemento dado por cada contorno de forma singular é, por sua vez, um processo relativamente complicado e que não traz vantagens em relação ao objetivo pretendido. A solução proposta é então a de agrupar contornos de forma a criar conjuntos de informação e efetuar a verificação da presença de cada conjunto na imagem adquirida.

Em primeiro lugar, após se efetuar a deteção dos contornos na imagem gerada, estes são organizados por ordem crescente de ocorrência de coordenadas do seu centro ao longo do eixo x da imagem. Desta forma, procede-se em segundo lugar ao agrupamento de contornos por um critério de proximidade. Na figura 8.11, observa-se o exemplo de alguns dos conjuntos obtidos, identificados na imagem gerada através de uma *bounding box*. Como se observa no logótipo da Continental, na imagem no topo da figura 8.11, alguns conjuntos aparecem sobrepostos, pela razão de que excedem o critério de proximidade estabelecido. Com efeito, um processo de deteção e agrupamento de conjuntos sobrepostos é proposto. Na figura 8.12 ilustra-se o resultado da correção destas situações.

Tendo os conjuntos de informação definidos, o próximo passo é o de concretizar a verificação da sua presença na imagem adquirida. Mais uma vez, partindo da premissa de que as imagens gerada e adquirida se encontram com a mesma escala, forma e disposição de conteúdo, a zona de ocorrência de cada grupo de contornos na imagem gerada será aproximadamente o mesmo que na adquirida, no caso de haver efetivamente ocorrência. Desta forma, a solução proposta para efetuar a verificação mencionada é a de averiguar se cada grupo de contornos existe na área correspondente da imagem adquirida. De notar que a seleção da área da referida imagem para cada conjunto, é novamente feita tendo em conta uma tolerância na sua localização. Seguidamente, o método proposto é semelhante ao utilizado para a localização das zonas de sincronismo na imagem adquirida, descrito no capítulo anterior: com a técnica de *template matching*, avaliar possíveis localizações do conjunto de contornos, que funciona como *template*, e validar a sua efetiva localização com *feature matching*. Com este processo, conseguem-se superar os problemas de pequenas diferenças de escala que ainda possam existir, bem como diferente orientação resultante de distorções na imagem adquirida. Assim, a ideia é então a de validar as áreas em que há um número de

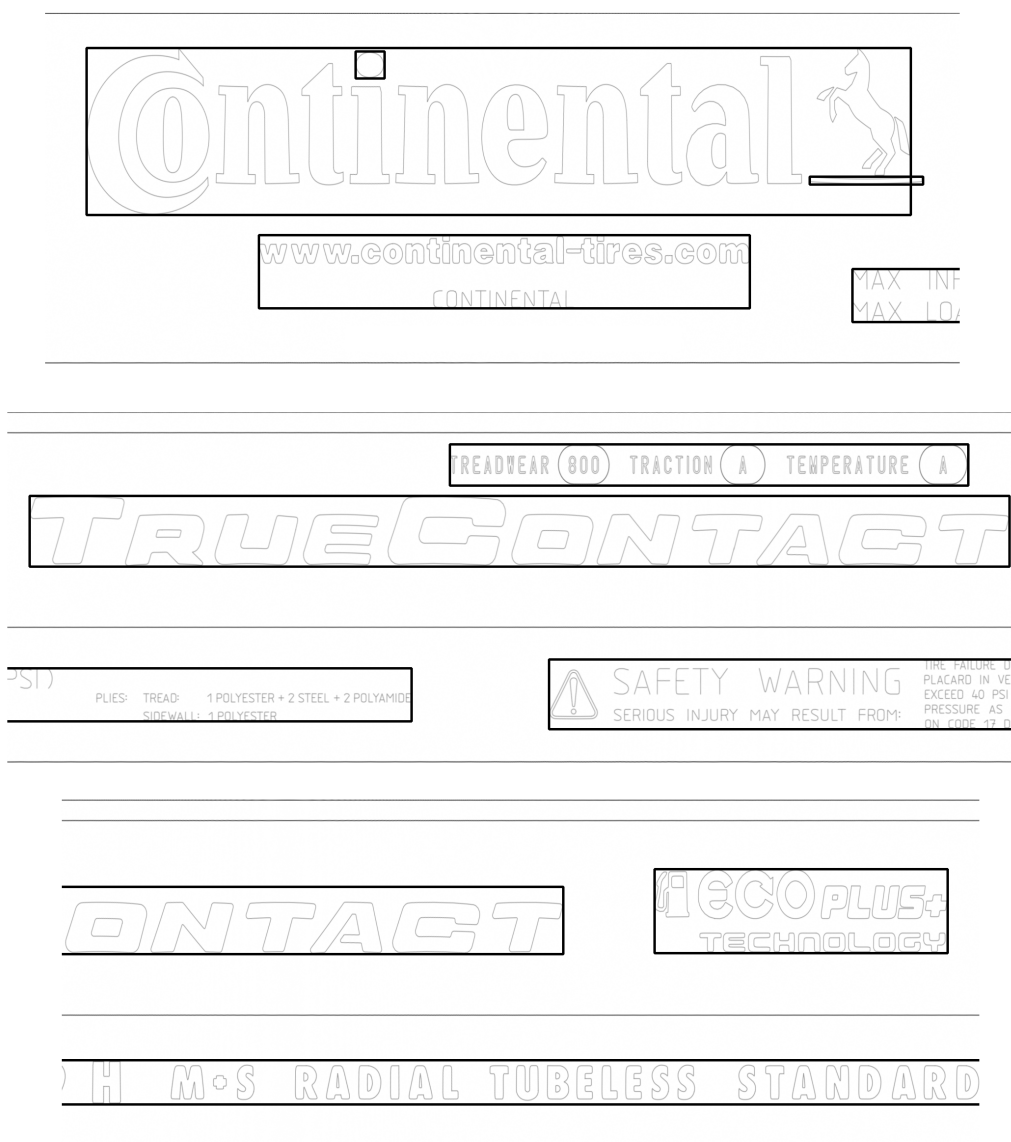


Figura 8.11: Ilustração do resultados do agrupamento de contornos identificados através de *bounding boxes*.

correspondências considerável de *features* e rejeitar aquelas para o qual esse facto não se verifica.

Por fim, ilustram-se nas figuras 8.13 a 8.15, exemplos da aplicação do processo referido localização de alguns conjuntos de contornos. No final verifica-se a identificação do campo de informação de forma correta.

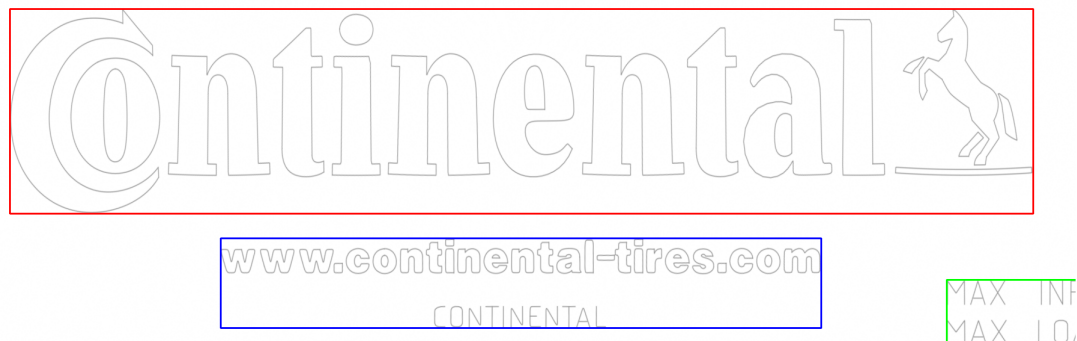


Figura 8.12: Ilustração do resultado da correção da sobreposição de conjuntos de contornos com o seu agrupamento.

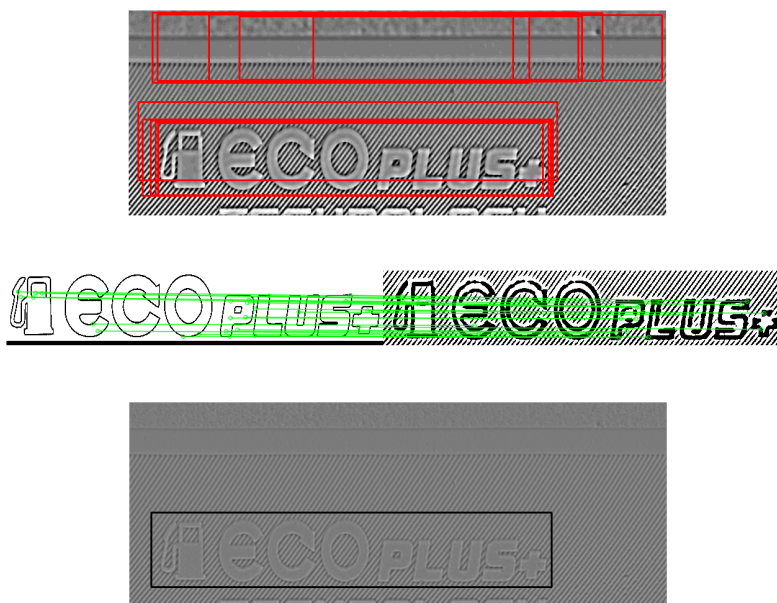


Figura 8.13: Ilustração da aplicação do processo de verificação da presença do conjunto de contornos relativo ao logótipo *ECO PLUS* na imagem adquirida. De cima para baixo: zonas de localização candidatas, dadas por *template matching*; melhor correspondência de *features*; e zona validada para localização.

