

RETINAL TRACKING

Ivânia Margarida Magalhães Pereira

UNIVERSIDADE DE COIMBRA

2007



DEPARTAMENTO DE FÍSICA

AIBILI

associação para
investigação biomédica e
inovação em luz e imagem

CENTRO DE NOVAS TECNOLOGIAS PARA

A MEDICINA

Retinal Tracking

Ivânia Margarida Magalhães Pereira

Relatório de Projecto da Licenciatura em Engenharia Biomédica

ORIENTADORA

Professora Doutora Filomena Botelho

Instituto de Biofísica e Biomatemática

IBILI – Instituto Biomédico de Investigação em Luz e Imagem – Faculdade de Medicina da
Universidade de Coimbra

SUPERVISOR

Engenheiro Rui Bernardes

Centro de Novas Tecnologias para a Medicina – Associação para a Investigação Biomédica
em Luz e Imagem

IBILI – Instituto Biomédico de Investigação em Luz e Imagem – Faculdade de Medicina
da Universidade de Coimbra

É preciso suportar duas ou três larvas se quiser saber como são as borboletas. Dizem que são muito bonitas...

Saint – Exupéry

Resumo

A visão é, provavelmente, o sentido mais importante no ser humano. Desenvolvido ao longo da evolução da espécie, tornou-se a forma de percepção por excelência do mundo circundante. É, por isso, óbvia a sua importância. As técnicas para o tratamento de doenças que afectam a visão têm sido desenvolvidas em larga escala ao longo dos últimos anos, especialmente ao nível de tratamento informático de imagem, o que permite a aplicação de novas metodologias na imagiologia médica. No entanto, estes métodos são ainda limitados pela ocorrência de movimentos rápidos e aleatórios do olho – movimentos sacádicos –, que servem para uma fixação da imagem na fóvea. É proposto, neste trabalho, um novo sistema de detecção do movimento translacional do globo ocular, a partir dos pontos característicos da imagem da retina, para posterior acoplamento a um sistema de disparo laser. Num trabalho futuro, a conjugação dos movimentos oculares com o movimento do sistema de disparos laser facilitará o tratamento de retinopatias ou outras enfermidades da visão, onde as aplicações laser possam ser aconselhadas como tratamento, ou para acompanhamento clínico de evolução de doenças.

Palavras-chave: retina, movimentos sacádicos, *tracking*, pontos característicos.

Agradecimentos

Queria deixar o meu mais sincero agradecimento a todas as pessoas envolvidas directa ou indirectamente neste projecto.

Aos Engenheiros Rui Bernardes e Pedro Baptista e à Prof. Dra. Filomena Botelho, pela orientação e apoio dado ao longo deste ano; sem eles, este projecto não seria possível. Aos restantes elementos do CNTM – AIBILI, que, de uma forma ou de outra, ajudaram na concretização deste projecto.

A todos os professores que me acompanharam ao longo destes anos, pela partilha de conhecimentos e ajuda prestada.

À minha família, que com esforço e muita paciência, me ensinou que desistir seria fácil demais e que acompanhou e apoiou ao longo destes anos.

Aos meus amigos, que me ajudam a encontrar um pedacinho de céu em cada sorriso e em cada lágrima e que, mesmo quando tudo parece perdido, nos ensinam a encarar o futuro com uma esperança renovada.

Índice

<i>Resumo</i>	<i>iv</i>
<i>Agradecimentos</i>	<i>v</i>
1. Introdução	- 8 -
1.1. Motivação	- 9 -
1.2. Objectivos	- 10 -
1.3. Organização do relatório	- 10 -
PARTE I	- 12 -
FUNDAMENTAÇÃO TEÓRICA	- 12 -
2. O olho e os seus constituintes	- 13 -
2.1. Anatomia geral do olho humano	- 13 -
2.2. Fisiologia da retina	- 16 -
2.3. Formação da imagem no olho	- 19 -
2.4. Movimentos oculares	- 19 -
2.5. Doenças degenerativas da retina	- 20 -
3. Representação de imagens	- 23 -
3.1. Espaços de cores	- 24 -
3.2. A imagem em Matlab	- 28 -
3.3. Regiões de uma imagem	- 31 -
3.4. Identificação de arestas	- 31 -
3.4.1. Filtragem	- 32 -
3.4.2. Diferenciação	- 33 -
3.4.3. Detecção	- 35 -
4. Retinal Tracking: State of Art	- 38 -
4.1. Detecção dos pontos característicos da imagem	- 39 -
4.1.1. Aproximações multi- escalares	- 39 -
4.1.2. Aproximações de matched filters	- 40 -
4.1.3. Detecção da linha de centro (aproximações a esqueletos)	- 40 -
4.1.4. Aproximações baseadas em ridges	- 40 -
4.1.5. Modelos paramétricos deformáveis – contornos activos (Snakes)	- 42 -
4.1.6. Esquemas matemáticos morfológicos	- 43 -
4.1.7. Aproximação por crescimento de região	- 44 -
4.1.8. Correspondência de templates	- 45 -
4.2. Determinação do movimento	- 45 -
4.2.1. Modelos probabilísticos	- 46 -
4.2.2. Subtracção do plano de fundo	- 47 -
4.2.3. Métodos estatísticos	- 47 -
4.2.4. Fluxo óptico	- 48 -
PARTE II	- 54 -
DESENVOLVIMENTO	- 54 -
5. Retinal Tracking: introdução de conceitos	- 55 -

5.1.	Derivadas parciais de segunda ordem	- 55 -
5.2.	Correlação	- 57 -
5.2.1.	Formulação matemática	Erro! Marcador não definido.
5.3.	Transformações entre imagens	- 60 -
5.3.1.	Transformações Euclidianas	- 60 -
5.3.2.	Transformações de similaridade	- 61 -
5.3.3.	Transformações afins	- 61 -
5.3.4.	Transformações projectivas	- 63 -
5.4.	RANSAC	- 66 -
5.4.1.	Descrição do algoritmo RANSAC	- 66 -
5.4.2.	Variáveis do processo	- 68 -
5.4.3.	Optimização	- 71 -
6.	<i>Descrição do software</i>	- 75 -
6.1.	Determinação da amplitude máxima de movimentos entre <i>frames</i>	- 75 -
6.2.	Sequência artificial de vídeo	- 76 -
6.3.	Sequência real de vídeo	- 77 -
6.3.1.	Considerações iniciais	- 77 -
6.3.2.	Detecção dos pontos característicos da imagem	- 78 -
6.3.3.	Correspondência de pontos entre duas frames consecutivas	- 79 -
6.3.4.	Implementação do algoritmo de RANSAC	- 81 -
6.4.	Ficheiros do programa	- 83 -
7.	<i>Análise de resultados</i>	- 84 -
7.1.	Sequência artificial de imagens	- 84 -
7.2.	Sequência real de vídeo	- 85 -
8.	<i>Conclusões e trabalho futuro</i>	- 88 -
<i>Anexo A. Relações básicas entre píxeis</i>		- 90 -
1.	Vizinhanças de um píxel	- 90 -
2.	Escolher um filtro de suavização	- 90 -
3.	Enriquecimento de arestas por diferenciação de valores discretos	- 91 -
4.	Detecção direccionada de arestas	- 93 -
5.	Detecção hierárquica	- 94 -
6.	Detecção por <i>clusters</i> – K-forms	- 94 -
7.	Fronteiras binárias	- 95 -
8.	Detectores de gradiente	- 96 -
9.	Detector de Roberts	- 96 -
10.	Detector de Prewitt	- 97 -
11.	Detector de Sobel	- 98 -
12.	Detector Laplaciano	- 99 -
13.	Laplaciano do operador Gaussiano	- 100 -
14.	Detector de Canny	- 101 -
<i>Anexo B. Vectores e valores próprios</i>		- 105 -
<i>Anexo C: Software desenvolvido</i>		- 108 -

1. Introdução

Os olhos são o espelho da alma. É através deles que conseguimos perceber o mundo que nos rodeia, desenvolver sensações e criar uma ideia mental do ambiente. As imagens do mundo exterior que são processadas no cérebro têm como entrada a retina, um dos mais importantes tecidos que constitui a camada posterior do olho. As células que constituem os seus epitélios são especializadas na transmissão de sinais eléctricos ao córtex cerebral. A retina recebe um estímulo directo da luz e converte a imagem em impulsos eléctricos, que envia ao cérebro através do nervo óptico. Cada metade das fibras deste nervo cruza para o lado oposto no quiasma óptico, uma área situada inferiormente à zona frontal do cérebro. Os feixes voltam a unir-se antes de chegarem ao córtex cerebral, onde se dá a interpretação dos impulsos e a codificação de informação. ^[4]

O olho humano, em geral, encontra-se, porém, sujeito a várias doenças ou malformações, que podem conduzir à cegueira. Entre as várias enfermidades destacam-se cataratas, descolamentos de retina, degenerescências, tumores, entre outros, que podem comprometer o fluxo sanguíneo (pelo colapso de vasos ou neovascularizações) até aos tecidos e a acuidade visual. É, por isso, essencial conhecer os constituintes e funcionamento do olho, para poder saber até que ponto estas doenças vão sacrificar a formação de imagens e a percepção do ambiente.

O tratamento de algumas enfermidades pode ser feito por fotocoagulação laser ou técnicas de ablação. No primeiro, um feixe laser é disparado na região afectada, eliminando neovascularizações ou estancando hemorragias, enquanto nas técnicas de ablação, aplicadas, por exemplo, na correcção da curvatura do cristalino, a estrutura é cortada, com auxílio de um laser, conferindo-lhe uma curvatura diferente, que permite a focagem correcta dos objectos.

Na aplicação dos disparos laser, é necessário ter em especial atenção os movimentos sacádicos do olho. Estes caracterizam-se pelas suas elevadas amplitudes, que variam entre 600 e 900° por segundo, e permitem fixar o objecto na fóvea. ^{[1], [2], [3]} Estes movimentos vão contribuir com um factor de deslocamento entre os disparos laser, influenciando a posição em que os vasos se encontram. A tarefa de fazer incidir o feixe numa localização específica encontra-se, então, comprometida, inviabilizando alguns tipos de cirurgias (em que os

disparos se tenham de realizar em localizações muito específicas, por exemplo) ou um planeamento exacto da estratégia de tratamento, para a qual seja necessário determinar localizações específicas de características da imagem.

Este problema pode ser ultrapassado recorrendo a metodologias de visão computacional, como a aplicação de filtros às imagens, reconhecimento de estruturas ou métodos estatísticos, permitindo inferir acerca do movimento entre imagens. O resultado obtido servirá de base à orientação dos disparos do laser e à monitorização da posição das características a realçar na imagem.

Contudo, a complexidade das estruturas que compõem a imagem, bem como o tempo de computação subjacente à sua detecção, torna a conjugação do sistema laser com o movimento ocular uma tarefa quase impossível de realizar em tempo real. Isto é, o tempo de computação necessário para a detecção do movimento é muito superior ao tempo de aquisição das imagens. Há, então, uma restrição da aplicação do sistema à monitorização do paciente, ou simplesmente ao acompanhamento da evolução de doenças, sem que possam servir de apoio prático aos procedimentos cirúrgicos actuais.

Uma tentativa para diminuir o tempo de processamento reside numa aquisição em vídeo do movimento do olho do paciente durante a aplicação dos disparos laser, e, a partir daí, numa quantificação do deslocamento entre *frames* consecutivas do vídeo. Para isso, foram implementados métodos matemáticos de cálculo simples, que permitam o acompanhamento entre *frames* de um número limitado de pontos característicos, para correlação destes e avaliação das respectivas correspondências e transformações entre imagens.

Podendo detectar o movimento em tempo real, ou seja, com uma velocidade de processamento de imagem igual à da aquisição vídeo, será possível uma sincronização dos disparos laser com o movimento ocular.

1.1. Motivação

A crescente evolução das enfermidades que afectam a população mundial e a necessidade de técnicas cirúrgicas cada vez mais avançadas servem de mote à conjugação de diversos ramos da ciência, como a medicina e as ciências computacionais. Este “casamento” entre ideologias favorece o desenvolvimento de novos *softwares* clínicos, que contribuem cada vez mais para

uma melhoria na qualidade de vida da população, conduzindo a uma modernização da medicina e acompanhamento das crescentes necessidades do ser humano.

1.2. Objectivos

Este relatório foi realizado como parte integrante da disciplina de Projecto, do curso de Licenciatura em Engenharia Biomédica.

O objectivo principal deste trabalho é a quantificação do deslocamento sofrido entre *frames* consecutivas de uma aquisição vídeo. Ou seja, partindo de um conjunto de imagens adquiridas de forma contínua com uma câmara Sony 3CCD ExwaveHAD, acoplada a uma lâmpada de fenda TAKAGI S10-15 e a um *frame grabber* Coreco Imaging PC-RGB, pretende detectar-se a translação existente entre duas imagens consecutivas, à mesma velocidade que estas são adquiridas.

1.3. Organização do relatório

Este relatório encontra-se dividido em duas partes. A primeira aborda conceitos matemáticos de imagem e os tipos de trabalhos já desenvolvidos a nível de tratamento de imagem e acompanhamento de cenas, como avaliação de possíveis alternativas a implementar, descrevendo trabalhos desenvolvidos e métodos utilizados. A segunda parte analisa o algoritmo desenvolvido para a detecção do movimento, com explicação teórica das metodologias abordadas, descrição do algoritmo desenvolvido e respectivos resultados obtidos. No final, apresenta-se uma conclusão ao trabalho e futuros desenvolvimentos.

Em anexo, apresentam-se alguns desenvolvimentos matemáticos dos conceitos abordados ao longo do relatório e o algoritmo desenvolvido para a detecção dos movimentos.

A bibliografia consultada em cada capítulo encontra-se no fim do mesmo, para uma maior facilidade de localização do tema explorado.

Referências bibliográficas

[1] Aitken, V.C.. "Retinal motion and microvascular Pulsation Measurements from SLO image sequences." IMTC 2005 instrumentation and measurement 17 - 19 May 2005 222 - 227. 26 Sep 2006

<ieeexplore.ieee.org/iel5/10678/33710/01604081.pdf?arnumber=1604081>.

[2] Markow, Michael et al. "Real time algorithm for retinal tracking." IEEE Transactions on biomedical engineering 40, num 12. December 1993 1269 - 1281. 16 Sep 2006

<ieeexplore.ieee.org/iel5/10/6416/00250583.pdf?arnumber=250583>.

[3] Bridgeman, Bruce . "A Theory of visual stability across saccadic eye movements" Behavioral and Brain Sciences 171994 247-292. 24 May 2007

<http://www.bbsonline.org/Preprints/OldArchive/bbs.bridgeman.html>

[4] MacKay, William. Neurofisiologia sem Lágrimas. Lisboa: Fundação Calouste Gulbenkian, 1999

PARTE I

FUNDAMENTAÇÃO TEÓRICA

2. O olho e os seus constituintes

O olho humano apresenta-se como uma estrutura altamente complexa, capaz da transformação da imagem em impulsos eléctricos, transmitidos ao cérebro.

É na fóvea que se dá a focagem da imagem, através de movimentos do globo ocular, que permitem uma aquisição contínua da imagem, e é nas camadas de fotorreceptores que se inicia a transmissão de impulsos.

No entanto, todas as estruturas que compõem o olho humano encontram-se vulneráveis a doenças, que podem provocar malformações e conduzir à cegueira.

Para permitir o diagnóstico destas enfermidades e o desenvolvimento de novas formas de tratamento, é necessário conhecer as estruturas que compõem o olho humano e o modo de formação da imagem, bem como os movimentos por ela responsáveis, e as doenças que podem afectar a visão.

2.1. Anatomia geral do olho humano

O olho humano é formado por 3 camadas concêntricas, conforme ilustra a figura.