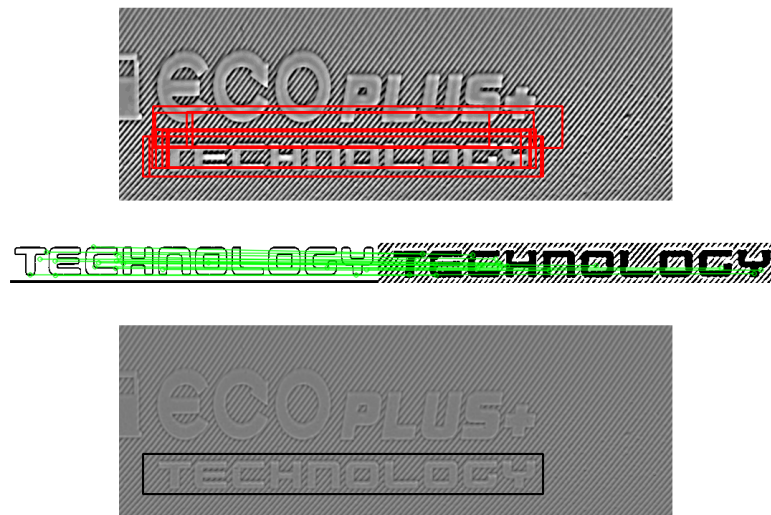


Figura 8.14: Ilustração da aplicação do processo de verificação da presença do conjunto de contornos relativo à escrita *TECHNOLOGY* na imagem adquirida. De cima para baixo: zonas de localização candidatas, dadas por *template matching*; melhor correspondência de *features*; e zona validada para localização.

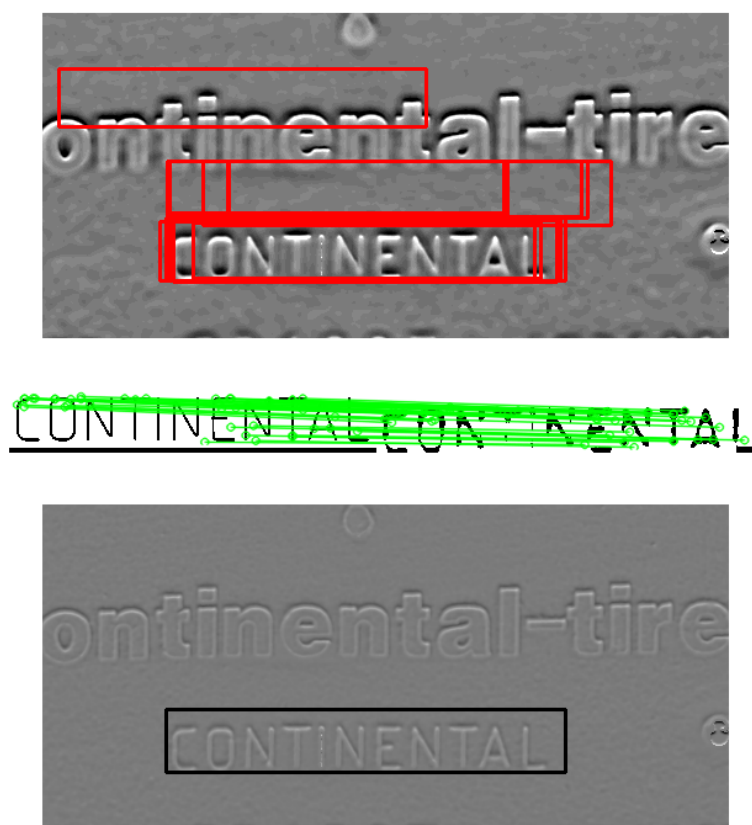


Figura 8.15: Ilustração da aplicação do processo de verificação da presença do conjunto de contornos relativo à escrita *CONTINENTAL* na imagem adquirida. De cima para baixo: zonas de localização candidatas, dadas por *template matching*; melhor correspondência de *features*; e zona validada para localização.

8.3 Resultados

Relativamente aos resultados obtidos, em primeiro lugar ilustram-se na figura 8.16 os relativos à leitura dos campos de informação de fabrico, país de fabrico e números identificativos do pneu. Como se pode observar, a leitura é efetuada de forma correta para todos os campos.



Figura 8.16: Ilustração dos resultados obtidos na leitura da informação relativa aos campos de informação de fabrico, país de fabrico e números identificativos do pneu, respetivamente, de cima para baixo.

Em segundo lugar, em relação à verificação da presença de cada conjunto de contornos na imagem adquirida, observam-se nas figuras 8.17 a 8.19 exemplos de alguns resultados obtidos.



Figura 8.17: Ilustração do resultado do processo de validação da presença de alguns campos de informação na zona do logótipo da CONTINENTAL, com uma *bounding box* a verde.

Como se pode observar nas figuras 8.17 e 8.18, a validação da presença dos campos de informação é feita de forma correta. Com o objetivo de verificar o funcionamento do método proposto para o caso em que as imagens não estão em concordância, na figura 8.19 utilizou-se uma imagem gerada manualmente alterada, tendo-se mudado a posição do campo relativo à escrita *TRUE CONTACT* e à informação do desgaste do piso, tração e temperatura. Verifica-se assim que há a identificação de que há uma falha na validação, sendo ilustrada com uma *bounding box* a vermelho numa área perto da zona em que acontece a falha.



Figura 8.18: Ilustração do resultado do processo de validação da presença de alguns campos na zona da informação relativa ao desgaste do piso, tração e temperatura, com uma *bounding box* a verde.

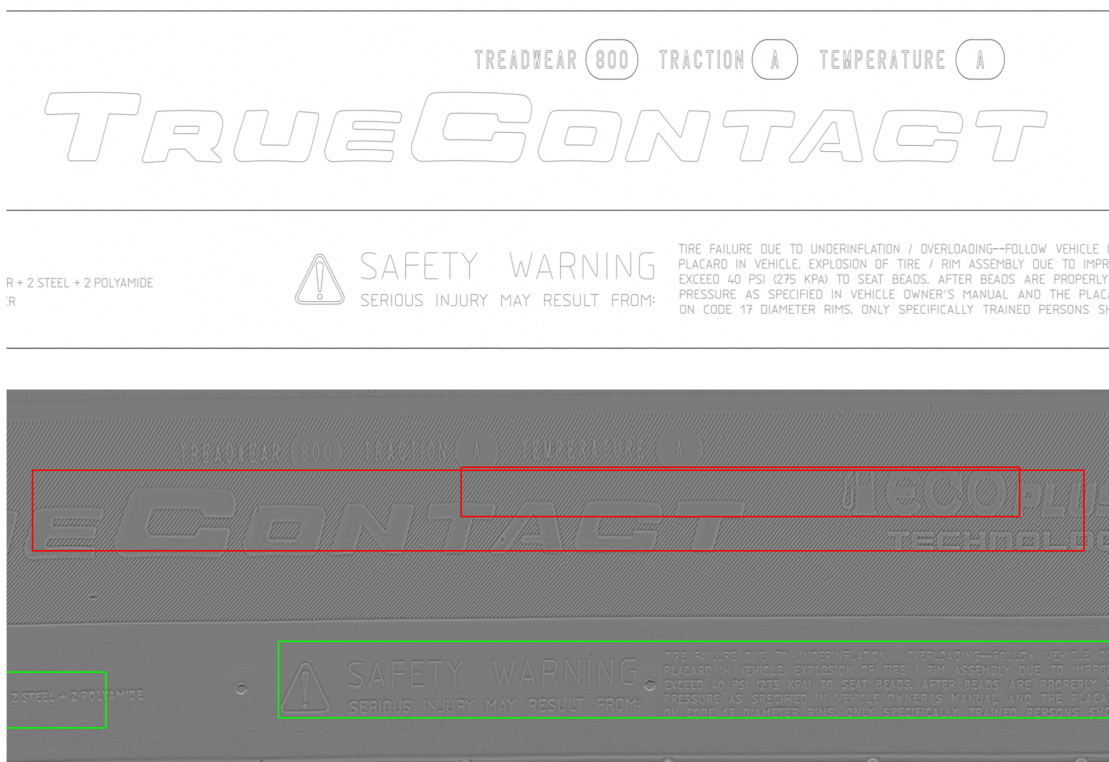


Figura 8.19: Ilustração do resultado do processo de rejeição da presença dos campos de informação, com uma *bounding box* a vermelho.