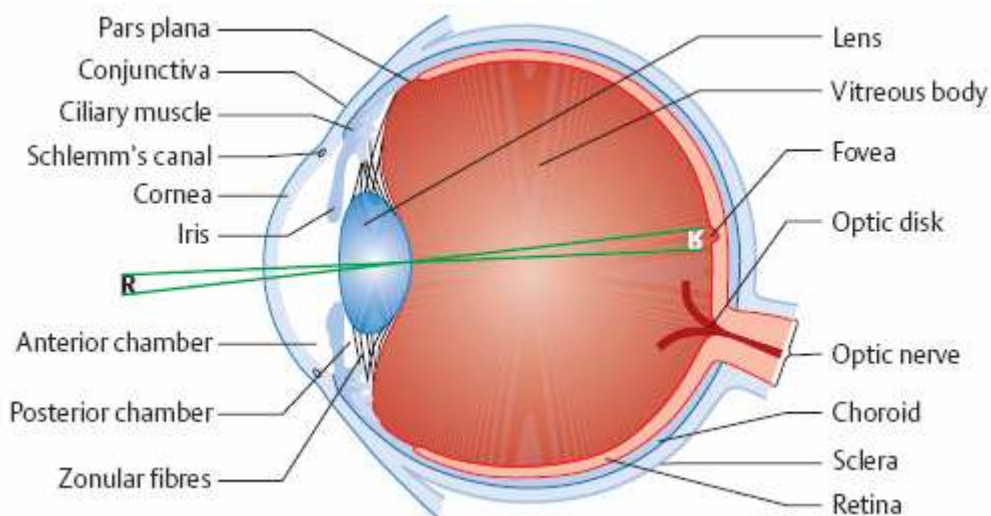


Figura 2.1. Anatomia geral do olho humano. [1].

A primeira camada é designada camada externa, que dá a forma arredonda ao globo ocular e contribui para manter a pressão no seu interior, e é constituída pela córnea e esclera.

A córnea é uma estrutura transparente, responsável, em grande parte, pelo poder refractivo do olho. É composta por cinco camadas de células: o epitélio, a conjuntiva, a membrana de Bowman, o estroma, a membrana de Descemet e o endotélio córneo. Além disso, cobre a pupila e a íris e protege os tecidos e os humores intraoculares. A sua face anterior encontra-se coberta por uma película lacrimal e tem uma forma convexa, com uma curvatura mais pronunciada que a íris; a sua face posterior está em contacto com o humor aquoso, constituindo a parede da camada anterior do olho. A córnea apresenta-se também como uma estrutura altamente enervada; uma alteração na distribuição espacial das terminações nervosas pode conduzir à *queratitis* neurotrófica ou neuroparalítica. As patologias associadas a este membro dependem de atrofas na sua vascularização, ou da sua transparência, afectando a acuidade visual. ^{[4], [6], [10]}

A esclera, contínua à córnea, constitui o que vulgarmente se chama a “parte branca do olho”. É uma membrana fibrosa, de onde se destacam três camadas, denominadas de episclera, estroma de colagénio e lâmina fusca. Serve de protecção aos tecidos intraoculares, suporta a tensão muscular e mantém a forma do globo. Posteriormente, encontra-se atravessada pelo nervo óptico e pelos vasos sanguíneos. ^{[4], [10]}

O estrato intermédio do olho humano é dividido em duas partes: a anterior, que integra a íris e o corpo ciliar, e a posterior, da qual faz parte a coróide. ^[5]

A íris é um músculo situado na parte mais anterior da camada intermédia, na forma de um disco, perfurado centralmente pela pupila. O controlo da quantidade de luz que entra no olho é feito por este músculo, que também é responsável pelo tamanho da pupila. Encontra-se imersa no humor aquoso e a sua superfície anterior está dividida numa zona pupilar central e numa zona ciliar periférica. A coloração que normalmente apresenta depende da concentração de melanina, uma proteína existente na zona anterior; se a sua concentração é baixa, a íris possui um tom azulado, e se for elevada possui um tom castanho. A base da íris insere-se no corpo ciliar, e é esta a zona de espessura mínima. ^{[4], [5], [10]}

O corpo ciliar é uma estrutura triangular cuja superfície interna, exposta ao humor aquoso, é caracterizada pela presença de cílios, revestidos pelo epitélio ciliar. Encontra-se rodeado pela íris, retina e esclera e é responsável pela nutrição e acomodação do globo (ajuste das

capacidades refractivas, por alteração da curvatura do cristalino, e controlada pelo músculo ciliar, que se encontra na parede do corpo ciliar) e pela secreção do humor aquoso, um fluido com uma composição semelhante à do sangue, excepto na concentração de proteínas (que é muito mais baixa). Sobre esta superfície encontram-se também os ligamentos responsáveis pelo suporte do cristalino – fibras zonulares. ^{[4], [6], [10]}

A coróide é uma área rica em células pigmentadas, o que confere um papel de filtro à luz. A sua rede vascular desenvolvida torna-a numa das estruturas responsáveis pela nutrição do olho. Encontra-se situada entre a esclera e a retina, separada da primeira pelo espaço supracoroideio. ^{[4], [10]}

A terceira camada encontra-se numa posição posterior ao corpo ciliar e é constituída pela retina, um tecido complexo que será descrito na secção seguinte. ^[5]

Estas três regiões encontram-se separadas por câmaras preenchidas com fluidos. Entre a córnea e a íris encontra-se a primeira câmara, a câmara anterior, e entre a íris e as fibras zonulares encontra-se a câmara posterior. A câmara vítrea situa-se entre a face posterior do cristalino e a retina e contém no seu interior o vítreo, um fluido viscoso e transparente. Caso esta última câmara degenerere e colapse, ocorre uma união à retina, que pode levar ao seu descolamento. ^{[4], [6]}

Há ainda a considerar duas estruturas: a conjuntiva e o cristalino, responsáveis pela protecção e focagem, respectivamente.

O cristalino é uma lente biconvexa, de convergência variável, sem enervação nem rede vascular, responsável pela acomodação. É formado por um conjunto de camadas concêntricas de células fibrosas, que aderem ao corpo ciliar, e é caracterizado por ter um crescimento contínuo ao longo da vida. A tensão no cristalino está relacionada com a contracção do músculo ciliar, onde se inserem os ligamentos suspensores. Quando ocorre a contracção do músculo ciliar, há um relaxamento dos ligamentos, e o cristalino tende a tornar-se mais convexo, aumentando a sua potência. No relaxamento muscular, os ligamentos são puxados para fora e o cristalino torna-se menos esférico. Além disso, ocorrem dois processos paralelos: a convergência, para fusão das imagens retinianas e a *miosis*, que diminui as aberrações nas alterações de curvatura do cristalino. ^{[4], [6], [10]}

A conjuntiva encontra-se ligada à córnea e à palpebra e possui uma pigmentação amarelada, que tende a aumentar com a idade, podendo levar à formação de cataratas. As suas funções são de protecção, quer por processos mecânicos (no epitélio e secreções), quer por fenómenos inflamatórios (de forma imunológica ou por substâncias antibacterianas) e mobilidade, permitindo os movimentos oculares. [4],[10]

2.2. Fisiologia da retina

A retina é parte integrante da camada posterior do olho, uma camada neurosensorial, onde se dá início ao processo da visão, recolhendo e transmitindo as informações codificadas pelos estímulos visuais. É uma estrutura parcialmente transparente, que limita internamente o vítreo. [2]

A sua nutrição é assegurada por duas vias sanguíneas: os vasos da coróide, responsáveis pela maioria do fluxo, e a artéria central retiniana. Os vasos da coróide abastecem a parte mais exterior da retina, enquanto que a artéria central fornece o sangue necessário à sobrevivência das camadas mais interiores. [2]

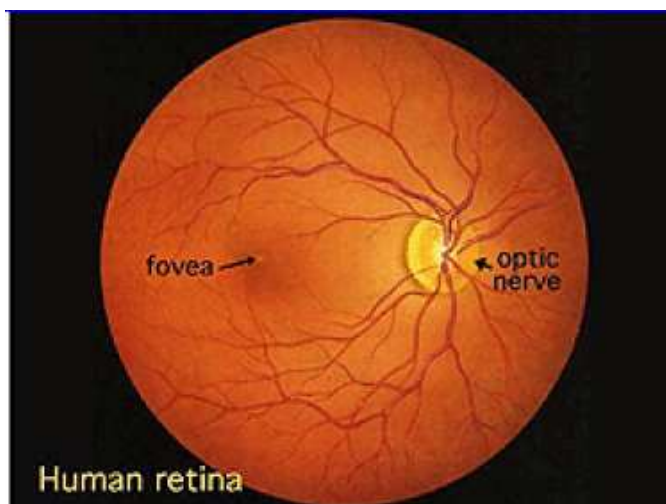


Figura 2.2. A retina humana. [2].

A retina pode considerar-se dividida em dois epitélios: o epitélio pigmentado e o neuroepitélio, constituído por nove camadas. O epitélio pigmentado é composto por uma única fileira de células, que aderem à coróide e que têm finas prolongamentos entre os fotorreceptores adjacentes, com uma elevada composição de melanina e responsáveis pelo aspecto granuloso do fundo do olho. O epitélio pigmentado é responsável pela absorção da

radiação luminosa (evitando a ocorrência de uma iluminação excessiva dos fotorreceptores), pela renovação destas células e pelas trocas metabólicas. Por sua vez, o neuroepitélio é constituído por nove camadas, distinguidas, do interior para o exterior, como a camada de fotorreceptores (responsáveis pela absorção das radiações luminosas e consequente transformação em impulsos eléctricos), a membrana limitante externa (que separa as porções de segmento interiores dos fotorreceptores do seu núcleo), a camada nuclear externa, a camada externa plexiforme, a camada nuclear interior, a camada interior plexiforme, a camada de células ganglionares, a camada de fibras nervosas, e, finalmente, a membrana limitante interior (composta por células de Muller).^{[2], [10]}

É no neuroepitélio que se distinguem várias estruturas fulcrais na recepção e transmissão sensorial, tais como disco óptico, a fóvea e a mácula. O disco óptico é uma área circular central mais brilhante, de diâmetro $1500 \mu m$, de onde partem a maioria dos vasos da retina, e contém os axónios das células ganglionares. A fóvea encontra-se no eixo óptico do olho, com uma forma oval e a uma distância do disco de aproximadamente duas vezes o diâmetro deste. É a região onde se forma a imagem e onde se concentram os cones, num arranjo hexagonal, garantindo uma eficácia máxima e uma espessura mínima da retina. A mácula encontra-se dividida em 4 regiões concêntricas: no centro encontra-se a fóveola, seguida da fóvea, parafóvea e perifóvea. É muitas vezes caracterizada por uma coloração amarelada, causada por pigmentos como a xantofila e os carotenóides e tem a função de filtro, evitando que determinados comprimentos de onda cheguem aos cones. Existe ainda, na parede da retina, um ponto onde todos os axónios se juntam para que se possam projectar para o exterior da retina e até ao cérebro. Este ponto é conhecido como “ponto cego”, uma vez que não tem células receptoras.^{[2], [10]}

Os fotorreceptores, responsáveis primários da transmissão dos estímulos visuais ao córtex cerebral, dividem-se em cones, encarregados da visão fotópica, e bastonetes, responsáveis pela visão escotópica. Os cones destacam-se na visão diurna e na percepção da cor e dos detalhes finos, respondendo preferencialmente a mudanças bruscas, enquanto que os bastonetes, muito mais sensíveis e concentrados na periferia da retina, estão directamente relacionados com a percepção nocturna, sendo preferencialmente activados como resposta a pequenas mudanças.^{[2], [3], [16]}

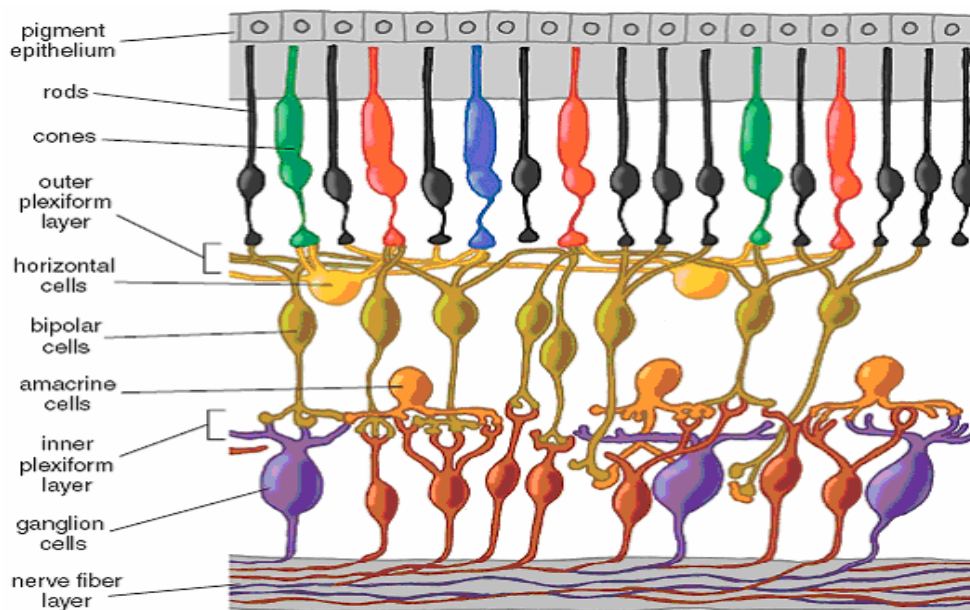


Figura 2.3. Disposição dos fotoreceptores. [3].

Os fótons chegam à retina e dão início, nos fotoreceptores, a uma reacção fotoquímica, alterando a permeabilidade das membranas e provocando uma reacção em cadeia de activação proteica, através da libertação de neurotransmissores nas sinapses das células. Estes fotoreceptores encontram-se ligados a células horizontais, que modificam o sinal antes da chegada às células bipolares e ganglionares. [3], [16]

As células bipolares, por sua vez, dividem-se também em dois tipos: ON e OFF, distinguidas pelo tipo de polarização a que são sujeitas e pelos receptores dos neurotransmissores. Adjacentes às células bipolares, encontram-se as células ganglionares, igualmente divididas em ON e OFF. As células ON são despolarizadas pela libertação de glutamato nas sinapses na presença de luz, enquanto que as células OFF são hiperpolarizadas no escuro. [3], [16]

Uma secção radial da retina evidencia que as células ganglionares se encontram numa zona mais profunda, próxima do cristalino, enquanto que os cones e bastonetes estão mais afastados do centro, contra o epitélio pigmentado e a coróide. [2]

Ou seja, o papel dos fotoreceptores centra-se na capacidade de recepção dos fótons e consequente transformação em sinais bioeléctricos, enviados aos neurónios. As vias de comunicação aqui descritas parecem extremamente simples; contudo, os metabolismos desencadeados e a activação proteica, bem como a sua condução e tipo de resposta, envolvem ainda mais conjuntos celulares e estruturas bem mais complexas. [3]

2.3. Formação da imagem no olho

A focagem de uma imagem a uma distância variável depende da curvatura do cristalino, controlada pelo músculo ciliar: para objectos distantes, o músculo relaxa para que o cristalino se torne menos esférico, ou pode contrair, para que a curvatura aumente, focando objectos próximos. A distância entre o centro do cristalino e a fóvea (distância focal) varia entre 14mm e 17mm. Para objectos a uma distância superior a 3m, o cristalino apresenta um poder refractivo mínimo e, quanto mais próximo o objecto, maior será esta potência. Assim, a formação da imagem é encarada como uma simples questão de óptica, onde a imagem que chega à fóvea é ampliada numa proporção dependente do raio de curvatura do cristalino, e invertida. ^[9]

2.4. Movimentos oculares

O globo ocular encontra-se rodeado de músculos, que regulam os movimentos de grande amplitude e permitem deslocamentos variáveis, de forma a perceber globalmente o espaço envolvente. No entanto, para que haja fixação de um ponto e focagem da imagem, ocorrem pequenos movimentos, designados movimentos sacádicos. Estes movimentos são deslocamentos súbitos do globo ocular, produzidos por uma mudança de impulsos. Estes movimentos são extremamente rápidos, com velocidades que podem atingir os 1000 graus por segundo, e separados entre si por períodos de fixação, durante os quais o olho permanece imóvel. Uma vez iniciados estes movimentos, não é possível pará-los nem modificá-los, e a sua amplitude depende do impulso neuronal. A sensibilidade visual é diminuída no período que antecede os movimentos sacádicos e enquanto estes decorrem – supressão sacádica. Isto é, nestes momentos, a informação visual é comprometida, o que significa que só é adquirida e enviada para o cérebro enquanto o olho permanece imóvel. Então, estes movimentos são suficientemente pequenos para que possa ocorrer uma focagem contínua da imagem no centro da fóvea. Ou seja, estes movimentos são considerados mecanismos compensatórios do afastamento da região central, como resultado de um movimento de larga escala. Em relação ao processamento das imagens no cérebro, entende-se como uma amostragem descontínua da posição dos olhos em relação ao alvo, em que os movimentos sacádicos são entendidos como um ajuste do erro de focagem. ^{[6], [14], [15]}

2.5. Doenças degenerativas da retina

A maioria das doenças que afecta a retina conduz a degenerações nas suas enervações. As retinopatias diabéticas são as que mais destacam, pela evolução da diabetes na população em geral, devido à alteração dos hábitos alimentares. Nestes casos, ocorrem microaneurismas, hemorragias pontuais, exudatos (depósitos amarelados na retina), dilatação dos vasos e neovascularizações progressivas na retina, afectando gravemente o fluxo sanguíneo para as regiões mais importantes, uma vez que o mesmo fluxo é repartido por mais vasos. A retinopatia diabética pode conduzir à cegueira, caso não seja reconhecida nos primeiros estádios de desenvolvimento. O seu tratamento passa pela fotocoagulação dos vasos, fazendo incidir um feixe laser no epitélio pigmentado da retina, onde é convertido em calor e cauteriza os vasos com malformações. [1], [7], [8], [10], [11], [12]

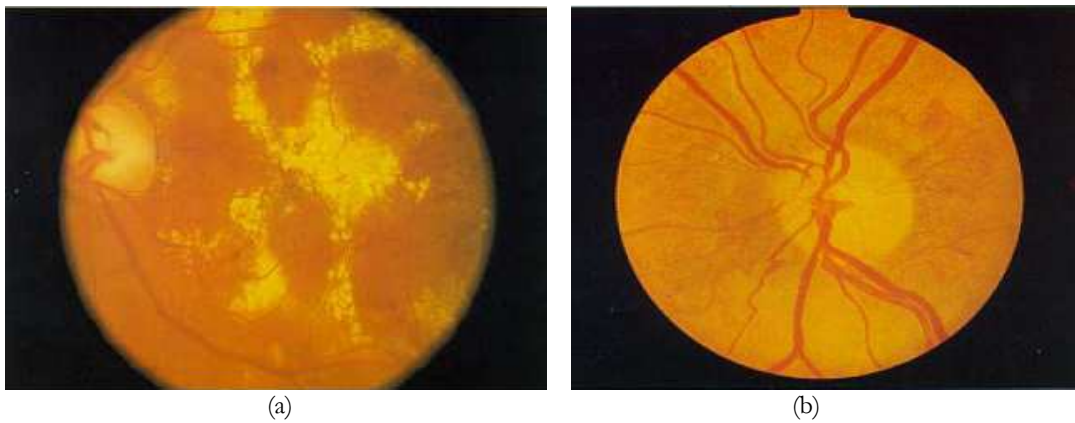


Figura 2.4. (a) Retinopatia diabética, com presença de exudatos. (b) Neovascularização provocada por retinopatia diabética [7].

Outra doença que afecta drasticamente a retina é a *retinis pigmentosa*, uma doença hereditária, para a qual ainda não há cura. Esta consiste num elevado número de mutações genéticas que afectam os bastonetes, levando à sua degeneração. É caracterizada por uma pigmentação negra na zona periférica da retina (reduzindo a visão nocturna) e um estreitamento dos vasos na saída do nervo óptico, comprometendo o fluxo sanguíneo. [7]

Para além das doenças que afectam a retina, há que considerar também as que danificam as restantes zonas do olho. É o caso das cataratas, caracterizadas pela opacidade do cristalino, em que ocorre um desvio ou mesmo bloqueio dos raios luminosos, o que leva a uma visão desfocada. Normalmente, é causada pelo envelhecimento dos tecidos, mas também pode ser provocada por traumatismos ou doenças metabólicas. O seu tratamento passa pela extracção

do cristalino e posterior inserção de uma lente intraocular artificial, de forma que os objectos possam ser novamente focados na retina. [7]

O glaucoma é outro dos grandes problemas do envelhecimento dos tecidos, em que a pressão na câmara vítrea aumenta, como consequência da incapacidade de troca de fluidos na câmara anterior do olho, danificando os vasos e os axónios das células ganglionares e provocando perda do campo de visão. O glaucoma pode também ser causado pelo bloqueio do ângulo de filtração por parte da íris na periferia da córnea. O seu tratamento pode ser feito com recurso a fármacos, que inibem a produção de fluidos ou promovem a sua expulsão, ou através de cirurgia laser. [1],[7]

A degeneração macular é outra das enfermidades que mais ocorre com o envelhecimento dos tecidos, podendo conduzir à cegueira. As áreas circundantes da mácula são prejudicadas pelo epitélio pigmentado, formando drusas (pontos brancos na retina) e permitindo a fuga de fluidos na parte posterior da fóvea, levando à cegueira. [1],[7]

É de notar que há ainda muitas doenças que afectam a visão, normalmente relacionadas directamente com o envelhecimento dos tecidos e consequentes alterações na elasticidade dos vasos e no poder refractivo do olho, ou que certas anomalias, como hipertensões e anemias, tumores, ou infecções, podem ter consequências na visão. [1],[7]

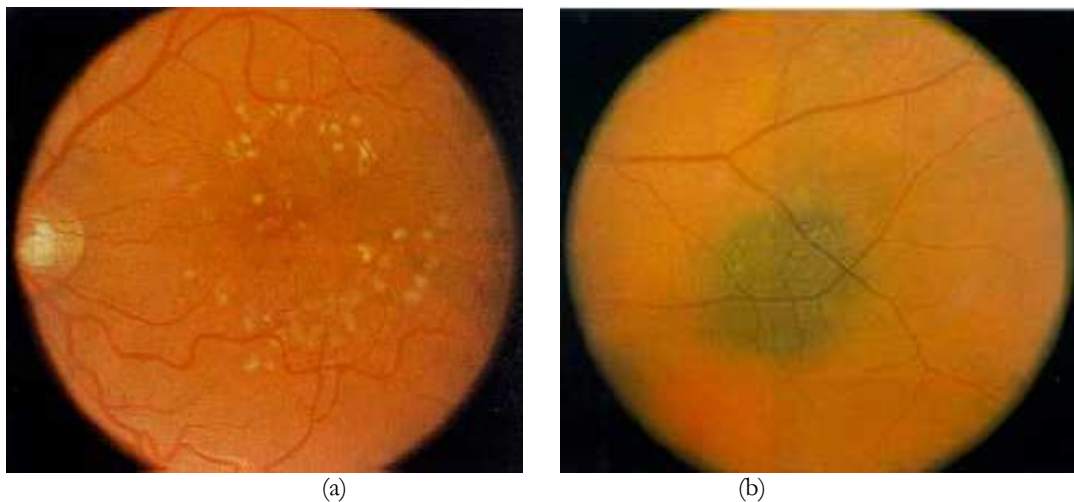


Figura 2.5. (a) Drusas presentes na mácula. b) Melanoma benigno na coróide – estacionário, pigmentado e plano. Adaptado de [7].

Referências Bibliográficas

- [1] Schlote, T. et al. Pocket Atlas of Ophthalmology. 1st ed. Thieme, 2006.
- [2] Kolb, Helga et al. "Simple Anatomy of the Retina." Webvision - The organization of the retina and Visual System. 2005. 16 Aug 2007 <http://retina.umh.es/webvision/sretina.html>.
- [3] Kolb, Helga. "How the Retina Works." American Scientist Volume 91 January–February 2003 28-35. 12 May 2007
<http://www.americanscientist.org/template/AssetDetail/assetid/16218?fulltext=true> .
- [4] Fernández, Félix J. A. et al, Oftalmología en Atención Primaria. 2nd ed. Alcalá La Real: Formación Alcalá, 2003.
- [5] Kolb, Helga et al. "Gross Anatomy of the Eye." Webvision - The organization of the retina and Visual System. 2005. 16 Aug 2007 <http://retina.umh.es/webvision/anatomy.html>
- [6] Starling, E . Princípios de Fisiologia Humana - segundo volume. 14th ed. Fundação Caloute Gulbenkian, 1968.
- [7] Ming, Arthur L. S. et al. Color Atlas of Ophthalmology. 3rd ed. World Science
- [8] Minassian, Darwin . "A National Research Strategy for Ophthalmology." Medical Retina - 5 year Research Plan March 2002 16-22. 27 July 2007
<http://www.rcophth.ac.uk/docs/publications/EyeResearchFinal.pdf> .
- [9] Gonzalez, Rafael et al. Digital Image Processing Using Matlab. New Jersey: Prentice Hall, 2004.
- [10] Lang, Gerhard K. . Ophthalmology - a Short Textbook. New York: Thieme, 2000.
- [11] Management of Diabetic Retinopathy. 3rd ed. National Health and Medical Research Council, 1997.
- [12] Ai, Everett. "Current management of diabetic retinopathy." The Western Journal of Medicine 157 July 1992 67 - 70. 24 May 2007
<http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=1021914> .
- [13] Holz, Frank G. et al. Medical Retina. Springer,
- [14] Bridgeman, Bruce . "A Theory of visual stability across saccadic eye movements" Behavioral and Brain Sciences 17:1994 247-292. 24 May 2007
<http://www.bbsonline.org/Preprints/OldArchive/bbs.bridgeman.html>
- [15] MacKay, William. Neurofisiologia sem Lágrimas. Lisboa: Fundação Caloute Gulbenkian, 1999.

3. Representação de imagens

Uma imagem é definida como uma função bidimensional, caracterizada pela quantidade de iluminação que incide na cena a adquirir e pela quantidade de energia reflectida, de tal forma que

$$f(x, y) = i(x, y)r(x, y) \quad (3.1)$$

A função $f(x, y)$ representa a intensidade associada a uma imagem, $i(x, y)$ a iluminação (quantidade de luz que chega à cena) e $r(x, y)$ a reflectância, nas coordenadas (x, y) . A cada coordenada da imagem é atribuído um valor não negativo dependente de r e i , que define a intensidade nessa coordenada. A imagem digital é definida como uma imagem discretizada por amostragem e por quantização, isto é, são designados valores discretos tanto para coordenadas espaciais como nos valores associados a f . Assim, a imagem passa a estar definida num espaço bidimensional discreto, considerando, em cada dimensão, amostras igualmente espaçadas, que definem o conjunto de valores possíveis para a imagem, concretizando a noção de escala. O resultado será uma matriz $M \times N$ de valores discretos, em que cada elemento é definido como um píxel, como exemplifica a figura. ^[1]

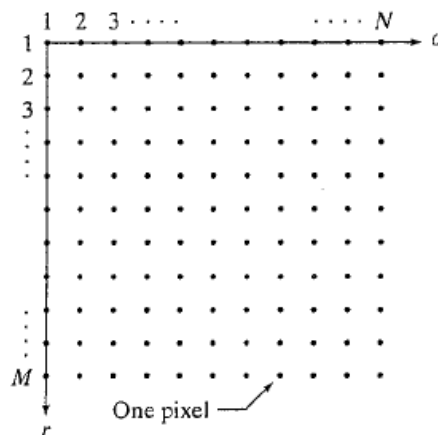


Figura 3.1. Representação da imagem como um espaço matricial bidimensional de dimensões $M \times N$. [1].

A resolução espacial de uma imagem está relacionada com a possibilidade de distinção dos detalhes de uma imagem. ^[1] Ou seja, a resolução espacial é definida pelas dimensões da

imagem, expressas em píxeis. Se tiver uma imagem 20×30 cm e píxeis de dimensão 1×1 mm, a resolução espacial será 200×300 píxeis.

As diferentes intensidades de cores que constituem uma imagem permitem a sua integração em diferentes espaços. Estes encontram-se relacionados entre si e as particularidades de cada um tornam as características da imagem mais intuitivas para o ser humano.

3.1. Espaços de cores

Os espaços de cor podem ser distinguidos de acordo com o diagrama seguinte.

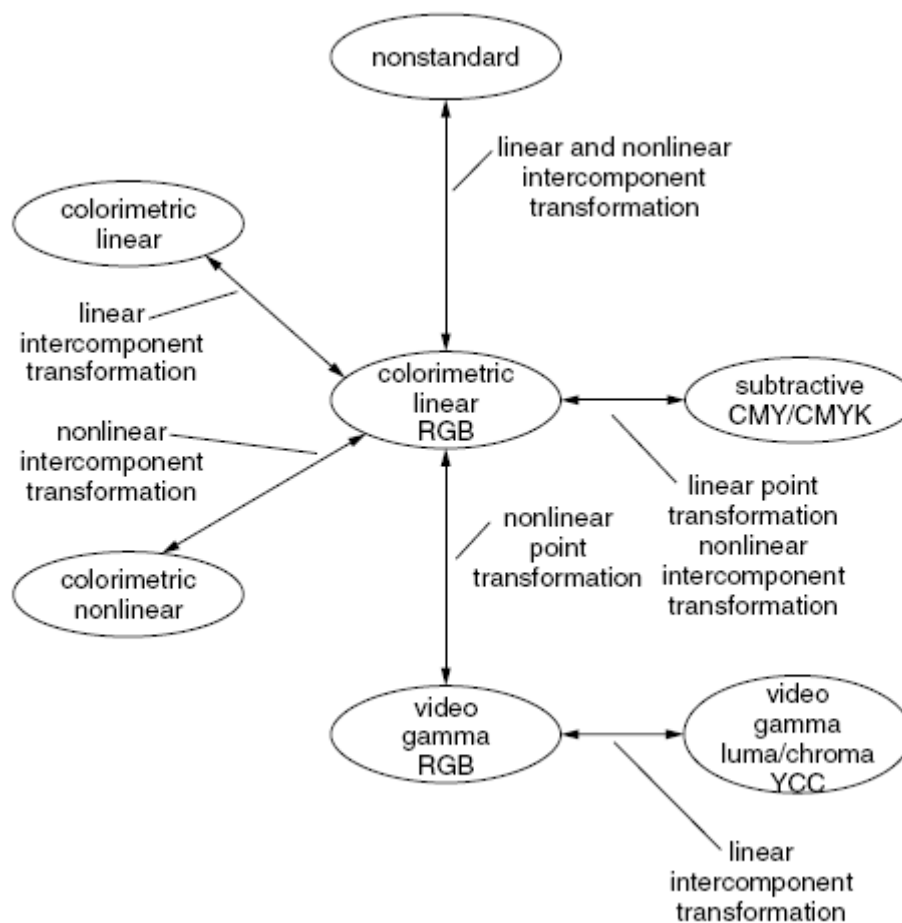


Figura 3.2. Relação entre os espaços de cores. [4].

Dos vários espaços de cor, os mais importantes de referir são os RGB e HSV.

As imagens de cores naturais são obtidas, normalmente, a partir de câmaras com três sensores de cores, cada um correspondente a uma cor primária: verde, vermelho e azul – RGB. ^[4] Para este sistema, é considerado um *byte* para cada valor de cor primária. Os sinais gerados em cada sensor de cor são proporcionais à quantidade de energia de cada comprimento de onda que detectam, e é esta que estabelece a intensidade da imagem. ^[4] Ou seja, a intensidade da imagem é proporcional à energia de cada sensor. Se todas as cores primárias estiverem presentes no seu máximo de intensidade, a resultante será a cor branca, da mesma forma que, de iguais proporções, resultarão tons de cinza, dependendo do valor de intensidade. Para níveis nulos de intensidade, obter-se-á preto.

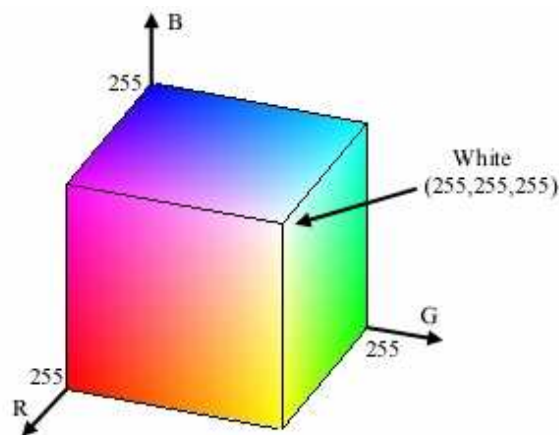


Figura 3.3. Cubo esquemático do sistema de cores RGB. [6].