8.4 Conclusões

De forma a concluir o presente capítulo, há algumas conclusões a assinalar em relação ao processo de interpretação e comparação da informação das imagens gerada e adquirida.

Em primeiro lugar, a leitura da informação dita não comparável é conseguida de forma correta para os casos anteriormente ilustrados. No entanto, para os casos das imagens adquiridas com um sistema que apenas contempla uma direção de iluminação, a sua leitura não é conseguida com sucesso. Isto deve-se ao facto de que a supressão de algumas linhas das letras dificulta consideravelmente a correta segmentação das mesmas. Na figura 8.20 encontra-se um exemplo do campo de país de fabrico de uma dessas imagens, observa-se a referida falta de algumas linhas.



Figura 8.20: Exemplo do campo de informação relativo ao país de fabrico numa imagem adquirida com uma só direção de iluminação.

Por outro lado, o principal ponto a assinalar em relação à interpretação da informação comparável prende-se com a seleção da informação presente na imagem gerada que interessa comparar. Ainda que se eliminem os campos que são lidos e que haja uma seleção na representação da informação presente no ficheiro DXF, verifica-se que há elementos presentes na imagem que não são relevantes ao objetivo pretendido. Na figura 8.21 ilustra-se o exemplo da deteção de um conjunto de contornos que representa um padrão incompleto presente na imagem gerada. O problema subjacente, é o de que é difícil efetuar o correto agrupamento dos diferentes campos de informação com a presença destes conjuntos de contornos próximos de outros conjuntos de interesse.

Como se pode concluir, esta questão põe em causa a consequente verificação da presença dos campos de informação na imagem adquirida. Por fim, é de referir ainda que tendo em conta o método proposto, não se garante a correta validação da presença ou falta de elementos desta natureza.

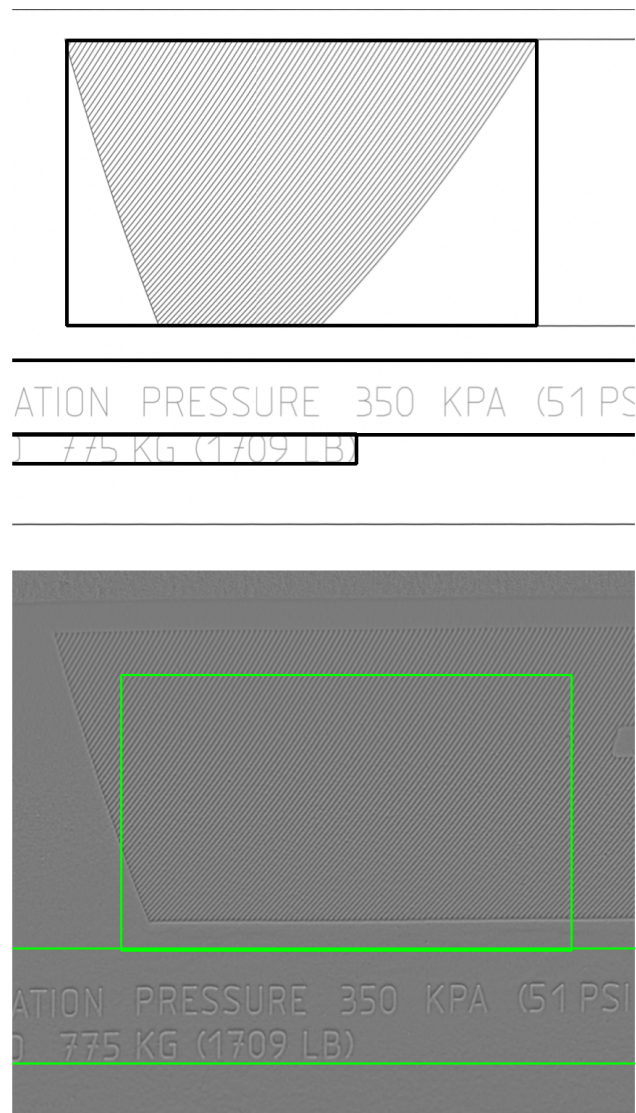


Figura 8.21: Ilustração de um conjunto de contornos relativo a um padrão incompleto na imagem gerada, em cima, e a sua detecção na imagem adquirida, em baixo.

Capítulo 9

Conclusões e Trabalho Futuro

9.1 Conclusões

Atualmente, a inspeção dos moldes utilizados para a vulcanização da parede dos pneus é feita de forma manual. Sendo uma tarefa difícil ao olho humano, são por vezes validados moldes que não estão de acordo com as suas especificações. Assim, esta dissertação incidiu no desenvolvimento do *software* a integrar num sistema que efetuasse a correta inspeção dos referidos moldes de forma automática, tendo como objetivo a deteção de erros tais como a falta de informação ou a sua presença no local errado.

Primeiramente, foi entendido o funcionamento do sistema de aquisição de imagem implementado pela Continental Mabor. A primeira versão do mesmo apresentava apenas dois sentidos de iluminação segundo a mesma direção. Como resultado, as imagens adquiridas apenas apresentavam informação segundo uma direção, dificultando a sua interpretação. Foi proposta e documentada uma nova versão do referido sistema de aquisição que inclui duas direções perpendiculares de iluminação, melhorando significativamente a informação contida nas imagens adquiridas.

Foi seguidamente validado um método para gerar uma imagem a partir do modelo CAD do molde em formato DXF de forma totalmente automática. Este método assenta principalmente na interpretação do modelo CAD como um conjunto de linhas parametrizáveis e sua posterior representação. É necessário ter em conta que o modelo CAD deve conter apenas informação relativa ao molde da parede em si e cuja presença interessa verificar. Alguns ficheiros interpretados no decorrer desta dissertação apresentaram linhas que traduzem informação textual como legenda do conteúdo do modelo ou ainda linhas de guia de desenho, como consequência da maneira como o modelo é projetado. Em primeira instância, esta questão dificulta a obtenção da imagem necessária apenas da parede do pneu e, em segunda instância, a localização dos campos de informação a inspecionar. No entanto, o método desenvolvido funcionou com sucesso para um número razoável de casos de teste.

De forma a possibilitar a comparação entre as imagens gerada e adquirida a partir do modelo CAD de um molde, foi ainda efetuado um processamento a cada uma das mesmas. Em relação à imagem gerada, foi implementado um método para efetuar a sua retificação, isto é, foi obtida

uma imagem retangular a partir de uma com forma anelar (do pneu) através de uma conversão de referencial. Por sua vez, a imagem adquirida foi processada de maneira a ser selecionada a sua secção horizontal e vertical que contém apenas a informação relativa à parede do pneu. Por fim, foi apresentada uma forma de efetuar a sincronização das imagens no que toca ao escalamento e conteúdo de cada uma. A este ponto, o sistema apresenta as imagens num ponto de comparação não automático, sendo uma solução consideravelmente mais simples que as atualmente adotadas.

Finalmente, implementou-se um método para efetuar a leitura dos campos de informação que, pelas razões anteriormente mencionadas, não constam do modelo CAD. Desta forma, a sua validação por um operador é, agora, um processo trivial. Relativamente à comparação das imagens, os entraves encontrados são cruciais na obtenção de resultados excelentes. No entanto, valida-se a sua utilização para um razoável número de casos em estudo, restando aperfeiçoar o seu funcionamento.

9.2 Trabalho Futuro

Para trabalho futuro deverá primeiramente alargar-se o número de modelos CAD estudados de forma a perceber que outros tipos de entidades do formato DXF poderão aparecer no projeto dos moldes. Com efeito, um método de parametrização para representação de cada novo tipo de entidade estabelece-se como um ponto a desenvolver.

Ainda em relação ao modelos CAD, interessa perceber que tipo de modelos existem disponíveis. Para a concretização dos objetivos do sistema de inspeção em estudo, apenas interessa que estes modelos apresentem a informação a comparar, ou, por outro lado, que estejam projetados de tal maneira que a representação das entidades relativas apenas à informação de interesse seja possível. No caso de não haver modelos nas condições pretendidas, identifica-se como tópico de trabalho futuro a seleção da informação de interesse, eliminando-se a restante.