O sistema RGB é um sistema aditivo, uma vez que todas as cores são criadas a partir da adição ao preto. Partindo de níveis nulos de intensidade, vão sendo adicionadas as componentes correspondentes, definindo assim as cores associadas a cada nível. ^{[3], [4]}

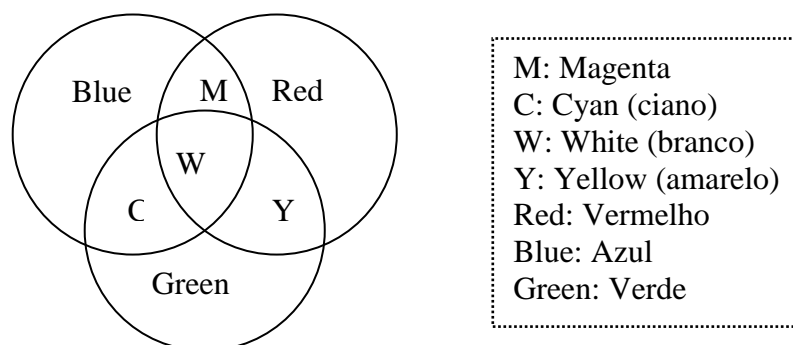


Figura 3.4. Esquema de cores aditivas. Adaptado de [8].

Uma forma de normalização de um sistema RGB consiste no conjunto de equações ^[2]

$$\begin{aligned} \text{Intensidade } I &= (R + G + B) / 3 \\ \text{vermelho } R &= R / (R + G + B) \\ \text{verde } G &= G / (R + G + B) \\ \text{azul } B &= B / (R + G + B) \end{aligned} \quad (3.2)$$

Esta manipulação permite uma facilidade de transformação para outros sistemas de cores. Além disso, no caso de diferentes objectos se encontrarem a distâncias diferentes das fontes de iluminação, a normalização permite uma interpretação de valores mais fidedigna da realidade.

A maior desvantagem do sistema RGB reside no facto de as suas componentes apresentarem elevados valores de correlação entre si, o que o torna um sistema inviável na compressão de imagem. ^[3]

Como alternativa ao sistema RGB, surgem os sistemas XYZ e CMY. No primeiro, as componentes são obtidas por simples avaliação de pesos de cada uma das cores primárias. ^[3]

$$\begin{aligned} X &= 0.490R + 0.310G + 0.200B \\ Y &= 0.177R + 0.812G + 0.011B \\ Z &= 0.000R + 0.010G + 0.990B \end{aligned} \quad (3.3)$$

No sistema CMY, as cores são obtidas por subtracção ao branco – sistema substractivo. ^[4] É composto por tons de magenta, ciano e amarelo, correspondentes às cores complementares de absorção das cores do sistema RGB. Ou seja, a magenta absorve o verde, o amarelo absorve o azul e o ciano absorve o vermelho. ^[2] Num cubo de cores, o ponto correspondente ao preto passa a corresponder ao branco e vice-versa.

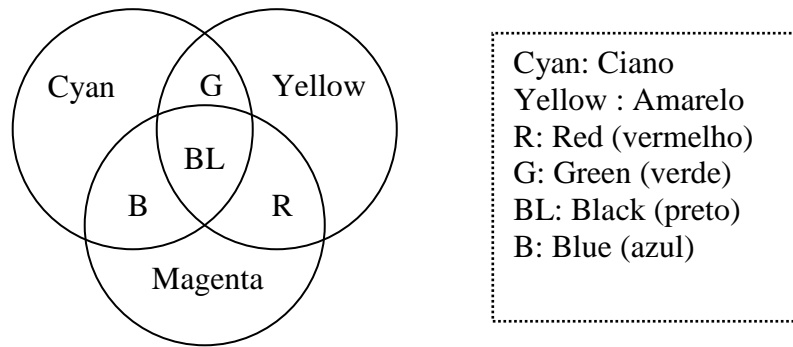


Figura 3.6. Esquema de cores subtrativas. Adaptado de [8].

Abordando uma outra linha de pensamento, em que é analisado o comportamento humano perante algumas ilusões de óptica, surgem os modelos YIQ E YUV. Estes definem-se como canais de percepção opostos ^[3], que distinguem luminância, definida como uma componente do sistema, e duas componentes de cromaticidade, baseadas nos canais de diferença de cor de R-Y (vermelho – amarelo) e B-Y (azul – amarelo). ^[3] Estes sistemas encontram-se mais relacionados com sinais vídeo e televisão e permitem uma transmissão em larguras de banda mais limitadas. ^[3] Os valores são obtidos a partir do sistema RGB por uma alteração do factor de peso de cada componente. Por exemplo, no sistema YUV, as equações representativas são ^[2]

$$\begin{aligned}
 Y &= 0.30R + 0.59G + 0.11B \\
 U &= 0.493 * (B - Y) \\
 V &= 0.877 * (R - Y)
 \end{aligned}
 \tag{3.4}$$

Este tipo de modelos baseado na oposição de cores é particularmente útil na detecção de arestas de uma imagem e de contrastes de cores. ^[3]

Numa outra perspectiva do espaço de cores, a informação pode ser codificada por separação de intensidade e critérios de cromaticidade – saturação e contraste. Este modelo é conhecido como HSI, ou HSV, caso a intensidade seja referida pelo termo valor. ^[2] Este sistema é graficamente obtido pela projecção do cubo do sistema RGB ao longo da diagonal principal. Num sistema 3D, é visível no eixo principal os valores de intensidade e os valores de H variam no ângulo de 0 a 2π . A descrição do algoritmo de derivação do sistema HSI a partir

do RGB é explicada em [2]. Este modelo, baseado em contraste, torna-se mais intuitivo para a percepção humana, comparativamente ao sistema RGB.

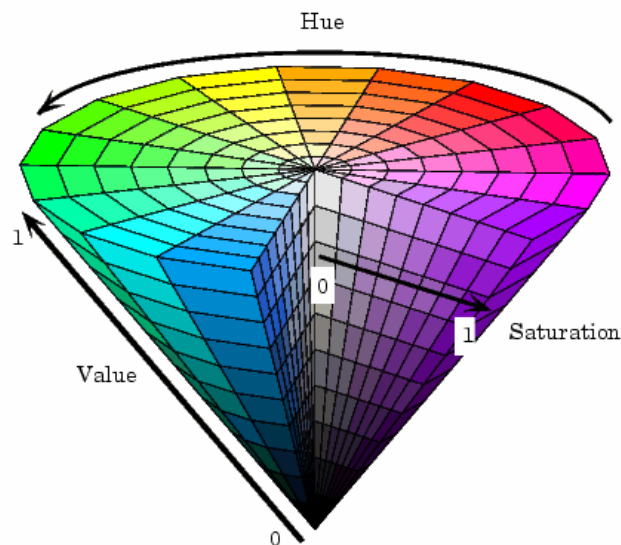


Figura 3.7. Esquema de cores HSV. [6].

A representação informática de uma imagem implica uma redefinição das imagens, para que estas possam ser submetidas a algumas manipulações. Uma das linguagens de programação mais desenvolvidas no tratamento de uma imagem é o Matlab.

3.2. A imagem em Matlab

No ambiente da linguagem de programação Matlab, as imagens são tratadas como espaços matriciais, bidimensionais, de valores discretos. Na referência a cada elemento, as localizações verticais dos píxeis são referidas pelo primeiro índice e as localizações horizontais são referidas pelos valores em segundo índice. Assim, um píxel indexado como (3,2) encontra-se na terceira linha e segunda coluna. ^[7]

As imagens podem ser identificadas conforme a sua organização matricial e o espaço de memória dispendido para o seu armazenamento. Assim, as imagens podem ser binárias, de intensidades, RGB ou indexadas, e armazenadas em formato *uint8*, *uint16*, *double* ou *logical*. ^{[1], [6]}

Uma imagem binária é representada por um array de valores lógicos de 0 ou 1, que determinam o branco ou preto, respectivamente. Pelas suas características dos dados, só podem ser armazenadas em formato *logical*.^[7]

Numa imagem de intensidades, é atribuído um valor de intensidade a cada píxel, normalmente numa escala de cinzentos, e a imagem apresentada é monocromática. Uma imagem em escala de cinzentos pode ser armazenada em formato *double*, *uint8* ou *uint16*. No primeiro, os arrays da imagem são tratados como matrizes de variáveis normais, compostas por valores numéricos e, portanto, sujeitas às funções definidas na linguagem de programação. No caso do armazenamento em formato *uint8*, é dispendido um espaço de memória de 1 *byte* a cada elemento e os valores variam numa gama de intensidade de 0 a 255. No entanto, podem também ser dispendidos 2 *bytes* para armazenamento na memória, e a gama passa a compreender valores entre 0 e 65535 – *uint16*.^{[6], [7]}

As imagens em RGB são definidas pela relação entre os valores dos *arrays* e os valores de cor atribuídos aos píxeis. Cada píxel é especificado por 3 valores, cada um para as diferentes componentes de cor – verde, azul e vermelho. Estas imagens são representadas por *arrays* $m \times n \times 3$, de classe *uint8*, *uint16* ou *double*.

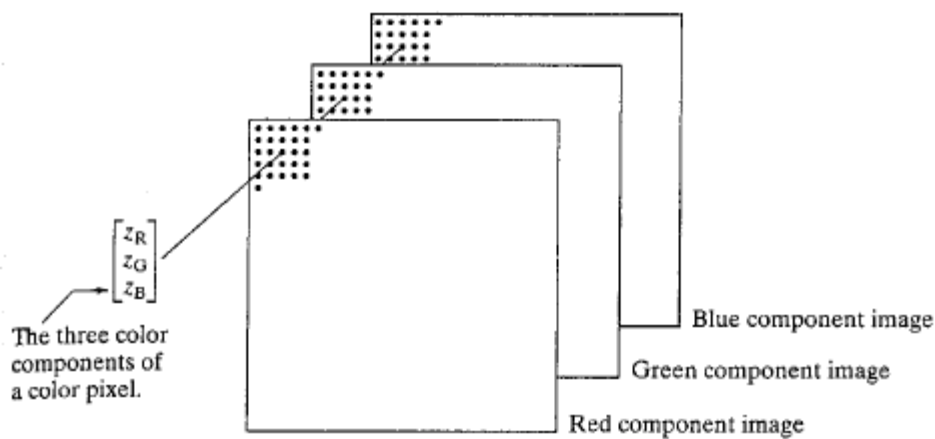


Figura 3.8. Esquematização de uma imagem do tipo RGB. [1]

A partir deste tipo de imagem pode ser construído um mapa de cores, definido como um conjunto de valores ordenados, de tamanho $n \times 3$, em que n corresponde ao número de cores definidas e o valor 3 refere-se a cada uma das componentes R, G e B. É através deste mapa que são identificados os índices para a construção de uma imagem indexada. Este tipo

de imagem tem dois componentes: uma matriz de dados e um mapa de cor. A cor de cada píxel da imagem indexada é determinada pelo valor correspondente da matriz de dados, de tamanho $m \times 3$, de classe *double*.^[1]

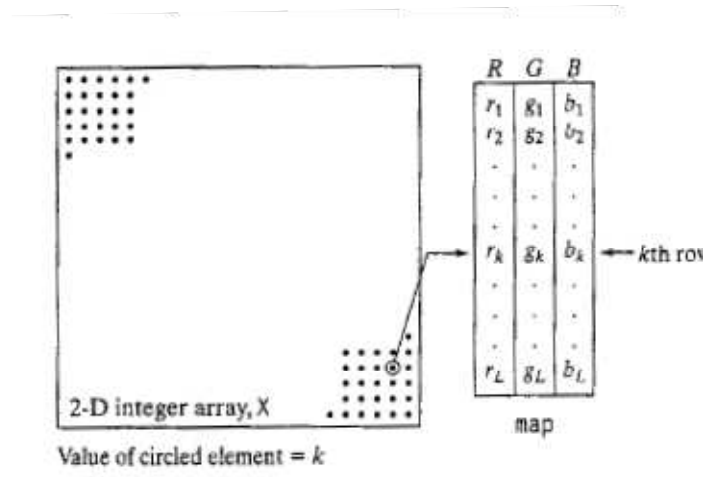


Figura 3.9. Representação dos elementos de uma imagem indexada. [1]

A transformação a partir de uma imagem RGB implica uma redução do número de cores no processo. Para isso, há uma aproximação de cores na imagem original, através de quantização – redução do número de cores –, ou mapeamento de cores. No caso de uma escala RGB, implica uma divisão do cubo representativo do espaço RGB em partes mais pequenas, atribuindo um valor de cor dentro de cada cubo mais pequeno igual ao seu valor central. Especificando um mapeamento de cores, este será usado como alternativa à quantização para especificar a melhor correspondência entre cores nos padrões da imagem RGB.^[1]

O Matlab tem funções já definidas para a conversão entre espaços de cores e tipos de imagens.^{[1], [6], [7]}

A integração de uma imagem num espaço de cor e a sua análise computacional permite a diferenciação de padrões e o reconhecimento das características mais importantes. Para isso, são identificadas as diferentes regiões que a definem.

3.3. Regiões de uma imagem

A aquisição e representação da imagem permitem obter informação do espaço envolvente e é através das suas características que se tem a percepção dos elementos. Uma não homogeneidade na imagem permite a distinção de várias características, que se podem destacar na textura, formação de arestas ou na formação de regiões suaves. A detecção de uma pequena quantidade de parâmetros deste género facilita a caracterização dos elementos e da própria imagem, que, para facilidade de diferenciação, são apresentadas, normalmente, numa escala de cinzentos. Assim, as regiões são definidas pela sua área e pela semelhança dos níveis de cinzento (estes permanecem constantes no domínio da região), e são atribuídas a objectos ou superfícies da imagem. As texturas da imagem, por sua vez, permitem distinguir certos padrões da imagem. Numa primeira impressão, a diversidade de valores que as compõem pode ser confundida com ruído. A textura pode ser modelada através do cálculo de correlações entre píxeis em diferentes direcções e escalas, estabelecendo similaridades. ^[5] Um exemplo destas aproximações a modelos é o caso dos campos aleatórios de Markov, que são descritos em [9]. O terceiro parâmetro, as arestas são, muitas vezes, consideradas as características mais importantes de uma imagem; são os limites de regiões ou a zona de transição entre uma região e um plano de fundo, e caracterizam-se pela variação, mais ou menos abrupta, do nível de cinzento. Pela sua detecção, é possível inferir acerca dos contornos dos objectos de uma imagem, e, conseqüentemente, dar uma ideia geral dos seus elementos e da sua localização. ^[5] Matematicamente e computacionalmente, é, provavelmente, a característica que mais interesse levanta, pela variedade de técnicas empregues na sua detecção.

3.4. Identificação de arestas

Tecnicamente, a detecção de arestas consiste na localização e distinção dos píxeis nos contornos dos objectos pertencentes à imagem, por um aumento do contraste entre estes píxeis e o plano de fundo, tornando as arestas mais visíveis. Na prática, estes conceitos são complementares, uma vez que a detecção privilegiará valores dos píxeis acima de um nível específico de intensidade, de forma a destacá-los. ^{[10], [12]}

No entanto, a tarefa de detecção pode ser dificultada por algumas das técnicas de processamento utilizadas. Estas podem conduzir, muitas das vezes, a artefactos, tais como paralelismo de arestas, que são entendidas como sendo de maiores dimensões (em vez da sua

classificação distinta, são atribuídas maiores dimensões a uma única aresta), ou não são detectadas ^[13]. Além disso, devido a factores como a intensidade da luz, lentes, movimento, e outros, é pouco provável que dois píxeis com o mesmo nível de cinzento na cena sejam representados com as mesmas propriedades na imagem ^[14]. Destes erros resultam uma identificação incorrecta dos pontos relevantes, podendo conduzir a falsos positivos – detecção de píxeis que não são verdadeiramente arestas – ou a falsos negativos – falha na detecção de arestas. ^[12]

Para que a detecção possa ser considerada robusta (e, conseqüentemente, sujeita a menos erros), consideram-se três estádios: filtragem, diferenciação e detecção. A filtragem da imagem vai permitir a eliminação de ruído, facilitando o processo de diferenciação. Neste, são calculadas as derivadas da imagem, que servirão de base à localização das arestas, no processo de detecção. ^[10]

3.4.1. Filtragem

Neste estágio, a imagem é filtrada com uma máscara ou um conjunto de máscaras, para redução do ruído (originado por amostragem, quantização ou focagem da câmara) ou para correcção de irregularidades na superfície dos objectos. ^[10]

Considerando g como o filtro, este é convolucionado com a imagem f , resultando numa imagem h . Esta operação dá-se no domínio temporal e pode ser vista como a passagem de uma máscara g na imagem, da esquerda para a direita, centrada em cada píxel de f , e computadorizando o produto píxel a píxel.

Matematicamente, a convolução é definida por:

$$h(x, y) = f(x, y) * g(x, y) \quad (3.5)$$

O filtro mais simples que pode ser usado é o filtro de média. O nível de cinzento em cada píxel é substituído pelo valor da média dos níveis numa vizinhança desse píxel, proporcionando um mecanismo de eliminação de ruído aleatório. No caso de um filtro 3×3 aplicado a uma imagem, ter-se-ão 9 elementos e o factor de peso será $\frac{1}{9}$ em cada um. A

máscara é centrada em cada píxel (i, j) da imagem f e o produto dos píxeis é calculado. Para uma imagem f definida em $M \times N$ píxeis, ^[10]

$$b(x, y) = \sum_{i=-1}^{i=1} \sum_{j=-1}^{j=1} f(x+i, y+j)g(i, j) \quad (3.6)$$

em que

$$M \geq x+i \geq 1$$

$$N \geq y+i \geq 1$$

Outro filtro muitas vezes usado é o Gaussiano, em que a importância de cada píxel é inversamente proporcional à distância do píxel central. É expresso como:

$$g(x, y) = e^{-\frac{(x^2+y^2)}{2\sigma^2}} \quad (3.7)$$

em que σ é o desvio padrão ^[10].

3.4.2. Diferenciação

Nesta etapa, o processamento centra-se nas localizações onde se registam alterações significativas de intensidade. A diferenciação é feita por cálculo das derivadas de uma função representativa da imagem ^[10].

A derivada de uma função contínua f é dada pela expressão

$$f' = \frac{df}{dx} = \lim_{\Delta x \rightarrow 0} \frac{f(x) - f(x - \Delta x)}{\Delta x} \quad (3.8)$$

Porém, ao tratar-se de imagens, é assumido um domínio discreto de valores, e o mínimo atribuído a Δx é 1. Considera-se, então, uma aproximação discreta de derivadas, denominada diferenças finitas, expressa como ^[10]

$$f' = \frac{df}{dx} \cong f(x) - f(x-1) \quad (3.9)$$

No entanto, para simplificação de linguagem, usar-se-á o termo “derivada” ao referir-se às diferenças finitas.

Na definição de diferenças finitas, é assumido que a primeira derivada deve ser nula nos níveis constantes de intensidade e não nula ao longo de variações de intensidade. ^[1]

O processo de diferenciação de imagens tem em conta a sua bidimensionalidade, o que significa que as derivadas são calculadas em ambas as direcções, por aplicação do operador gradiente. Este é identificado por $\nabla f(x, y)$ e composto por (f_x, f_y) , em que f_x e f_y são as derivadas nas direcções x e y , respectivamente. ^[10]

$$\begin{aligned} \nabla_x f(x, y) &= f(x, y) - f(x-1, y) \\ \nabla_y f(x, y) &= f(x, y) - f(x, y-1) \end{aligned} \quad (3.10)$$

A direcção θ e magnitude M do operador gradiente são definidas como

$$\begin{aligned} \theta &= \arctan \frac{f_y}{f_x} \\ M(x, y) &= \sqrt{f_x^2(x, y) + f_y^2(x, y)} \end{aligned} \quad (3.11)$$

Qualquer píxel cuja magnitude ultrapasse um valor de limiar estabelecido é interpretado como um *edge* da imagem.

A derivada direcciona pode ser então expressa em função da direcção do gradiente, através da expressão ^[10]

$$f'_\theta = \frac{\partial f}{\partial x} \cos \theta + \frac{\partial f}{\partial y} \sin \theta \quad (3.12)$$

As derivadas de primeira ordem são utilizadas para realce de contornos, enquanto que as derivadas de segunda ordem são, normalmente, ideais para um realce de detalhes finos, incluindo ruído e pontos isolados. ^[1]

3.4.3. Detecção

No processo de detecção, os pontos de alterações significativas da imagem são localizados no espaço. Para isso, aplicam-se maioritariamente dois métodos: supressão dos valores não-máximos e magnitude do gradiente normalizada. ^[10]

Supressão dos valores não-máximos

A direcção do gradiente é sempre perpendicular ao *edge*, e as alterações dos níveis de cinzento registam-se, na sua maioria, nessa direcção. De acordo com estas constatações, se o valor do nível de cinzento de um determinado píxel não muda significativamente na direcção do gradiente, este não é considerado um *edge* e é, então, suprimido. Essa supressão é definida algebricamente como

$$M(x, y) = \begin{cases} M(x, y) & \text{se } M(x, y) > M(x', y') \\ M(x, y) & \text{se } M(x, y) > M(x'', y'') \\ 0 & \text{outros} \end{cases} \quad (3.13)$$

em que $M(x', y')$ e $M(x'', y'')$ são as magnitudes do gradiente em ambos os lados do *edge* na direcção do gradiente. Todos os píxeis que não forem eventuais arestas terão valor 0 e são eliminados. Após este processo, o gradiente poderá ser normalizado e testado para valores de limiar. É necessário ter em conta que o número de direcções a ter em consideração precisa de ser estipulado, também ^[10].

Magnitude do gradiente normalizada

Pela expressão da magnitude do gradiente expressa anteriormente, assume-se que a normalização deste valor será dada por

$$N(x, y) = \frac{M(x, y)}{\max_{i=1\dots n, j=1\dots n} M(i, j)} \times 100 \quad (3.14)$$

com escala compreendida entre 0 e 100. Note-se que o valor mínimo assumido para M é zero.

Na construção de um mapa de arestas, será assumido o valor de 1 caso de $N(x, y)$ ser superior ao estipulado por T ^[10].

Toda a imagem se encontra vulnerável a erros, quer sejam de aquisição (a nível do sistema óptico, por exemplo), quer sejam de tratamento. Para que os resultados da identificação de arestas sejam credíveis, é importante minimizar estes erros, o que pode ser feito de variadas formas, dependentes do tempo computacional. É, por isso, necessário estabelecer à partida, um compromisso entre tempo e processamento. Quanto melhor o processamento, melhor a eliminação de erros, mas maior será o esforço e tempo computacional exigido. É atendendo aos objectivos do processamento que este compromisso pode ser honrado.

Referências Bibliográficas

- [1] Gonzalez, Rafael et al. Digital Image Processing Using Matlab. New Jersey: Prentice Hall, 2004.
- [2] Shapiro, Linda. et al. Computer Vision. 1st ed. Prentice Hall, 2000.
- [3] Sangwine, S.J. et al. The Colour Image Processing Handbook. 1st Ed. Chapman & Hall, 1998.
- [4] Pratt, William. Digital Image Processing. 3rd ed. Wiley-Interscience, 2001.
- [5] Perry, Stuart William. et al. Adaptative Image Processing - A Computational Intelligence Perspective. CRC Press LLC, 2002.
- [6] Matlab v.7.0.0.19920(R14), Mathworks, 6 May 2004, www.mathworks.com
- [7] Semmlow, John L., Biosignal and Biomedical Image Processing - Matlab-Based Applications. 1st Ed. New York: MARCEL DEKKER, 2004
- [8] Carvalho, Ivan. "Tratamento de Imagens". 2004. 4 October 2006
- [9] Kindermann, Ross. et al. Markov Random Fields and their applications, Providence, American Mathematical society, 2000
- [10] Mubarak, Shah. et al. "Edge Detection." Fundamentals of Computer Vision. 1997. 6 April 2007 <http://www.cs.ucf.edu/courses/cap6411/book.pdf>.
- [11] Forsyth, David A. ,"Edge Detection". Computer Vision: A Modern Approach. 1st Ed. Prentice Hall, 2003.
- [12] Parker, J.. "Advanced Edge Detection Tecniques: The Canny and the Shen-Castan Methods." Techniques in Computational Vision. 7 April 2007 <http://pages.cpsc.ucalgary.ca/~parker/501/edgedetect.pdf>
- [13] Ritter, Gerhard X. et al. Handbook of Computer Vision Algorithms in Image Algebra. 1st Ed. CRC Press, 2001
- [14] Tuytelaars, T. et al "A Survey on Local Invariant Features." 5 May 2006 1-64. 28 February 2007 http://homes.esat.kuleuven.be/~tuytelaar/survey_inv_features.pdf.

4. Retinal Tracking: State of Art

A detecção do movimento numa sequência de imagens pode ser levada a cabo através da localização dos vasos na imagem ou de outros aspectos característicos (como a mácula ou disco óptico, caso estes se encontrem representados na imagem). A imagem pode, inclusive, ser avaliada em diferentes escalas, realçando diferentes características conforme a escala das imagens, e, a partir daqui, aplicar a técnica que melhor se adequa ao objectivo pretendido. Um dos métodos mais simples consiste na análise da imagem píxel a píxel, detectando arestas ou pontos de variação de luminosidade, com base na análise descrita no capítulo anterior e por meio dos operadores apresentados no anexo A. Esta técnica vai permitir, essencialmente, localizar as fronteiras dos vasos ou os seus pontos característicos.^{[13], [16], [21]} Os operadores podem, e devem, ser aplicados em várias direcções, de forma a detectar os contornos conforma a sua orientação. Os contornos dos vasos podem, também, ser extraídos desenvolvendo algoritmos de contornos activos – *snakes*. Uma outra metodologia de segmentação avalia as diferentes texturas ou regiões presentes na imagem, diferenciando-as com base em critérios de classificação e similaridade. Esta classificação de regiões é, normalmente, desenvolvida a partir da aplicação de esquemas matemáticos morfológicos ou por formação de *clusters* (agrupamento de píxeis conforme a sua vizinhança e similaridade).

Em ambas as abordagens, podem ser estabelecidas comparações com modelos já desenvolvidos e ser feita uma análise estatística do seu comportamento.^[22] A curvatura ou forma das regiões pode ser inferida analisando o gradiente ou os diferenciais de segunda ordem da imagem, para cálculo dos zeros da função.

Após a determinação dos pontos ou regiões de maior interesse numa sequência de imagens, o movimento pode ser avaliado por comparação entre as suas posições relativas na imagem, por estabelecimento de correspondências entre janelas de comparação – *templates* – ou por análise do fluxo óptico da imagem. A posição relativa de uma característica, comparativamente à posição anterior, pode ser inferida recorrendo a equações de predição (isto é, inferindo acerca da localização a partir da anterior), baseando-se, para isso, em factores probabilísticos bayesianos^[28] e em factores de correcção de erros. Um excelente exemplo do método de predição e correcção é o filtro de Kalman.^{[29], [30]}

É de notar que as várias metodologias usadas para detecção de pontos de interesse de uma imagem dependem do tipo e da qualidade de imagem a que são aplicados (que são critérios imperativos na decisão de aplicação de pré-processamento) e na sua autonomia (se são eficazes só por si, sem haver uma necessidade de recorrer a métodos de pós-processamento). Na sua maioria, os métodos são complementares entre si e aplicados como um conjunto.

Alguns dos artigos consultados neste projecto apresentam um pseudo-código dos algoritmos desenvolvidos pelos autores de cada um, para uma melhor compreensão das implementações. É o caso dos artigos referenciados como [2] e [4].

Pretende este capítulo apresentar, de forma breve, as abordagens mais significativas que poderiam servir como auxílio aos objectivos do projecto desenvolvido. Note-se, contudo, que algumas delas não foram implementadas na prática.

4.1. Detecção dos pontos característicos da imagem

4.1.1. Aproximações multi-escalares

As aproximações multi-escalares aproveitam a imagem nas suas várias resoluções, permitindo um aumento da velocidade de processamento. As características de mais fácil distinção, como os vasos de maiores dimensões, são processadas a uma baixa resolução, enquanto que as características de maior detalhe são detectadas a uma resolução superior, o que conduz a uma maximização de respostas num conjunto discreto de orientação das máscaras aplicadas ^{[1], [6], [7], [18], [37]}. Estas máscaras podem ser construídas a partir de diferentes operadores, como o de Sobel ou de Canny (apresentadas em anexo), modificando-os conforme o tamanho e estrutura desejados. Estas características fazem desta metodologia uma aproximação muito robusta, do ponto de vista da eficácia da detecção. ^[21]

Tendo como base a magnitude do gradiente, calculada para cada uma das escalas, a resposta à orientação será o resultado da soma das magnitudes ^{[6], [9]}. Esta análise servirá de fonte para a detecção seguimento de vasos ou avaliação de pontos característicos, para um possível *tracking* entre *frames*. ^{[35], [36]} O cálculo adicional das derivadas de segunda ordem e a análise dos extremos destas respostas permitem a extracção dos pontos de maior interesse da imagem.

4.1.2. Aproximações de matched filters

A aplicação desta técnica envolve a convolução da imagem com múltiplos filtros, em diferentes direcções, para extracção de arestas (que, normalmente, correspondem aos contornos dos vasos). Estes filtros descrevem o aspecto esperado de um determinado sinal e são desenvolvidos conforme a orientação dos vasos e tamanho. ^[38] A máscara na direcção tangente é uma gaussiana, que suaviza as respostas ao longo do pico, e na direcção normal corresponde à derivada de segunda ordem da gaussiana. ^[19] A explicação da localização e da resposta a estes filtros encontra-se bem fundamentada em [6] e [10].

Após a aplicação destes filtros podem ser estabelecidos valores de limiar, complementados com histerese (estabelecimento de limites mínimos e máximos), análise de conectividade e *thinning*, para detecção de arestas da imagem. ^{[10], [20]}

4.1.3. Detecção da linha de centro (aproximações a esqueletos)

As aplicações desta teoria servem de mote ao mapeamento de objectos e análise de estruturas de uma imagem, daí que um dos primeiros objectivos seja criar um mapa de arestas (pela aplicação dos detectores descritos em anexo) que facilite a detecção das características da imagem ^[13]. Em alternativa à criação deste mapa, podem ser extraídos pontos iniciais da imagem – *seed points* –, escolhidos pelo utilizador, localizados por coordenadas específicas ou por uma grelha de píxeis sobreposta à imagem, conforme é explicado em [14].

Após a identificação das arestas, as linhas centrais dos vasos são determinantes na reconstrução do esqueleto da rede vascular, conforme descrita em [8] e [13]. No primeiro, a largura do vaso é calculada a partir das arestas da imagem. No segundo, são criados os mapas de arestas e de orientação destes e são analisadas as fronteiras dos vasos, com base em pressupostos relativos à largura dos vasos e vizinhança de píxeis. A partir daqui, é analisado o paralelismo entre vasos. O resultado final destas aplicações é a reconstrução da rede vascular através da linha central do vaso.

4.1.4. Aproximações baseadas em ridges

Um *ridge* é definido como o local no qual a intensidade assume um máximo local na direcção da curvatura principal. As suas aplicações fundamentam-se numa imagem em escala de

cinzentos, interpretando-a como um mapa de relevos a três dimensões (as duas primeiras relativas às coordenadas dos píxeis e a terceira relativa à intensidade dos píxeis). O processo pode ser iniciado com um *seed point* de cada vaso da rede vascular, normalmente fornecido pelo operador, ou por delimitação de regiões de interesse ^{[14], [39]}, para extracção do eixo médio dos vasos e conseqüente criação do mapa de relevos. O eixo médio dos vasos é procurado nas direcções dadas pela computadorização de segundas derivadas, guardadas numa matriz – matriz Hessiana ^[6],

$$H = \nabla \nabla^T L(x, \sigma) \quad (4.1)$$

em que L representa a convolução de uma função gaussiana com a imagem original e

$$\nabla = \left(\frac{\partial}{\partial x_1}, \frac{\partial}{\partial x_2} \right)^T \quad (4.2)$$

São, então, procurados os pontos numa imagem em que a primeira derivada da intensidade na direcção da maior curvatura tem sinal. Esta direcção equivale ao vector próprio da matriz da matriz Hessiana. O sinal do maior valor próprio desta matriz, λ , determina se é um mínimo ($\lambda > 0$) ou um máximo local ($\lambda < 0$) ^[12].