Por fim, relativamente à concretização do método proposto para verificação da conformidade das imagens gerada e adquirida, é de primordial importância começar por alargar o número de casos de estudo. Desta forma, novos testes de validação do método são um ponto a ter em conta. Por outro lado, ainda que não fosse objetivo desta dissertação, propõe-se o estudo de um método mais minucioso, não só para verificar a presença dos campos de informação com a mesma disposição do modelo CAD, mas também para averiguar a presença de algum erro nos campos.

Anexo A

Índices de Carga e Velocidade de um pneu

A.1 Valores para o Índice de Velocidade (SI) de um pneu

Tabela A.1: Letra relativa ao Índice de Velocidade (SI) de um pneu e sua correspondência em km/h

SI	km/h	SI	km/h	SI	km/h
A1	5	D	65	Q	160
A2	10	E	70	R	170
A3	15	F	80	S	180
A4	20	G	90	T	190
A5	25	J	100	U	200
A6	30	K	110	H	210
A7	35	L	120	V	240
A8	40	M	130	W	270
B	50	N	140	Y	300
C	60	P	150		

A.2 Valores tabelados para o Índice de Carga (LI) de um pneu

Tabela A.2: Valores para o Índice de Carga (LI) de um pneu e sua correspondência em kg

LI	kg	LI	kg	LI	kg	LI	kg	LI	kg	LI	kg
19	77.5	50	190	81	462	112	1120	143	2725	174	6700
20	80	51	195	82	475	113	1150	144	2800	175	6900
21	82.5	52	200	83	487	114	1180	145	2900	176	7100
22	85	53	206	84	500	115	1215	146	3000	177	7300
23	87.5	54	212	85	515	116	1250	147	3075	178	7500
24	90	55	218	86	530	117	1285	148	3150	179	7750
25	92.5	56	224	87	545	118	1320	149	3250	180	8000
26	95	57	230	88	560	119	1360	150	3350	181	8250
27	97.5	58	236	89	580	120	1400	151	3450	182	8500
28	100	59	243	90	600	121	1450	152	3550	183	8750
29	103	60	250	91	615	122	1500	153	3650	184	9000
30	106	61	257	92	630	123	1550	154	3750	185	9250
31	109	62	265	93	650	124	1600	155	3875	186	9500
32	112	63	272	94	670	125	1650	156	4000	187	9750
33	115	64	280	95	690	126	1700	157	4125	188	10000
34	118	65	290	96	710	127	1750	158	4250	189	10300
35	121	66	300	97	730	128	1800	159	4375	190	10600
36	125	67	307	98	750	129	1850	160	4500	191	10900
37	128	68	315	99	775	130	1900	161	4625	192	11200
38	132	69	325	100	800	131	1950	162	4750	193	11500
39	136	70	335	101	825	132	2000	163	4875	194	11800
40	140	71	345	102	850	133	2060	164	5000	195	12150
41	145	72	355	103	875	134	2120	165	5150	196	12500
42	150	73	365	104	900	135	2180	166	5300	197	12850
43	155	74	375	105	925	136	2240	167	5450	198	13200
44	160	75	387	106	950	137	2300	168	5600	199	13600
45	165	76	400	107	975	138	2360	169	5800	200	14000
46	170	77	412	108	1000	139	2430	170	6000	201	14500
47	175	78	425	109	1030	140	2500	171	6150	202	15000
48	180	79	437	110	1060	141	2575	172	6300	203	15500
49	185	80	450	111	1090	142	2650	173	6500	204	16000

Anexo B

Sistemas de Aquisição de Imagem e de Iluminação

Em tarefas de inspeção visual automática, a aquisição de imagem é um processo importante e a sua execução de forma mais cuidada possível é imprescindível ao sucesso do desenvolvimento do sistema de inspeção. Com efeito, o dito processo deve ser estudado e levado a cabo de maneira a evidenciar a informação desejada no objeto a adquirir e eliminar a não desejada tanto quanto possível. Como consequência, será necessário um processamento mais simples para atingir os resultados pretendidos.

Serve então este anexo para apresentar os conceitos relativos aos sistemas de aquisição de imagem e de iluminação envolvidos no processo de aquisição dos moldes. É importante referir que esta aquisição foi da responsabilidade da Continental Mabor. No entanto, um bom conhecimento e documentação de alguns conceitos e do método utilizado julga-se interessante.

Assim, na secção [B.1](#), aborda-se o tipo de câmara utilizada e o seu princípio de funcionamento; a secção [B.2](#), é dedicada à constituição e modo de funcionamento do sistema de aquisição; e, por fim, na secção [B.3](#) apresenta-se uma breve explicação do método utilizado para gerar a imagem final do molde adquirido.

B.1 Princípio de funcionamento da câmara

O primeiro ponto a identificar em relação à aquisição de imagens é que a mesma foi realizada com recurso a uma câmara de linha. Esta assenta no princípio de que se faz a aquisição de uma única linha de pixeis por captura. Assim, implica-se que haja movimento do objeto a adquirir sob a câmara de forma a que o mesmo seja adquirido por completo, linha a linha. Pressupõe-se então que o dito movimento aconteça com direção perpendicular à linha de pixeis adquirida. A imagem final do objeto é conseguida com a agregação de todas as linhas de pixeis adquiridas. Na figura [B.1](#) ilustra-se a aquisição de imagem de um objeto, tendo em conta a agregação do resultado de várias capturas.

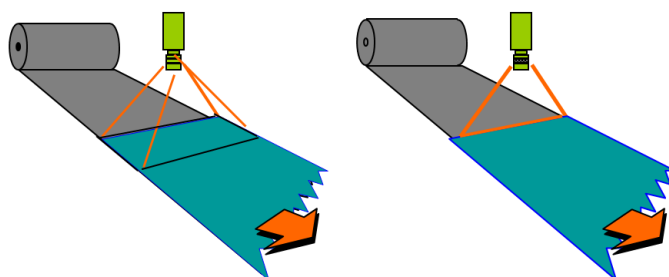


Figura B.1: Ilustração comparativa da aquisição através de uma câmera de área, à esquerda, e de uma câmera de linha, à direita.

Como se pode observar na figura B.1, esta técnica de aquisição baseada na agregação de várias capturas pode ser conseguida não só com o uso de uma câmera de linha mas também com uma câmera dita tradicional, que captura uma área. Este tipo de aquisição é muito utilizada em aplicações em que são necessárias imagens com elevada resolução de objetos de tamanho relativamente grande. Todavia, a utilização de câmaras de linha apresenta vantagens a assinalar.

Com a utilização de câmaras de área surgem problemas de distorção e a agregação das imagens resultantes das capturas pode apresentar problemas de sobreposição. Ou seja, a tarefa de reconstrução de uma imagem final de qualidade é mais difícil. Por outro lado, com a utilização de câmaras de linha, efetuando a sincronização da velocidade do movimento do objeto com a taxa de captura das linhas de pixels e tendo ainda em conta a resolução métrica por pixel pretendida, conseguem-se obter imagens com elevada qualidade mais facilmente. Na figura B.2 pode-se observar um exemplo de um excerto da aquisição de um molde utilizando uma câmera de linha, com resolução de 0.15mm por pixel. Outra questão ainda é que é necessário ter em conta o tempo de exposição da câmera de área para adquirir imagens nítidas. Por conseguinte, são conseguidas velocidades de aquisição mais rápidas com a utilização de câmaras de linha. Para finalizar, a questão da iluminação também é um facto relevante. Com a aquisição através de uma câmera de linha conseguem-se superar problemas de iluminação não uniforme, típicos de uma aquisição com câmaras de área. Isto acontece uma vez que a iluminação e aquisição têm em conta uma área reduzida do objeto a adquirir.



Figura B.2: Exemplo de imagem resultante da aquisição de um molde utilizando uma câmera de linha.

B.2 Sistemas de aquisição e iluminação

B.2.1 Constituição do sistema

O sistema de aquisição utiliza, como foi referido em B.1, uma câmara de linha para a obtenção da imagem do molde. Especificamente, a câmara utilizada foi uma *Teledyne DALSA Spyder 3*, com 2k de resolução.

Acoplada à referida câmara, foi utilizada um lente *Carl Zeiss Planar 1.4/50 ZF.2*. Esta lente possui uma abertura máxima de $f/1.4$ e uma distância focal de 50mm.

Na figura B.3 está representada a montagem da câmara com a lente referida.



Figura B.3: Montagem da câmara e lente utilizada no sistema de aquisição.

Relativamente ao sistema de iluminação, foram montados sobre o molde três painéis com três LEDs¹ de luz branca em cada um. Com efeito, o tipo de sistema utilizado pode ser dividido em três sistemas de iluminação direta, frontal e em campo escuro. Os LEDs neles empregados possuem um valor de corrente nominal de 350mA para 3V de valor de tensão, emitindo 100 lumens nessas condições de alimentação. Na figura B.4 está representada a montagem dos referidos sistemas.

Para enviar os sinais de disparo à câmara, optou-se pelo uso de um Arduíno UNO. Este, que possui um microcontrolador *ATmega 328p* incorporado, pode ser visto na figura B.5. Foi ainda utilizado um controlador, cuja função é a de receber sinais do Arduíno e enviá-los aos painéis de LEDs. Uma imagem do controlador pode ser vista na figura B.6.

Para alimentação da câmara e do controlador, foram escolhidas fontes de alimentação de 12V e 24V respetivamente. Estas fontes estão representadas na figura B.7.

Por fim, o movimento do molde sob a câmara foi conseguido com a utilização de uma mesa giratória na qual se coloca o mesmo. Na figura B.8, observa-se a montagem da câmara com a lente, dos painéis de LEDs e da mesa giratória com um molde sobre a mesma.

¹LED - *Light Emitting Diode*



Figura B.4: Montagem dos painéis de LEDs utilizados no sistema de aquisição.

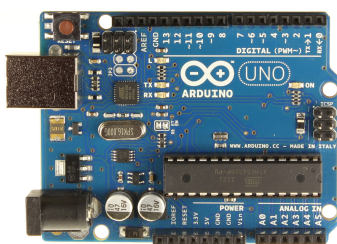


Figura B.5: Arduíno UNO.



Figura B.6: Controlador utilizado para envio de sinal aos painéis de LEDs.

B.2.2 Estrutura de funcionamento

O sistema de aquisição segue a estrutura ilustrada no esquema da figura B.9.

Analisando o esquema da figura, a aquisição começa com a comunicação do PC para o Arduíno, através da porta série. Essa comunicação concretiza o início do programa do referido microprocessador, que envia os sinais de disparo à câmara e ao controlador dos painéis de LEDs.

É importante realçar que a aquisição das linhas é feita uma vez com cada painel de LEDs aceso. Ou seja, a aquisição de uma linha de pixels do molde é efetuada três vezes, uma com cada painel aceso. Desta maneira, a frequência de disparo da câmara é o triplo da frequência de cada painel. Mais precisamente, a frequência de aquisição da câmara é de aproximadamente 16.5kHz e cada painel liga com uma frequência aproximada de 5.5kHz.

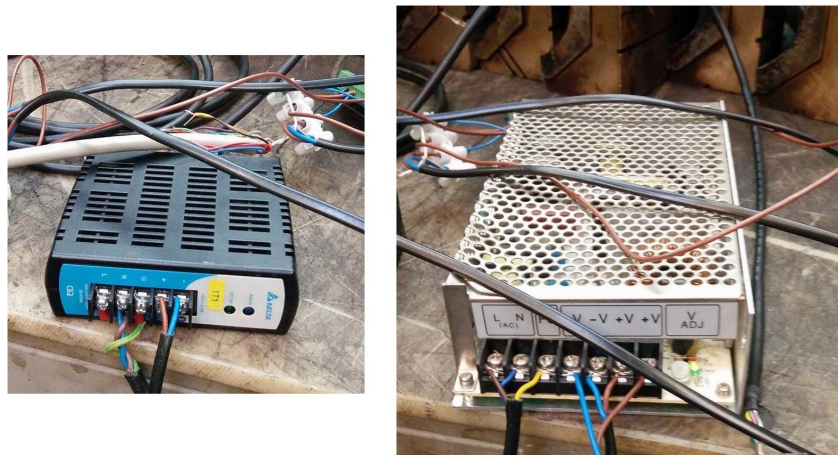


Figura B.7: Fontes de alimentação utilizadas para alimentar a câmara, à esquerda, e o controlador, à direita.



Figura B.8: Montagem da câmara com a lente e dos painéis de iluminação sobre a mesa giratória onde se encontra o molde.

Como resultado, serão obtidas três imagens consecuentes das agregações das linhas adquiridas com cada painel aceso. Isto é, haverá uma imagem resultante apenas com iluminação pelo topo, outra com iluminação pela esquerda e ainda outra com iluminação pela direita. Na secção seguinte,

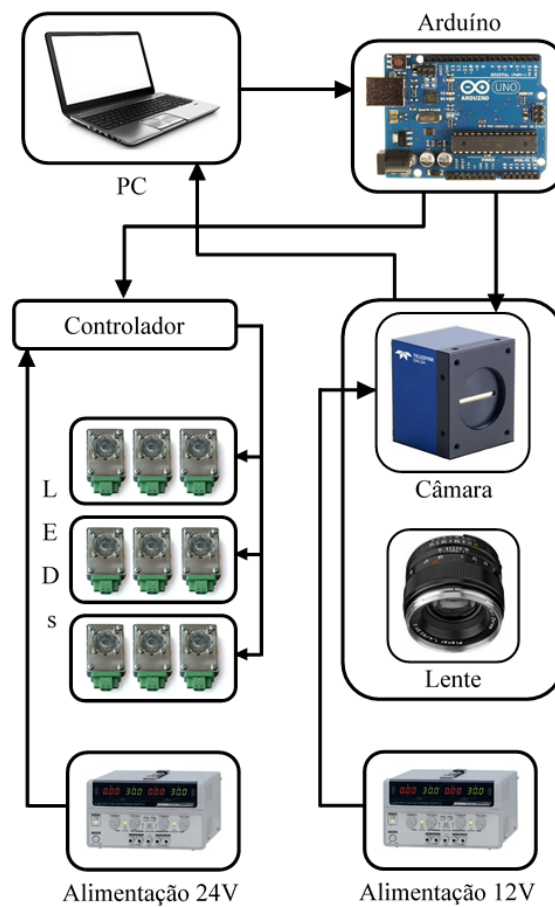


Figura B.9: Estrutura esquematizada do sistema de aquisição.

B.3, é explicada a razão da aquisição das referidas três imagens.

Por fim, é de notar que as linhas adquiridas pela câmara são enviadas para o PC, no qual será feita a sua agregação originando-se as três imagens mencionadas.

B.3 Imagem adquirida

Para a obtenção da imagem do molde a ser processada foi utilizada a técnica de *photometric stereo*. Esta técnica, introduzida por Woodham[34], permite estimar a superfície de um objeto através de várias imagens do mesmo, capturadas de um ponto fixo mas com diferentes direções de iluminação. Com efeito, estabelece-se a premissa de que a quantidade de luz refletida por uma superfície depende da orientação dessa mesma superfície em relação à fonte de luz e, por conseguinte, ao ponto de captura. Na figura B.10, ilustra-se um sistema de aquisição com diferentes direções de iluminação, com o objetivo de se aplicar a técnica de *photometric stereo* para obter a superfície do objeto.

Esta técnica apresenta a vantagem de eliminar problemas induzidos pela iluminação, tais como sombras ou reflexões. Desta maneira, tendo em conta o caso em estudo da aquisição de imagem

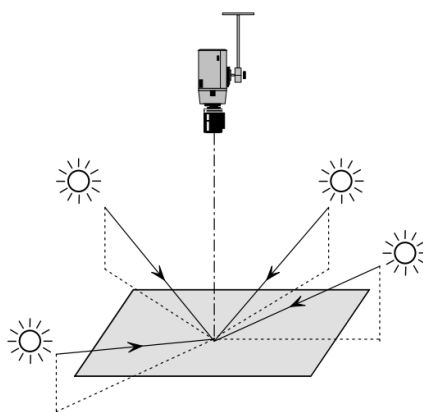


Figura B.10: Exemplo de aquisição tendo a câmara num ponto fixo e várias direções de iluminação. [11]

de moldes de pneus, sendo estas superfícies metálicas com relevos, facilmente apresentariam os problemas de iluminação mencionados. Procedeu-se assim à montagem do painel de LEDs, referido em B.2.1, de forma a adquirir três imagens, cada uma iluminada com um painel diferente. Assim, com o cruzamento da informação vinda das diferentes imagens capturadas, é possível atingir uma imagem final que representa uma boa estimativa da superfície do dito molde. É de referir que a aplicação desta técnica está pressuposta no *software* de aquisição referido na arquitetura do sistema de inspeção, na página 33. Este tem como entrada as três imagens resultantes das iluminações pela esquerda, direita e topo e como saída apresenta a imagem reconstruída.

Na figura B.11 encontra-se um exemplo de um excerto das imagens resultantes de uma aquisição com cada direção de iluminação e a respetiva reconstrução da superfície.