A curvatura pode também ser analisada pelas expressões

$$\kappa = \frac{-(L_{yy}L_x^2 + L_{xx}L_y^2 - 2L_xL_yL_{xy})}{(L_x^2 + L_y^2)^{3/2}} \quad (4.3)$$

$$|\nabla L| = \sqrt{L_x^2 + L_y^2}$$

em que κ representa a curvatura, L_{xx}, L_{xy}, L_{yy} são as segundas derivadas de L, L_x^2 e L_y^2 são os quadrados das primeiras derivadas e $|\nabla L|$ é a magnitude do gradiente. A aplicação destas expressões encontra-se explicada em [18].

4.1.5. Modelos paramétricos deformáveis – contornos activos (*Snakes*)

Os modelos paramétricos deformáveis são usados para detectar contornos através da aplicação de curvas paramétricas. Esta ideia introduzida por Kass et al, em 1988 ^[40], interpreta as *snakes* como um caso geral de correspondência de um modelo deformável através da minimização da energia. Uma descrição bastante completa deste processo encontra-se em [27].

Uma *snake* é definida como um conjunto de pontos de controlo numa imagem, cada um com uma energia associada, que aumenta ou diminui conforme as forças que neles actuam. As forças internas (de elasticidade e de ligação) provocam um aumento da energia e são definidas pela relação entre o ponto de controlo de uma imagem e a sua vizinhança. Estas energias são responsáveis pela suavização dos contornos da imagem e dependem dos parâmetros intrínsecos da *snake*, como a curvatura ou o comprimento.

As forças externas provocam uma diminuição da energia e conduzem as *snakes* para características como as linhas e arestas, permitindo salientar características da imagem de maior detalhe. A energia externa é a garantia de minimização da energia da *snake*, o que significa que aos valores mínimos de energia estão associados os picos máximos dos vasos.

É extraída uma rede inicial de contornos, por estabelecimento de um valor de limiar, e é a partir desta que se iniciam as iterações para crescimento de contornos. ^[2] Durante o desenvolvimento, os contornos que se encontram em áreas onde não há arestas vão encolhendo até desaparecerem, enquanto que os outros vão crescendo ou separando até reconstruir a área de píxeis que estão acima do limiar estabelecido ^{[2], [4]}.

Uma outra técnica que explora o conceito de energia é a entropia local, explicitada em [11]. É estabelecido um limiar para detecção de vasos, que tem em conta a distribuição espacial dos níveis de cinzento dos píxeis. De duas imagens com histogramas idênticos, mas com uma distribuição espacial diferente, resulta numa entropia distinta, o que implica um estabelecimento de um limiar diferente. Este tipo de limite permite conservar os detalhes da imagem ^[11].

4.1.6. Esquemas matemáticos morfológicos

Os operadores morfológicos são, normalmente, aplicados a imagens binárias e de escala de cinzentos e permitem analisar, de uma forma mais clara, as diferentes regiões de uma imagem, conforme a sua intensidade. Estes operadores definem-se como erosão e dilatação, podendo ainda ser conjugados, para processos de *opening* e *closing*.

A erosão de uma imagem f por um elemento estruturante B (equivalente ao núcleo de convolução dos filtro lineares) é definida como ^[7]

$$\left[\varepsilon_B (f) \right] (x) = \min_{b \in B} \{ f(x+b) - B(b) \} \quad (4.4)$$

Cada píxel do objecto que esteja na vizinhança dos píxeis pertencentes ao plano de fundo passar a pertencer também ao plano de fundo.

A dilatação de uma imagem f é definida analogamente como ^[7]

$$\left[\delta_B (f) \right] (x) = \max_{b \in B} f(x+b) \quad (4.5)$$

Todos os vizinhos de um píxel do objecto que pertençam ao plano de fundo passam a pertencer ao próprio objecto, aumentando assim os contornos do objecto.

O gradiente morfológico apresenta-se como a diferença aritmética entre dilatação e erosão, como indica a expressão:

$$\rho(f) = \delta_B(f) - \varepsilon_B(f) \quad (4.6)$$

Pela conjugação destes métodos, obtém-se os processos de *opening* – erosão seguida de dilatação, usada para eliminar pequenas estruturas – e *closing* – dilatação seguida de erosão.^[41]

^[42]

O processo de *opening*, γ , de uma imagem é definida pela operação de erosão de f por B , seguido pela dilatação com um elemento transposto \tilde{B}

$$\gamma_B(f) = \delta_{\tilde{B}} \left[\varepsilon_B(f) \right] \quad (4.7)$$

O processo de *closing*, ϕ , corresponde a uma dilatação de f por B seguida por uma erosão por um elemento transposto \tilde{B}

$$\phi_B(f) = \varepsilon_{\tilde{B}} [\delta_B(f)] \quad (4.8)$$

A conjugação destes métodos permite a distinção das diferentes regiões que compõem uma imagem, facilitando a aplicação de algumas técnicas de detecção de vasos. Uma dessas técnicas designa-se por *top hat transformation*, que permite detectar elevações locais em planos de fundo arbitrários, realçando a forma dos vasos [7].

Um exemplo de aplicação destes operadores morfológicos permite estabelecer uma distinção entre objectos e, a partir daí, preencher as descontinuidades que possam apresentar. Para isso, são, normalmente, aplicados algoritmos de dilatação, diminuindo o espaço interno entre contornos. O efeito desta aplicação pode ser visto em [5].

4.1.7. Aproximação por crescimento de região

Este tipo de aproximação segmenta os píxeis da imagem em regiões, baseando-se em critérios de similaridade de valores e proximidade espacial. Isto é, dois píxeis podem ser agrupados se estiverem num intervalo de proximidade definido e apresentarem o mesmo valor de intensidade. [38], [39], [41], [43]

A forma mais simples de segmentação para a aplicação desta técnica passa por estabelecer um valor limite de intensidade para identificação de regiões e agrupar os píxeis conforme se encontrem ou não dentro desse limite. Para a identificação de múltiplas regiões, podem ser estabelecidos limiares adaptativos, que permita a distinção em diferentes regiões.

A segmentação pode também ser levada a cabo por detecção de arestas, e, a partir destas, extrair as regiões. Este método é essencialmente usado na detecção de mácula e disco óptico ou no desenvolvimento de grupos de píxeis que apresentam similaridades de determinadas características entre eles [4], [12].

As principais desvantagens deste método são a necessidade de um ponto de iniciação, implicando uma interacção do utilizador, e uma segmentação excessiva, pelas variações de intensidades e presença de ruído, requerendo um posterior processamento.

4.1.8. Correspondência de templates

A correspondência de *templates* reconhece um modelo de uma estrutura específica (*template*) numa imagem na sua totalidade, ou seja, funciona como um contexto da imagem e assemelha-se a um modelo desenhado *a priori*.

Uma abordagem deste processo encontra-se descrita em pormenor em [3]. Para construir o *template*, durante um processo de monitorização (não em tempo real), cada píxel de cada linha é verificado para a existência da primeira margem de um vaso vertical; caso este exista (se o filtro exceder algum valor mínimo), então todas as larguras que correspondem às esperadas para os vasos são verificadas para semelhança. São aplicados filtros horizontais e verticais a cada linha e coluna, respectivamente, e as respostas obtidas são ordenadas em tabelas e seleccionadas as melhores. Estas servirão para construir os *templates*, que serão usados na detecção dos vasos de uma imagem. Uma vez que os movimentos da retina estão limitados a translações, o deslocamento é calculado por avaliação das coordenadas de um conjunto restrito de pontos de controlo. ^{[4],[15],[16],[22]}

4.2. Determinação do movimento

A detecção de movimentos ao longo de sequências temporais envolve, na maior parte das vezes, uma correspondência entre pontos característicos da imagem em *frames* consecutivas. ^[4]

^[15] Ou seja, o estabelecimento de relações coerentes entre características permite inferir acerca da transformação sofrida entre duas imagens. A determinação do movimento pode ser efectuada a partir de modelos probabilísticos, determinação do plano de fundo ou cálculo do fluxo óptico.

4.2.1. Modelos probabilísticos

O aprimoramento da determinação de correspondências entre características da imagem e do respectivo movimento levou ao desenvolvimento de teorias muito completas e de aplicações dinâmicas, como o caso de inferências bayesianas de probabilidades ^{[28], [37], [44]} e filtros de Kalman. ^{[30], [31]} Estas duas abordagens têm uma relação muito estreita, uma vez que o filtro de Kalman é um filtro probabilístico, baseado na teoria bayesiana.

Este tipo de filtro permite estimar o estado de um sistema dinâmico por avaliação do estado no momento temporal anterior, através de uma predição seguida de uma correcção. A predição é baseada num modelo dinâmico e numa matriz de erro – erro de covariância, enquanto que a correcção é baseada num modelo de observação, recorrendo à actualização da medição e da covariância de erro. ^{[29], [30]}

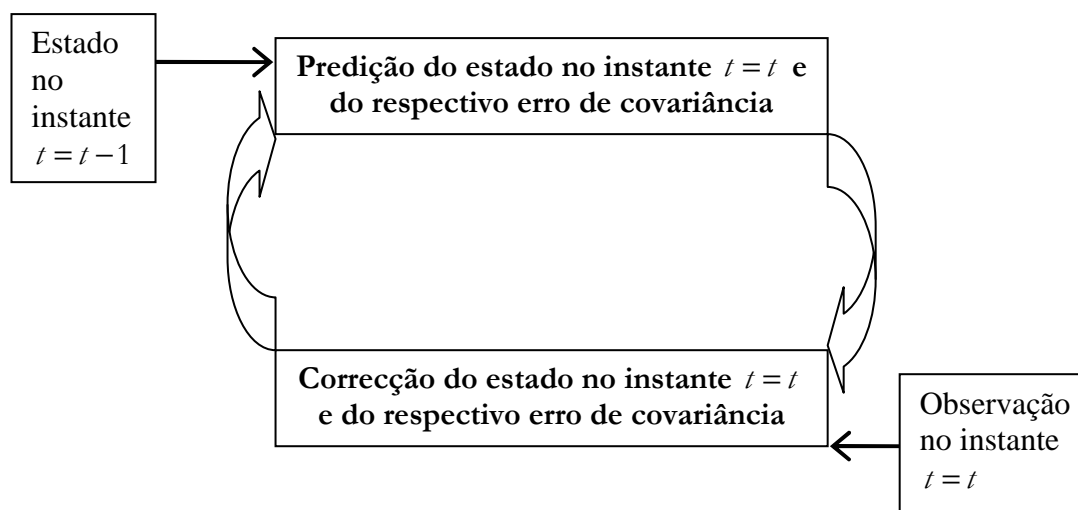


Figura 4.1. Esquemática do filtro de Kalman. Adaptado de [30].

As componentes deste filtro são, essencialmente, o vector de estado, o modelo dinâmico e o modelo de observação.

O vector de estado descreve o estado do sistema dinâmico e representa os graus de liberdade do sistema dinâmico. É composto por dois valores, referentes ao mesmo tempo, um estabelecido à priori, um valor previsto antes da actualização, e outro estabelecido à posteriori, corrigido após a actualização do estado. ^[29]

O modelo dinâmico descreve a transformação do vector de estado ao longo do tempo e é normalmente representado por um sistema de equações diferenciais, constituídas por uma matriz dinâmica, dependente do estado a ser estimado, e pelo vector de estado. ^[29]

O modelo de observação representa a relação entre o vector de estado e as medidas e é descrito por um sistema de equações lineares. ^{[29], [30]}

Este filtro pode ser estendido a sistemas não lineares, recorrendo a modelos dinâmicos reprogramados para casos específicos em que as situações não podem ser descritas por equações diferenciais – filtro de Kalman estendido. No entanto, tratam-se de algoritmos que necessitam de mais tempo de computação, comparativamente com as aplicações em sistemas lineares, uma vez que a actualização dos vectores de estado terá de ser ajustada a matrizes de transição e de observação dependentes do estado em cada momento. ^[29]

O recurso a estes filtros para detecção do movimento humano em sequências de imagens é descrito e exemplificado nos trabalhos de Miguel Velhote Correia e João Manuel Tavares em [32] e [33].

4.2.2. Subtracção do plano de fundo

Nesta abordagem, a detecção do movimento é efectuada pela modelação do plano de fundo ao longo da sequência temporal, e pelo cálculo da diferença entre a imagem actual e um fundo referenciado. Normalmente, este plano é modelado pela mediana dos valores dos píxeis ao longo de uma série de imagens e é preferencialmente estático. ^[22] No entanto, é preferível um modelo de fundo adaptativo às condições da imagem e, conseqüentemente, mais robusto e menos sensível a variações de intensidade de píxeis. Um dos exemplos de aplicação encontra-se descrito em [34].

4.2.3. Métodos estatísticos

Estes métodos surgem do desenvolvimento adaptativo do plano de fundo, em que são analisadas as características individuais dos píxeis ou de um conjunto de píxeis, para construção e actualização dos modelos. Cada píxel é classificado de forma comparativa com o modelo como pertencente ao plano de fundo ou ao objecto e pode ser modelado por uma mistura e conseqüente avaliação de funções gaussianas, de modo a determinar a sua

intensidade. Outra abordagem passa pela representação de cada píxel como um triplo de intensidades máxima e mínima e diferença máxima de intensidade entre *frames*.

Este avanço na determinação de planos de fundo leva a um desenvolvimento de um plano de fundo mais completo e robusto, permitindo ajustes dinâmicos em cada *frame* analisada, conforme os píxeis que a constituem e as condições de luminosidade. ^[22]

4.2.4. Fluxo óptico

O campo do fluxo óptico de uma imagem é a distribuição bidimensional da velocidade aparente dos píxeis, isto é, cada píxel tem associado um vector de velocidade, dependente do movimento a que é sujeito ^[23].

A proposta inicial de computadorização do fluxo óptico parte de Horn e Schunck, em 1981, baseado no cálculo das derivadas espaço-temporais da intensidade da imagem (equivalente ao seu mapa de brilho) ^[26] e assumindo uma intensidade I praticamente constante da imagem. Matematicamente, para um píxel com o vector de posição \vec{x} , num intervalo de tempo δt , a função de intensidade $I(\vec{x}, t)$ para uma imagem I é calculada por

$$I(\vec{x}, t) \approx I(\vec{x} + \delta\vec{x}, t + \delta t) \quad (4.9)$$

Aplicando uma expansão da série de Taylor, em que O representa os termos a partir da segunda ordem (e que podem ser desprezados), obtém-se a equação do gradiente

$$I(\vec{x} + \delta\vec{x}, t + \delta t) = I(\vec{x}, t) + \nabla I \cdot \delta\vec{x} + I_t \cdot \delta t + O^2 \quad (4.10)$$

que, de uma forma simplificada,

$$\nabla I(x, t) \cdot \mathbf{v} + I_t(x, t) = 0 \quad (4.11)$$

em que $\mathbf{v} = \left[\frac{\partial x}{\partial t}, \frac{\partial y}{\partial t} \right]$ representa o vector de fluxo óptico ^[26]. Esta equação é denominada a equação de restrição do fluxo óptico. No caso discreto, as aproximações efectuadas encontram-se descritas em [24].

No entanto, não é possível determinar directamente a componente nas direcções da velocidade ao longo dos contornos de igual brilho (*isobrightness*), dada a indeterminação do operador gradiente. Para superar este problema, foi implementado, por Lukas e Kanade, um método de cálculo de derivadas espaciais de primeira ordem aplicando o método de mínimos quadrados, para calcular o vector de fluxo ^{[25], [26], [27]}. A partir daqui, surge a expressão

$$E = \iint \left(\nabla I \cdot v - I_t + \lambda \nabla^2 (v_x + v_y) \right) dx dy \quad (4.12)$$

Esta expressão recorre ao cálculo de multiplicadores de Lagrange e encontra-se descrita em [25].

Como foi analisado, a conjugação de múltiplas técnicas de segmentação permite a detecção de pontos de controlo de uma imagem, que se destacam pela sua curvatura, intensidade ou localização específica. A sua determinação serve de base à aplicação de algoritmos para análise de movimento ou de transformação sofrida entre imagens, permitindo inferir acerca das suas posições com o decorrer do tempo. É, por isso, fundamental estudar a imagem como um conjunto de diferentes características, avaliando a sua evolução ao longo das sequências temporais.