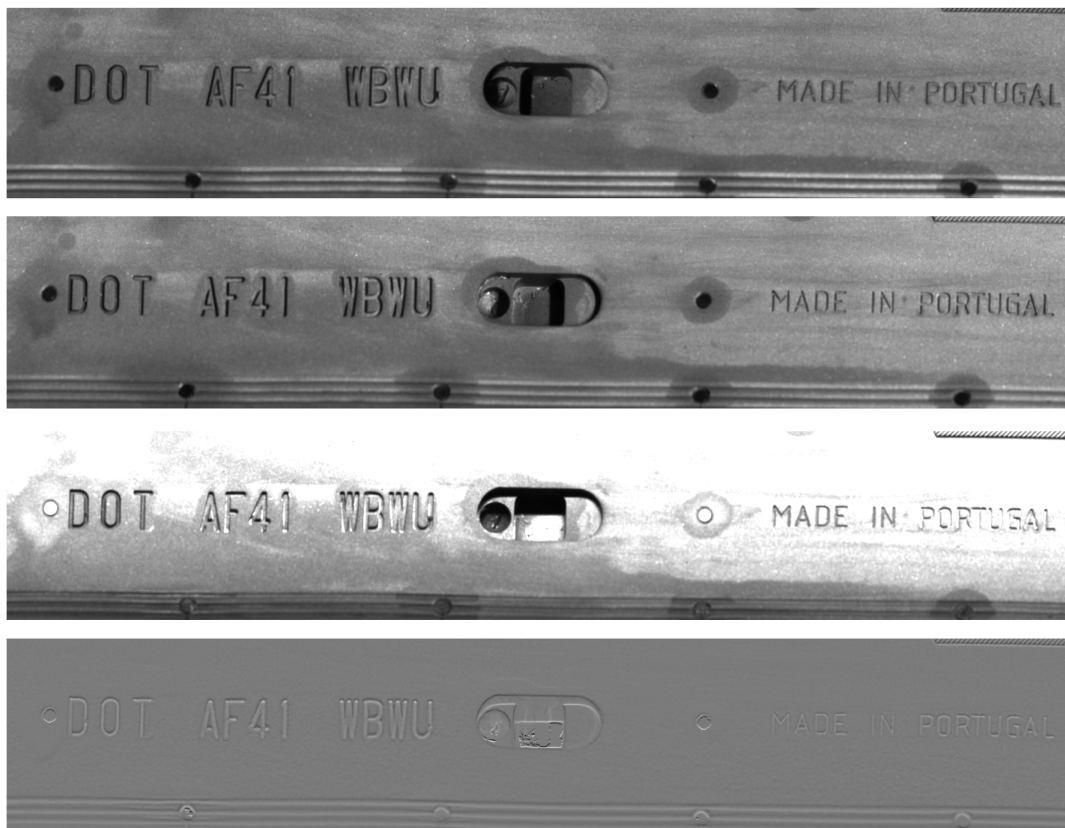


Figura B.11: Imagens resultantes da técnica de *photometric stereo*, de cima para baixo: imagem com iluminação pela esquerda, imagem com iluminação pela direita, imagem com iluminação pelo topo, e imagem final gerada.

Anexo C

Informação extraída do modelo CAD para teste da biblioteca

C.1 Linha 1

Tabela C.1: Atributos da entidade *LINE* identificada com o número 1 no modelo da figura 5.1

color	256
dxftype	LINE
end	(50.00001406018236, 4.2963119995e-06, 0.0)
extrusion	(0.0, 0.0, 1.0)
invisible	0
layer	0
linetype	None
ltscale	1.0
paperspace	False
shadow_mode	None
start	(1.40601823659e-05, 4.2963119995e-06, 0.0)
thickness	0.0
transparency	None
true_color	None

C.2 Circunferência 2

Tabela C.2: Atributos da entidade *CIRCLE* identificada com o número 2 no modelo da figura 5.1

center	(1.40601823659e-05, 4.2963119995e-06, 0.0)
color	256
dxftype	CIRCLE
extrusion	(0.0, 0.0, 1.0)
invisible	0
layer	0
linetype	None
ltscale	1.0
paperspace	False
radius	5.0
shadow_mode	None
thickness	0.0
transparency	None
true_color	None

C.3 Linha 3

Tabela C.3: Atributos da entidade *LINE* identificada com o número 3 no modelo da figura 5.1

color	256
dxftype	LINE
end	(25.00001406018237, 43.30127448553393, 0.0)
extrusion	(0.0, 0.0, 1.0)
invisible	0
layer	0
linetype	None
ltscale	1.0
paperspace	False
shadow_mode	None
start	(50.00001406018236, 4.2963119995e-06, 0.0)
thickness	0.0
transparency	None
true_color	None

C.4 Linha 4

Tabela C.4: Atributos da entidade *LINE* identificada com o número 4 no modelo da figura 5.1

color	256
dxftype	LINE
end	(1.40601823659e-05, 4.2963119995e-06, 0.0)
extrusion	(0.0, 0.0, 1.0)
invisible	0
layer	0
linetype	None
ltscale	1.0
paperspace	False
shadow_mode	None
start	(25.00001406018237, 43.30127448553393, 0.0)
thickness	0.0
transparency	None
true_color	None

C.5 Circunferência 5

Tabela C.5: Atributos da entidade *CIRCLE* identificada com o número 5 no modelo da figura 5.1

center	(50.00001406018236, 4.2963119995e-06, 0.0)
color	256
dxftype	CIRCLE
extrusion	(0.0, 0.0, 1.0)
invisible	0
layer	0
linetype	None
ltscale	1.0
paperspace	False
radius	5.0
shadow_mode	None
thickness	0.0
transparency	None
true_color	None

C.6 Circunferência 6

Tabela C.6: Atributos da entidade *CIRCLE* identificada com o número 6 no modelo da figura 5.1

center	(25.00001406018237, 43.30127448553393, 0.0)
color	256
dxftype	CIRCLE
extrusion	(0.0, 0.0, 1.0)
invisible	0
layer	0
linetype	None
ltscale	1.0
paperspace	False
radius	10.0
shadow_mode	None
thickness	0.0
transparency	None
true_color	None

C.7 Arco de circunferência 7

Tabela C.7: Atributos da entidade *ARC* identificada com o número 7 no modelo da figura 5.1.

center	(25.00001406018236, 4.2963119949e-06, 0.0)
color	256
dxftype	ARC
endangle	135
extrusion	(0.0, 0.0, 1.0)
invisible	0
layer	0
linetype	None
ltscale	1.0
paperspace	False
radius	4.999999999999996
shadow_mode	None
startangle	5.15e-14
thickness	0.0
transparency	None
true_color	None

Anexo D

Significado dos atributos comuns às entidades do formato DXF

Tabela D.1: Atributos comuns às entidades do formato DXF

Atributo	Significado
<i>dxftype</i>	Representa o tipo de entidade, por exemplo <i>CIRCLE</i> ou <i>LINE</i> .
<i>color</i>	Traduz um código de cor com um valor entre 1 e 255. Pode ainda tomar o valor 0 que significa que a cor é definida pelo bloco a que a entidade pertence, ou o valor 256 que indica que a cor é definida pela camada a que a entidade pertence.
<i>extrusion</i>	Vetor unitário que representa o referencial de desenho da entidade respetiva. Este atributo é importante na representação de cada entidade, pelo que lhe é dedicada a subsecção 5.3.1.
<i>invisible</i>	Representa a visibilidade da entidade no modelo final. É útil, por exemplo, quando são desenhadas linhas guia de auxílio ao desenho do modelo, mas que não fazem parte dele. Pode tomar o valor 0 que significa que é visível, ou 1, invisível.
<i>layer</i>	Toma um valor de um número inteiro que define a camada na qual foi desenhada a entidade. Desta maneira, apenas traduz uma informação de como é elaborado o modelo para maior comodidade de desenho, não apresentando qualquer relevo para o desenvolvimento do presente projeto. Isto deve-se ao facto de que apenas é considerada a vista superior do modelo dos moldes, sendo transparente à definição de diferentes camadas.
<i>linetype</i>	Indica o tipo de linha com a qual foi desenhada a entidade, por exemplo <i>DASHED</i> , <i>Continuous</i> ou <i>DASHDOT</i> . Pode ainda ter o valor <i>None</i> que indica que o tipo de linha é definido e partilhado por todas as entidades pertencentes à mesma camada.
<i>ltscale</i>	Aparece sob a forma de um valor decimal que traduz a escala de desenho da entidade.
<i>paperspace</i>	Traduz a vista utilizada no projeto. Pode tomar o valor <i>False</i> que significa que foi utilizada a vista de modelo (<i>modelspace</i>), ou <i>True</i> que implica que o desenho está segundo a vista de <i>paperspace</i> . Esta última é normalmente utilizada quando se deseja imprimir uma representação do modelo, sendo tipicamente acrescentada informação textual que o caracteriza.
<i>shadow_mode</i>	Exprime a utilização de sombras no desenho do modelo, sendo útil apenas para aspetos gráficos.
<i>thickness</i>	Toma um decimal que exprime a espessura de linha com a qual a entidade foi desenhada.
<i>true_color</i>	Este atributo é utilizado para definir o código RGB da cor de desenho da entidade segundo um valor de 24bits sob a forma <i>RRGGBB</i> . Pode apresentar o valor <i>None</i> no caso de não se definir a cor desta forma.