Referências Bibliográficas

- [1] Kollner, Th. M. "Multiscale Detection Of curvilinear Structures in 2D and 3D image Data." 1995 7 August 2007
<<http://cobnitz.codeen.org:3125/citeseer.ist.psu.edu/cache/papers/cs/26735/ftp:zSzzSzf. vision.ee.ethz.chzSzpublicationszSz1995zSzpostscriptszSzkollerICCV95.pdf/koller95multisc ale.pdf>>.
- [2] Solouma, Nahed H.. "A new real time retinal tracking system for image guided laser treatment." IEEE Transactions on biomedical engineering 49, num 9Sept 2002 1059 - 1067. 16 Sep 2006
ieeexplore.ieee.org/iel5/10/22092/01028431.pdf?isnumber=&arnumber=1028431
- [3] Markow, Michael. "Real time algorithm for retinal tracking." IEEE Transactions on biomedical engineering 40, num 12 December 1993 1269 - 1281. 16 Sep 2006
<ieeexplore.ieee.org/iel5/10/6416/00250583.pdf?arnumber=250583>.
- [4] Solouma , Nahed H.. "Real time retinal tracking for laser treatment planning and administration." Medical imaging 2001: image processing 4322(2001)2001 1311 - 1321. 21 Feb 2007 <<http://www.bme.emory.edu/~ykadah/papers/spie-01-3.pdf>>.
- [5] Hirata, Nina S. T. . "Bases da Morfologia Matemática." 4 Sep 2007
<<http://www.vision.ime.usp.br/~abvg/mm/mm2/index.html>>.
- [6] Sofka, Michal. "Retinal vessel centerline extraction using multiscale matched filters, confidence and aresta measures." 2 July 2006 16 Sep 2006
<<http://citeseer.ist.psu.edu/cache/papers/cs2/17/http:zSzzSzwww.vision.cs.rpi.eduzSzpublicationszSzpdfszSzsofka-tmi-draft.pdf/sofka06retinal.pdf>>.
- [7] Medeiros, Nilcilene das Graças. "Segmentação morfológica de imagens utilizando o gradiente morfológico multi escala." IMTC 2005 instrumentation and measurement 17 - 19 May 2005 222 - 227. 26 Sep 2006 <www.rbc.ufrj.br/_pdf_54_2002/54_08.pdf>.
- [8] Englmeier, K.H.. "Multi resolution retinal vessel tracker based on directional smoothing." Medical imaging 46832002 230 - 237. 17 Feb 2007
<<http://citeseer.ist.psu.edu/cache/papers/cs/26619/http:zSzzSzwww.isi.uu.nlzSzMeetingszSz.zSzTGVzSzjoes2.PDF/multi-resolution-retinal-vessel.pdf>>.
- [9] Kondo, Toshiaki. "Detection of anatomical features in retinal images using a gradient orientation." 155 - 158. 26 Feb 2007
<ieeexplore.ieee.org/iel5/9709/30646/01414380.pdf?arnumber=1414380>.
- [10] Hoover, Adam. "Locating blood vessels in retinal images by piecewise threshold probing of a matched filter response." Transactions on medical imaging 19March 2000 203 - 210.

- [11] Chanwimaluang, Thitiporn. "An efficient algorithm for extraction of anatomical structures in retinal images." 16 Sep 2006
<<http://ieeexplore.ieee.org/Xplore/login.jsp?url=/iel5/8824/27937/01247157.pdf>>.
- [12] Staal, Joes. "Ridge-Based Vessel Segmentation in colour images of the retina." Transactions on medical imaging 23, no 4 April 2004 501 - 509. 12 Nov 2006
<ieeexplore.ieee.org/iel5/42/28634/01282003.pdf?arnumber=1282003>.
- [13] Lalonde, Marc. "Non-recursive paired tracking for vessel extraction from retinal images." 16 Sep 2006 <<http://www.crim.ca/perso/langis.gagnon/articles/vi2000.pdf>>.
- [14] Grisan, Enrico . "A new tracking system for the robust extraction of retinal vessel structure." Proceedings oh the 26th annual international conference of the IEEE EMBS Sep 2004 23 Nov 2006
<ieeexplore.ieee.org/iel5/9639/30462/01403491.pdf?arnumber=1403491>.
- [15] Becker, Douglas. "Real-time image processing algorithms for an automated retinal laser surgery system" IEEE 1995 426 - 429. 17 Feb 2007
<ieeexplore.ieee.org/iel3/4052/11606/00529737.pdf?arnumber=529737>.
- [16] Becker, Douglas. "Image Processing Algorithms for Retinal Montage Synthesis, Mapping, and Real-Time Location Determination." IEEE transactions on Biomedical Engineering Vol. 45, No. 1 January 1998 105 - 117. 26 Sep 2006
<<http://citeseer.ist.psu.edu/cache/papers/cs/32913/http:zSzzSzwww.cs.rpi.edu:z~canazSzPublicationszSzJournalPaperszSzIEEEbiomedicalEng-Becker.pdf/becker98image.pdf>>.
- [17] Pinz, Axel. "Mapping the Human retina." IEEE transactions on medical Imaging vol. 17, no. 4 August 1998 606 - 619. 26 Sep 2006
<<http://citeseer.ist.psu.edu/cache/papers/cs/14174/http:zSzzSzwww.emt.tu-graz.ac.at:z~pinzzSzonlinepaperszSzIEEETMI98.pdf/mapping-the-human-retina.pdf>>.
- [18] Bretzner, Lars. "Feature tracking with automatic selection of spacial scales." 5 March 2007
<<http://citeseer.ist.psu.edu/cache/papers/cs/26438/ftp:zSzzSzftp.nada.kth.se:zCVAPzSzreportszSzcvap201.pdf/bretzner96feature.pdf>>.
- [19] Poli, Riccardo. "An algorithm for real time vessel enhancement and detection." 15 Nov 2006
<<http://citeseer.ist.psu.edu/cache/papers/cs/16175/http:zSzzSzwww.cs.bham.ac.uk:z~rmpzSzpaperszSzPoli-CMPB1996.pdf/poli96algorithm.pdf>>.

- [20] Parker, J.. "Advanced Aresta Detection Techniques: The Canny and the Shen-Castan Methods." Techniques in Computational Vision. 7 April 2007
- [21] Tuytelaars, T. et al "A Survey on Local Invariant Features." 5 May 2006 1-64. 28 February 2007 http://homes.esat.kuleuven.be/~tuytelaa/survey_inv_features.pdf.
- [22] Wang, Liang. "Recent Developments in Human Motion Analysis." <<http://nlpr-web.ia.ac.cn/English/irds/papers/wangliang/PR%20.pdf>>.
- [23] Barbosa, Ricardo Luis. "A Computação de Fluxo óptico em imagens obtidas por um sistema móvel de mapeamento terrestre." Revista Brasileira de Cartografia 572005 72 - 78. 30 September 2006 <http://www.rbc.ufrj.br/_pdf_57_2005/57_2_02.pdf>.
- [24] Horn, B. K. P. . Robot Vision. 1st Ed. The MIT Press, 1986.
- [25] Streeter, Lee. "A direct geometric algebra Optical flow solution." 19 March 2007 <<http://pixel.otago.ac.nz/ipapers/26.pdf>>.
- [26] Paragios, Nikos . Handbook of Mathematical Models in Computer Vision. 1st ed. Springer, 2006.
- [27] McInerney, Tim. "Deformable models im medical image analysis : a survey." Medical image analysis 1996 1-26. 26 July 2007 <www.cs.ucla.edu/~dt/papers/mia96/mia96.pdf>.
- [28] Jähne, Bernd . Handbook of Computer Vision and Applications - Volume 2 - signal processing and Pattern Recognition. 1st Ed. Academic Press, 1999
- [29] Welch, Greg et al. "An Introduction to the Kalman Filter." SIGGRAPH 2001 1 - 47. 19 February 2007 <www.cs.unc.edu/~welch/media/pdf/kalman_intro.pdf>.
- [30] Kleinbauer, Rachel. "Kalman Implementation with Matlab." Universitat Stuttgart November 2004 1-35. 19 February 2007 <elib.unistuttgart.de/opus/volltexte/2005/2183/pdf/kleinbauer.pdf>.
- [31] Grewal, Mohinder et al Kalman Filtering - Theory and Practice using Matlab. 2nd. Wiley - interscience, 2001.
- [32] Pinho, Raquel. et al. "An Efficient and Robust Tracking System using." 28 June 2007 <http://paginas.fe.up.pt/~tavares/downloads/publications/artigos/VIPSIVenice2006_Raquel.pdf>.
- [33] Pinho, Raquel. "Correspondência entre pontos no seguimento de movimento entre imagens." 28 June 2007 <http://paginas.fe.up.pt/~tavares/downloads/publications/artigos/6APAET_RP_JT_MC.pdf>.
- [34] Stauffer, Chris . "Adaptive background mixture models for real-time tracking." 25 September 2006 <www.ai.mit.edu/projects/vsam/Publications/stauffer_cvpr98_track.pdf>.

- [35] Fidrich, Marta. "Multiscale Extraction and Representation of Features from Medical Images." October 1994 13 April 2007
<<http://citeseer.ist.psu.edu/cache/papers/cs/14912/ftp:zSzzSzfpt.inria.frzSzINRIAzSzpubli-icationzSzpubli-ps-gzzSzRRzSzRR-2365.pdf/fidrich94multiscale.pdf>>.
- [36] Kadi, Timor . "Saliency, Scale and Image Description." 2000 25 May 2007
<www.robots.ox.ac.uk/~timork/Saliency/ijcv_SalScale.pdf>.
- [37] Goldbaum, Michael . "Automated diagnosis and image understanding with object extraction, object classification and inferencing in retinal images." 15 November 2006
<people.brandeis.edu/~altaylor/icip-1996-analysis.pdf>.
- [38] Shapiro, Linda. Computer Vision. 1st ed. Prentice Hall, 2000.
- [39] Mubarak, Shah. Fundamentals of Computer Vision. 1997. 6 April 2007
<<http://www.cs.ucf.edu/courses/cap6411/book.pdf>>.
- [40] - M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models", *Int. J. of Comp. Vision*, vol. 1, pp. 321-331, 1988.
- [41] Gonzalez, Rafael. Digital Image Processing Using Matlab. New Jersey: Prentice Hall, 2004.
- [42] Ritter, Gerhard X.. Handbook of Computer Vision Algorithms in Image Algebra. 1st Ed. CRC Press, 2001
- [43] Acharya, Tinku . Image Processing - Principles and Applications. 1st Ed. Wiley, 2005.
- [44] Forsyth, David A. . Computer Vision: A Modern Approach. 1st Ed. Prentice Hall, 2003.

PARTE II
DESENVOLVIMENTO

5. *Retinal Tracking*: introdução de conceitos

O algoritmo de detecção do movimento desenvolvido envolve a aplicação de conceitos matemáticos, tais como derivadas de segunda ordem ou correlação, bem como a implementação de um algoritmo de eliminação de erros e estimação de modelos – RANSAC. Para uma melhor compreensão das suas aplicações, será feita uma breve introdução teórica dos conceitos empregues.

5.1. Derivadas parciais de segunda ordem

Considere-se uma função vectorial $f : S \rightarrow \mathbb{R}^m$ definida num conjunto S em \mathbb{R}^n com valores em \mathbb{R}^m . Seja $f_i : S \rightarrow \mathbb{R}$ ($i = 1, \dots, m$) a componente i da função f , com derivadas parciais num ponto interior c e em cada ponto de uma vizinhança aberta deste. Então, pode considerar-se o limite

$$\lim_{t \rightarrow 0} \frac{(D_j f_i)(c + te_k) - (D_j f_i)(c)}{t} \quad (5.1)$$

em que D_j é a primeira derivada em função de j , e e_k é o k -ésimo vector unitário em \mathbb{R}^n . Quando este limite existe, é considerado a derivada de segunda ordem de f_i em c e denotado $D_{kj}^2 f_i(c)$. Ou seja, a derivada de segunda ordem é obtida por diferenciação parcial de f_i em relação à k -ésima variável e posterior diferenciação parcial do resultado $D_j f_i$ em respeito à j -ésima variável. ^[1]

No entanto, ao tratar-se de imagens descritas como funções bidimensionais discretas, é efectuada uma aproximação às derivadas de segunda ordem, recorrendo à fórmula de Taylor de segunda ordem. Esta é expressa por

$$\phi(c + u) = \phi(c) + (D\phi(c))u + \frac{1}{2} u' B u + r(u) \quad (5.2)$$

em que $\phi(c)$ é a função no ponto c , $D\phi(c)u$ é a sua derivada em função de u , B é uma matriz simétrica, dependente de c mas não de u e em que $\lim_{u \rightarrow 0} \frac{r(u)}{\|u\|^2} = 0$. [1]

Tendo em conta que o segundo diferencial é simplesmente o diferencial do diferencial,

$$d^2 f = d(df) \quad (5.3)$$

Pode definir-se, para uma função de valores reais $\phi: S \rightarrow \mathbb{R}$, que seja duas vezes diferenciável num ponto $c \in S$, uma função $\psi(x)$ que represente o seu diferencial, isto é,

$$\psi(x) = d\phi(x; u) \quad (5.4)$$

Então, o diferencial em função de x é dado por

$$\psi(x) = \sum_{j=1}^n u_j D_j \phi(x) \quad (5.5)$$

Para diferenciais de segunda ordem,

$$D_i \psi(x) = \sum_{j=1}^n u_j D_{ij}^2 \phi(x), \quad (i=1, \dots, n) \quad (5.6)$$

O diferencial de segunda ordem de ϕ , por definição, é equivalente ao diferencial de primeira ordem de ψ , e então

$$d^2 \phi(x; u) = u'(H\phi(x))u \quad (5.7)$$

À matriz composta pelas segundas derivadas chama-se matriz Hessiana. Esta é definida, para uma função de valores reais $\phi: S \rightarrow \mathbb{R}, S \subset \mathbb{R}^n$, como uma matriz quadrada expressa por

$$H\phi(c) = \begin{bmatrix} D_{11}^2 \phi(c) & D_{12}^2 \phi(c) & \cdots & D_{1n}^2 \phi(c) \\ D_{21}^2 \phi(c) & D_{22}^2 \phi(c) & \cdots & D_{2n}^2 \phi(c) \\ \vdots & \vdots & \vdots & \vdots \\ D_{n1}^2 \phi(c) & D_{n2}^2 \phi(c) & \cdots & D_{nn}^2 \phi(c) \end{bmatrix} \quad (5.8)$$

para um ponto c de S , em que as n^2 derivadas de segunda ordem, $D_{kj}^2\phi(c)$, existem.

Esta matriz é simétrica, ou seja, se ϕ é duas vezes diferenciável em c então, pelo teorema de Schwartz,

$$D_{jk}^2\phi(c) = D_{kj}^2\phi(c), \quad k, j = 1, \dots, n \quad (5.9)$$

Note-se que

$$H\phi(x) = \frac{\partial}{\partial x'} \left(\frac{\partial \phi(x)}{\partial x'} \right)' = D(D\phi(x))' \quad (5.10)$$

Na área de tratamento de imagem, esta matriz é aplicada para detecção dos pontos de interesse. Os seus valores próprios permitem inferir acerca de mudanças no vector normal ao nível de intensidade passando por um ponto particular, fornecendo informações acerca da forma dos vasos. Nas localizações centradas em estruturas tubulares, o valor próprio de H é próximo de zero, reflectindo a ausência de uma curvatura ao longo da direcção do vaso. À medida que a curvatura se altera, o valor próprio assume valores diferentes, consoante se trata de um mínimo ($\lambda > 0$) ou de um máximo local ($\lambda < 0$). É através da análise destes valores que é possível avaliar o comportamento dos vasos. O cálculo dos vectores e valores próprios de uma matriz é explicado no anexo B. ^{[2], [3], [4]}

5.2. Correlação

A correlação é uma função matemática que permite analisar o deslocamento sofrido entre duas imagens semelhantes e o seu cálculo depende da posição relativa dos valores dos píxeis numa região limitada. As características da imagem que permitem estabelecer a função são distinguidas por representarem um padrão que produz variações de intensidades quando a luz reflecte na sua superfície ^[6], o que torna uma forma de comparação viável, mesmo em condições de diferenças de luminosidade. Por exemplo, nas imagens do fundo da retina, os vasos são distinguidos do plano de fundo por se apresentarem com uma intensidade de luminosidade diferente. Após a determinação de pontos característicos das imagens, um dos pontos da primeira imagem é comparado com os da segunda, dentro de um determinado intervalo de distância. Neste intervalo, é procurada a posição correspondente ao valor que

assegura o maior grau de semelhança entre as duas imagens. ^[8] A figura seguinte ilustra esta situação.