Anexo E

Atributos de entidades do formato DXF

E.1 *ELLIPSE*

Tabela E.1: Exemplo de valores dos atributos de uma entidade do tipo *ELLIPSE*.

center	(216.093848, 170.307495, 0.0)
color	95
dxftype	ELLIPSE
endparam	6.262971159950909
extrusion	(0.0, 0.0, 0.9999999999999998),
invisible	0
layer	125
linetype	Continuous
ltscale	1.0
majoraxis	(-3.367488962974389, 1.004098642686894, 0.0)
paperspace	False
ratio	0.920301
shadow_mode	None
startparam	4.712388980531962
thickness	0.0
transparency	None
true_color	None

E.2 *SPLINE*

Tabela E.2: Exemplo de valores dos atributos de uma entidade do tipo *SPLINE*.

color	155
controlpoints	((205.0463026450309, 502.7185226212512, 0.0), (206.0538736011572, 503.0181911139134, 0.0), (207.0477845541379, 501.0478606001322, 0.0), (206.8304836064834, 503.249167948118, 0.0), (207.9267000255154, 503.5752010874473, 0.0), (207.9905948921683, 499.5297367802853, 0.0), (206.9515096695249, 499.2206954164748, 0.0), (205.0463026450309, 502.7185226212512, 0.0))
degree	1
dxftype	SPLINE
endtangent	None
extrusion	(0.0, 0.0, 1.0)
fitpoints	()
flags	9
invisible	0
knots	(0.0, 0.0, 0.142857143, 0.2857142860000001 0.428571429, 0.5714285710000001, 0.7142857140000001, 0.857142857, 1.0, 1.0)
layer	125
linetype	Continuous
ltscale	1.0
normalvector	(0.0, 0.0, 1.0)
paperspace	False
shadow_mode	None
starttangent	None
thickness	0.0
tol_controlpoint	1,00E-10
tol_fitpoint	1,00E-10
tol_knot	1,00E-07
transparency	None
true_color	None
weights	(1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0)

E.3 POINT

Tabela E.3: Exemplo de valores dos atributos de uma entidade do tipo *POINT*.

color	1
dxftype	POINT
extrusion	(0.0, 0.0, 1.0)
invisible	1
layer	1
linetype	Continuous
ltscale	1.0
paperspace	False
point	(259.829646, 325.276344, 0.0)
shadow_mode	None
thickness	0.0
transparency	None
true_color	None

E.4 INSERT

Tabela E.4: Exemplo de valores dos atributos de uma entidade do tipo *INSERT*.

attribs	[]
attribsfollow	False
col_count	1
col_spacing	0.0
color	23
dxftype	INSERT
extrusion	(0.0, 0.0, 1.0)
insert	(0.0, 0.0, 0.0)
invisible	0
layer	125
linetype	Continuous
ltscale	1.0
name	HATCH97
paperspace	False
rotation	0.0
row_count	1
row_spacing	0.0
scale	(1.0, 1.0, 1.0)
shadow_mode	None
thickness	0.0
transparency	None
true_color	None

Referências

- [1] Carsten Steger, Markus Ulrich, e Christian Wiedemann. *Machine Vision Algorithms and Applications*. Wiley-VCH, 2008.
- [2] H. K. Chethan e G. Hemantha Kumar. Image dewarping and text extraction from mobile captured distinct documents. Em *International Conference and Exhibition on Biometrics Technology, September 2, 2010 - September 4, 2010*, volume 2 de *Procedia Computer Science*, páginas 330–337. Elsevier. URL: <http://dx.doi.org/10.1016/j.procs.2010.11.043>, doi:10.1016/j.procs.2010.11.043.
- [3] Sushruth Shastry, G. Gunasheela, Thejus Dutt, D. S. Vinay, e Sudhir Rao Rupanagudi. "i- a novel algorithm for optical character recognition (ocr). Em *2013 IEEE International Multi Conference on Automation, Computing, Control, Communication and Compressed Sensing, iMac4s 2013, February 22, 2013 - February 23, 2013*, Proceedings - 2013 IEEE International Multi Conference on Automation, Computing, Control, Communication and Compressed Sensing, iMac4s 2013, páginas 389–393. IEEE Computer Society. URL: <http://dx.doi.org/10.1109/iMac4s.2013.6526442>, doi:10.1109/iMac4s.2013.6526442.
- [4] S. L. Wasankar, H. Mahajan, D. Deshmukh, e H. Munot. Self intelligence with text recognition. Em *2010 International Conference on Signal and Image Processing (ICSIP 2010), 15-17 Dec. 2010*, 2010 International Conference on Signal and Image Processing (ICSIP 2010), páginas 521–5. IEEE. URL: <http://dx.doi.org/10.1109/ICSIP.2010.5697531>, doi:10.1109/ICSIP.2010.5697531.
- [5] Vinh-Tiep Nguyen, Khanh-Duy Le, Minh-Triet Tran, e Anh-Duc Duong. Fast template matching using pruning strategy with haar-like features. Em *2012 4th International Conference on Intelligent Human-Machine Systems and Cybernetics, IHMSC 2012, August 26, 2012 - August 27, 2012*, volume 2 de *Proceedings of the 2012 4th International Conference on Intelligent Human-Machine Systems and Cybernetics, IHMSC 2012*, páginas 246–251. IEEE Computer Society. URL: <http://dx.doi.org/10.1109/IHMSC.2012.155>, doi:10.1109/IHMSC.2012.155.
- [6] Truong Dai-Duong, Nguyen Vinh-Tiep, Duong Anh-Duc, Ngoc Chau-Sang Nguyen, e Tran Minh-Triet. Realtime arbitrary-shaped template matching process. Em *2012 12th International Conference on Control Automation Robotics amp; Vision (ICARCV 2012), 5-7 Dec. 2012*, 2012 12th International Conference on Control Automation Robotics Vision (ICARCV 2012), páginas 1407–12. IEEE. URL: <http://dx.doi.org/10.1109/ICARCV.2012.6485395>, doi:10.1109/ICARCV.2012.6485395.
- [7] T. S. Newman e A. K. Jain. A system for 3d cad-based inspection using range images. *Pattern Recognition*, 28(10):1555–74, 1995. URL: [http://dx.doi.org/10.1016/0031-3203\(95\)00028-X](http://dx.doi.org/10.1016/0031-3203(95)00028-X), doi:10.1016/0031-3203(95)00028-X.

- [8] Z. Zhang e M. Sarhadi. An integrated cad/cam system for automated composite manufacture. Em *12th International Conference on Computer Aided Production Engineering, 5-7 Aug. 1996*, volume 61 de *J. Mater. Process. Technol. (Switzerland)*, páginas 104–9. Elsevier. URL: [http://dx.doi.org/10.1016/0924-0136\(96\)02473-9](http://dx.doi.org/10.1016/0924-0136(96)02473-9), doi:10.1016/0924-0136(96)02473-9.
- [9] Kim Minsung, Choi Woosung, Kim Byung-Chul, Kim Hokyeong, Seol Jae Hun, Woo Jonghun, e Ko Kwang Hee. A vision-based system for monitoring block assembly in ship-building. *Computer-Aided Design*, 59:98–108, 2015. URL: <http://dx.doi.org/10.1016/j.cad.2014.09.001>, doi:10.1016/j.cad.2014.09.001.
- [10] I. Viana, F. Bugarin, N. Cornille, e J. J. Orteu. Cad-guided inspection of aeronautical mechanical parts using monocular vision. Em *Twelfth International Conference on Quality Control by Artificial Vision 2015, 3-5 June 2015*, volume 9534 de *Proc. SPIE (USA)*, página 95340I (9 pp.). SPIE. URL: <http://dx.doi.org/10.1117/12.2182922>, doi:10.1117/12.2182922.
- [11] Jiahua Wu. *Rotation Invariant Classification of 3D Surface Texture Using Photometric Stereo*. Tese de doutoramento, Heriot-Watt University, 2003.
- [12] R Deepa, S Usha, e P V Shashi Kumar. Automated vision inspection system for a plastic injection mould component. *5 th International 26 th All India Manufacturing Technology, Design and Research Conference (AIMTDR 2014)*, Dezembro 2014.
- [13] Tadhg Brosnan e Da-Wen Sun. Improving quality inspection of food products by computer vision—a review. *Journal of Food Engineering*, 61:3–16, 2004.
- [14] C J Smith e K Adendorff. Advantages and limitations of an automated visual inspection system. *SA Journal of Industrial Engineering*, páginas 27–36, Junho 1991.
- [15] Yung-Chun Liu, Yu-Lu Hsu, Yung-Nien Sun, Song-Jan Tsai, Chiu-Yi Ho, e Chung-Mei Chen. A computer vision system for automatic steel surface inspection. *5th IEEE Conference on Industrial Electronics and Applications (ICIEA 2010)*, Junho 2010.
- [16] David Page, Andreas Koschan, Sophie Voisin, Ngozi Ali, e Mongi Abidi. 3d cad model generation of mechanical parts using coded-pattern projection and laser triangulation systems. *Assembly Automation*, páginas 230–238, 2005.
- [17] Mai Moussa Chetima e Pierre Payeur. Automated tuning of a vision-based inspection system for industrial food manufacturing. *IEEE International Instrumentation and Measurement Technology Conference*, 2012.
- [18] Yogitha. S e Sakthivel.P. A distributed computer machine vision system for automated inspection and grading of fruits. *International Conference on Green Computing Communication and Electrical Engineering (ICGCCEE)*, Março 2014.
- [19] Ana Maria Mendonça. Sistemas baseados em visão - condicionamento do ambiente de aquisição, Acedido 24/02/2016.
- [20] J. P. Lewis. Fast template matching. *Vision Interface 95*, páginas 120–123, 1995.
- [21] Changxin Gao e Nong Sang. Biologically inspired template matching using scene context. Em *2011 IEEE Computer Society Conference on Computer Vision and Pattern*

- Recognition Workshops, CVPRW 2011, June 20, 2011 - June 25, 2011*, IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops. IEEE Computer Society. URL: <http://dx.doi.org/10.1109/CVPRW.2011.5981776>, doi:10.1109/CVPRW.2011.5981776.
- [22] Wong Ze-Hao, K. Abdullah, e C. J. Wong. Template matching using multiple templates weighted normalised cross correlation. Em *2014 IEEE Symposium on Computer Applications amp; Industrial Electronics (ISCAIE), 7-8 April 2014*, 2014 IEEE Symposium on Computer Applications and Industrial Electronics (ISCAIE), páginas 131–5. IEEE. URL: <http://dx.doi.org/10.1109/ISCAIE.2014.7010224>, doi:10.1109/ISCAIE.2014.7010224.
- [23] J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6):679–98, 1986. URL: <http://dx.doi.org/10.1109/TPAMI.1986.4767851>, doi:10.1109/TPAMI.1986.4767851.
- [24] Shi Jianbo e C. Tomasi. Good features to track. Em *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 21-23 June 1994*, Proceedings 1994 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No.94CH3405-8), páginas 593–600. IEEE Comput. Soc. Press. URL: <http://dx.doi.org/10.1109/CVPR.1994.323794>, doi:10.1109/CVPR.1994.323794.
- [25] Paul Viola e Michael Jones. Robust real-time object detection. 2001.
- [26] J. H. M. Byne e J. A. D. W. Anderson. A cad-based computer vision system. *Image and Vision Computing*, 16(8):533–9, 1998. URL: [http://dx.doi.org/10.1016/S0262-8856\(98\)00100-0](http://dx.doi.org/10.1016/S0262-8856(98)00100-0), doi:10.1016/S0262-8856(98)00100-0.
- [27] Jin-hwa Choi, Byung-cheol Jeon, Gun-hee Kim, Bong-cheol Shin, Honghee Lee, Myeong-woo Cho, Eun-je Seong, e Dong-sam Park. A web-based on-machine mould matching and measurement system based on cad/cam/cai integration. *Journal of Zhejiang University (Science)*, 7(9):1474–81, 2006. URL: <http://dx.doi.org/10.1631/jzus.2006.A1474>, doi:10.1631/jzus.2006.A1474.
- [28] Nguyen Duc Dung, Ko Jae Pil, e Jeon Jae Wook. Determination of 3d object pose in point cloud with cad model. Em *2015 21st Korea-Japan Joint Workshop on Frontiers of Computer Vision (FCV), 28-30 Jan. 2015*, 2015 21st Korea-Japan Joint Workshop on Frontiers of Computer Vision (FCV), página 6 pp. IEEE. URL: <http://dx.doi.org/10.1109/FCV.2015.7103725>, doi:10.1109/FCV.2015.7103725.
- [29] M. Rodrigues, M. Kormann, C. Schuhler, e P. Tomek. An intelligent real time 3d vision system for robotic welding tasks. Em *2013 9th International Symposium on Mechatronics and its Applications (ISMA), 9-11 April 2013*, Proceedings of the 2013 9th International Symposium on Mechatronics and its Applications (ISMA), página 6 pp. IEEE. URL: <http://dx.doi.org/10.1109/ISMA.2013.6547393>, doi:10.1109/ISMA.2013.6547393.
- [30] Chen Ching-Chang, Wen Chun-Ming, C. Y. E. Wang, e Chen Yi-Cheng. A visual guided robot using cad models and fuzzy controller for industrial manipulation. Em *2014 CACS International Automatic Control Conference (CACS), 26-28 Nov. 2014*, 2014 CACS International Automatic Control Conference (CACS), páginas 87–90. IEEE. URL: <http://dx.doi.org/10.1109/CACS.2014.7097167>, doi:10.1109/CACS.2014.7097167.

- [31] I. Vaamonde, A. Souto-Lopez, e A. Garcia-Diaz. Fast and intuitive programming of adaptive laser cutting of lace enabled by machine vision. Em *Industrial Laser Applications Symposium (ILAS 2015), 17-18 March 2015*, volume 9657 de *Proc. SPIE (USA)*, página 965707 (6 pp.). SPIE. URL: <http://dx.doi.org/10.1117/12.2183791>, doi:10.1117/12.2183791.
- [32] R. Ahmad, S. Tichadou, e J. Y. Hascoet. New computer vision based snakes and ladders algorithm for the safe trajectory of two axis cnc machines. *Computer Aided Design*, 44(5):355–66, 2012. URL: <http://dx.doi.org/10.1016/j.cad.2011.12.008>, doi:10.1016/j.cad.2011.12.008.
- [33] R. Ahmad, S. Tichadou, e J. y Hascoet. 3d safe and intelligent trajectory generation for multi-axis machine tools using machine vision. *International Journal of Computer Integrated Manufacturing*, 26(4):365–85, 2013. URL: <http://dx.doi.org/10.1080/0951192X.2012.717720>, doi:10.1080/0951192X.2012.717720.
- [34] Robert J. Woodham. Photometric method for determining surface orientation from multiple images. *Optical Engineering*, 19(1), 1980. doi:10.1117/12.7972479.
- [35] Cao Minhao e Zhong Fei. Inspection system for position of frame line in the mould based on machine vision. *Advanced Materials Research*, páginas 1093–1097, 2013.
- [36] K. Barnard, P. Duygulu, e D. Forsyth. Recognition as translating images into text. Em *Internet Imaging IV, 21-22 Jan. 2003*, volume 5018 de *Proc. SPIE - Int. Soc. Opt. Eng. (USA)*, páginas 168–78. SPIE-Int. Soc. Opt. Eng. URL: <http://dx.doi.org/10.1117/12.478427>, doi:10.1117/12.478427.
- [37] G. N. Sazaklis, E. M. Arkin, J. S. B. Mitchell, e S. S. Skiena. Geometric decision trees for optical character recognition. Em *Proceedings of Thirteenth Annual ACM Symposium on Computational Geometry, 4-6 June 1997*, Proceedings of the Thirteenth Annual Symposium on Computational Geometry, páginas 394–6. ACM. URL: <http://dx.doi.org/10.1145/262839.263020>, doi:10.1145/262839.263020.
- [38] M. S. Nixon, X. U. Liu, C. Direkoglu, e D. J. Hurley. On using physical analogies for feature and shape extraction in computer vision. *Computer Journal*, 54(1):11–25, 2011. URL: <http://dx.doi.org/10.1093/comjnl/bxp070>, doi:10.1093/comjnl/bxp070.
- [39] Mark Nixon Alberto Aguado. *Feature Extraction and Image Processing*. Academic Press/Elsevier, Hungaria, 2 edição, 2008.
- [40] H. Golnabi e A. Asadpour. Design and application of industrial machine vision systems. *Robotics and Computer Integrated Manufacturing*, 23(6):630–637, 2007.
- [41] Económico TV – Informação com Valor. "como se faz"um pneu. Acedido: 26/11/2015. URL: <https://www.youtube.com/watch?v=mn1MHFcfd4g>.
- [42] Ying Wu. Radiometry, brdf and photometric stereo. *EECS 432-Advanced Computer Vision Notes Series 3*.
- [43] Halcon's imaging software technique 'photometric stereo' explained, 2014. URL: <http://multipix.com/support-articles/imaging-software-technique-photometric-stereo-explained/>.