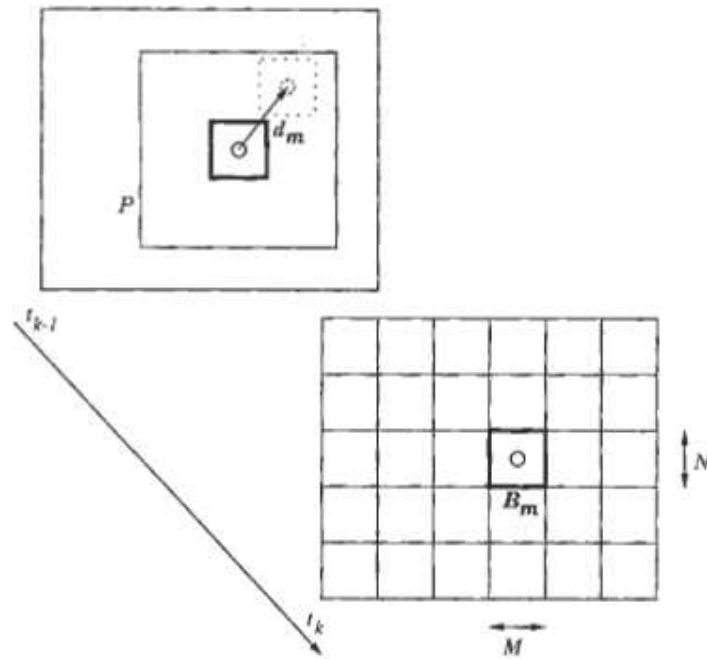


Figura 5.1. Correspondência de blocos. [9]

No caso de valores discretos, como no espaço matricial das imagens, considera-se uma janela k , de largura w e altura h , ambos valores ímpares, de tal forma que $m = (w - 1) / 2$ represente metade da largura e $n = (h - 1) / 2$ a metade da altura. Para uma imagem f , a correlação é então expressa por

$$C(x, y) = \sum_{j=-n}^n \sum_{i=-m}^m k(i, j) f(x+i, y+j) \quad (5.11)$$

Muitas vezes este valor é normalizado, dividindo o valor da correlação pela soma dos níveis de cinzento ^[7]:

$$C_{norm}(x, y) = \frac{\sum_{j=-n}^n \sum_{i=-m}^m k(i, j) f(x+i, y+j)}{\sum_{j=-n}^n \sum_{i=-m}^m f(x+i, y+j)} \quad (5.12)$$

A janela da primeira imagem será construída com um tamanho $p = (2m + 1) \times (2n + 1)$ e será centrada em (x, y) , com um vector $w(x, y)$. Na segunda imagem, a correspondência terá uma posição $(x + d_1, y + d_2)$, com um vector $w'(x + d_1, y + d_2)$, tal que a função de correlação seja expressa por

$$C(d) = \frac{1}{|w - \bar{w}|} \frac{1}{|w' - \bar{w}'|} (w - \bar{w}) \cdot (w' - \bar{w}') \quad (5.13)$$

onde \bar{w} representa o vector cujas coordenadas são a média das coordenadas de w . A figura 5.2. representa a correlação ao longo de uma linha. [8]

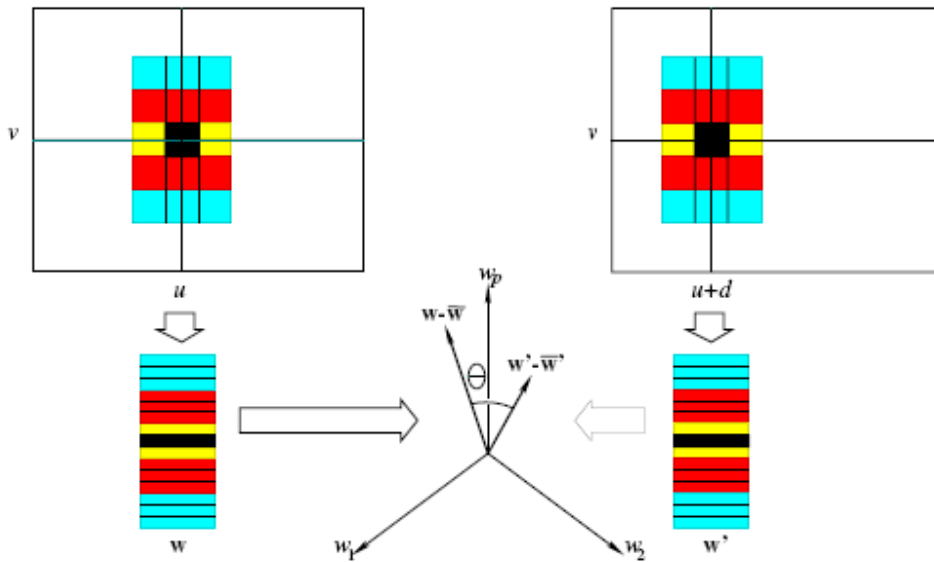


Figura 5.2. Exemplo da correlação entre duas janelas. [8]

O ângulo θ será calculado a partir do inverso do cosseno entre os vectores $w - \bar{w}$ e $w' - \bar{w}'$.

Os valores desta função variam entre -1 e 1, alcançando o seu máximo quando os padrões apenas diferem num valor constante de escala. Maximizando a função de correlação, há uma

minimização da norma da diferença entre os vectores $\frac{1}{|w - \bar{w}|} (w - \bar{w})$ e $\frac{1}{|w' - \bar{w}'|} (w' - \bar{w}')$.

Ou seja, minimiza a soma das diferenças quadradas dos valores dos píxeis das janelas de comparação [8].

Em termos de produto interno vectorial, os vectores serão paralelos, o que implica uma maximização do coeficiente de correlação cruzada normalizada, onde f representa a imagem.
[10]

$$CC(x, y) = \frac{\sum_x \sum_y [f(x, y) - \bar{f}(x, y)][w(x - x', y - y') - \bar{w}]}{\left(\sum_x \sum_y [f(x, y) - \bar{f}(x, y)]^2 \sum_x \sum_y [w(x - x', y - y') - \bar{w}]^2 \right)^{1/2}} \quad (5.14)$$

Este coeficiente tem valores nulos quando não há qualquer similaridade entre os padrões e tem um valor máximo quando há correspondência ideal.

5.3. Transformações entre imagens

5.3.1. Transformações Euclidianas

As transformações euclidianas correspondem a uma modelação do movimento de um corpo rígido, isto é, que não sofre deformações com a alteração da posição. Mais concretamente, uma transformação planar deste género pode ser expressa por

$$x' = H_E x = \begin{bmatrix} R & t \\ 0^T & 1 \end{bmatrix} x \quad (5.15)$$

equivalente a

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta & t_x \\ \sin \theta & \cos \theta & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (5.16)$$

em que R representa uma matriz bidimensional de rotação, t representa uma matriz-coluna de 2 elementos de translação. No caso de R ser a matriz identidade ($\cos \theta = 1$), ter-se-ão ângulos nulos, o que implica uma translação pura. Esta transformação tem dois graus de liberdade para a translação e um grau de liberdade para a rotação. Ou seja, será necessário especificar três parâmetros para definir uma transformação euclidiana.

Os invariantes desta transformação, isto é, os elementos que permanecem inalterados durante a transformação, são a distância entre dois pontos e o ângulo entre duas linhas e a área.^[11]

5.3.2. Transformações de similaridade

Uma similaridade é uma transformação composta por uma escala isotrópica. Isto significa que a matriz de rotação é multiplicada por um escalar s , representativo da escala em que ocorre a transformação. Esta transformação preserva a forma e tem quatro graus de liberdade, uma vez que um deles será correspondente à escala (o outro à rotação e os outros dois à translação, como na transformação euclidiana): ou seja, a similaridade pode ser calculada a partir de dois pares de pontos correspondentes. Ao contrário da transformação euclidiana, a distância entre dois pontos não é invariante; no entanto, a razão entre duas distâncias é, uma vez que os factores de escala se anulam. Matricialmente,

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \varepsilon \cos \theta & -\varepsilon \sin \theta & t_x \\ \varepsilon \sin \theta & \varepsilon \cos \theta & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (5.17)$$

ε pode assumir valores de 1 ou -1 e quando $\varepsilon = 1$ há uma preservação da orientação e considera-se a transformação como euclidiana, isto é, composta apenas por rotação e translação. No caso de assumir o valor negativo, ocorre um reverso da orientação, ou seja, uma reflexão.^[11]

5.3.3. Transformações afins

Uma transformação afim é uma transformação linear não-singular, seguida de uma translação. Matricialmente,

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (5.18)$$

Estabelecendo uma matriz $A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$ e um vector $t = [t_x \quad t_y]^T$, a matriz anterior é equivalente a

$$x' = H_A x = \begin{bmatrix} A & t \\ 0^T & 1 \end{bmatrix} x \quad (5.19)$$

Esta transformação tem em conta uma deformação nos ângulos internos, isto é, se tiver, por exemplo, um rectângulo, este assemelhar-se-á a um paralelogramo após a transformação, pois os seus ângulos internos serão alterados. Esta constatação leva a concluir que os ângulos entre duas linhas não são uma característica invariante, como acontecia com a transformação Euclidiana.

Para entender os efeitos de uma transformação linear, considere-se esta como a composição de rotações e escala anisotrópicas. Esta propriedade estabelece uma variação das características ao longo da direcção. A matriz A pode ser então entendida como

$$A = R(\theta)R(-\phi)DR(\phi) \quad (5.20)$$

em que $R(\theta)$ e $R(\phi)$ são rotações, ϕ especifica o ângulo de rotação, de modo a indicar a direcção em que esta é feita, e D é a matriz diagonal

$$D = \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \quad (5.21)$$

Esta decomposição é obtida pela diagonalização da matriz A ^[13], da forma

$$A = UDV^T = (UV^T)(VDV^T) = R(\theta)(R(-\phi)DR(\phi)) \quad (5.22)$$

A matriz A assemelha-se a uma concatenação de uma rotação em ϕ , seguida de um ajuste de escala por λ_1 e λ_2 , rodada em x e y , e outras duas rotações, em $-\phi$ e θ . A existência de

um ângulo ϕ comprova a anisotropia da transformação afim, o que acrescenta mais dois graus de liberdade, relativamente às transformações de similaridade, dados pela direcção da escala e pela razão entre λ_1 e λ_2 . Ter-se-á então uma transformação planar afim caracterizada por seis graus de liberdade, estabelecidos a partir de três pares de correspondências. Os factores invariantes neste tipo de transformações reduzem-se então ao paralelismo de linhas, à razão entre áreas e à razão entre os comprimentos de segmentos de linha paralelos. ^{[11], [12]}

5.3.4. Transformações projectivas

Surgem, por fim, as transformações projectivas, entendidas como uma transformação linear não-singular de coordenadas homogéneas. Trata-se de uma generalização das transformações afins, em que a última linha é generalizada.

Através da geometria projectiva, pretende desenvolver-se uma relação matemática entre o mundo tridimensional com o captado por uma câmara e registado num plano a duas dimensões. Ou seja, há uma relação entre sistemas de coordenadas para representação geométrica de pontos.

Tem-se então

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (5.23)$$

$$x' = H_p x = \begin{bmatrix} A & t \\ \mathbf{v}^T & v \end{bmatrix} x \quad (5.24)$$

em que $\mathbf{v} = (v_1, v_2)^T$. Note-se que este vector era nulo na transformação afim.

Relacionando as coordenadas homogéneas e cartesianas, obtém-se ^[11]



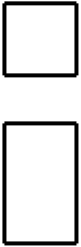
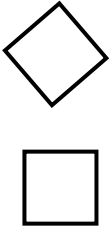
$$x' = \frac{b_{11}x + b_{12}y + b_{13}}{b_{31}x + b_{32}y + b_{33}} \quad (5.25)$$

$$y' = \frac{b_{21}x + b_{22}y + b_{23}}{b_{31}x + b_{32}y + b_{33}} \quad (5.26)$$

A resolução destas igualdades implica que existam 8 equações para determinar as incógnitas (assumindo, à partida, numa projecção planar, $b_{33} = 1$). Para isso são necessários 4 pares de pontos correspondentes, garantindo que quaisquer três sejam não colineares. ^[11]

As transformações apresentadas encontram-se sintetizadas na tabela seguinte. A partir destas matrizes, é possível inferir a posição actual dos píxeis para que o movimento relativamente à posição anterior seja compensado.

Tabela 1 resumo das transformações entre imagens e as respectivas propriedades. Adaptado de [11].

Grupo	Matriz	Invariantes	Exemplos
Projectiva 8 graus de liberdade	$\begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix}$	Colinearidade Intersecção Tangencia Inflexões Concorrência	
Afim 6 graus de liberdade	$\begin{bmatrix} a_{11} & a_{12} & t_x \\ a_{21} & a_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix}$	Paralelismo Razão entre áreas Razão entre comprimentos de linhas paralelas Combinações lineares entre vectores.	
Similaridade 4 graus de liberdade	$\begin{bmatrix} sr_{11} & sr_{12} & t_x \\ sr_{21} & sr_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix}$	Razão entre comprimentos Ângulos internos	
Euclidiana 3 graus de liberdade	$\begin{bmatrix} r_{11} & r_{12} & t_x \\ r_{21} & r_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix}$	Comprimento (distância entre pontos) Área	

5.4. RANSAC

No estabelecimento de correspondências entre imagens, com características invariantes no tempo, considera-se, muitas vezes, a distância entre os pontos característicos da imagem, de acordo com um determinado valor de limiar, a sua similaridade, ou a distância entre píxeis vizinhos, para o estabelecimento de pontos comuns. Estas abordagens têm como pano de fundo o método de mínimos quadrados, para uma estimativa de parâmetros, de forma a otimizar uma relação entre os pontos e o modelo escolhido. No entanto, levantam-se questões problemáticas no tratamento de distorções, ou de estruturas semelhantes, repetidas ao longo da imagem, uma vez que as técnicas utilizadas não possuem mecanismos internos para detecção e correcção de erros. Os métodos aplicados partem simplesmente da suposição que o desvio máximo esperado entre quaisquer pontos de um modelo depende do tamanho do conjunto de pontos escolhido, o que implica ter sempre pontos suficientemente robustos para compor o modelo. Além disso, os conjuntos de pontos a relacionar em cada imagem nem sempre são simétricos, uma vez que os que são identificados numa das imagens não são necessariamente todos os identificados na outra. ^[13]

Para ultrapassar estas adversidades, é introduzido, por volta da década de 80, um novo modelo para o estabelecimento de correspondências, Random Sample Consensus – RANSAC –, por Martin A. Fischler e Robert C. Bolles, que serviu, em primeira fase, para adaptação de modelos a dados experimentais ou análises cartográficas ^[14]. A interpretação fidedigna de resultados relaciona-se com a capacidade de encontrar, de forma iterativa, a melhor correspondência entre os dados e os modelos de identificação criados, e com a determinação dos melhores valores para o modelo estimado.

5.4.1. *Descrição do algoritmo RANSAC*

Este processo pode ser visto como um algoritmo de optimização, uma vez que permite uma parametrização das melhores posições de um determinado modelo, sem que seja necessário recorrer qualquer tipo de informação local acerca do objectivo. Em vez disso, recorre a uma amostragem de uma função descritiva do modelo em localizações discretas, criando aleatória e consecutivamente pequenos subconjuntos de dados, gerando soluções para o modelo pretendido. A sua aleatoriedade permite uma maior autonomia na inicialização do algoritmo. Estas suas particularidades tornam-no num método de eliminação de erros de classificação (detecção incorrecta de características da imagem) e medição (identificação das mesmas

características, mas com uma determinação incorrecta dos seus parâmetros) empregue em inúmeras situações. ^[14]

As amostras dos subconjuntos são desenvolvidas uniforme e aleatoriamente a partir dos dados de entrada. Cada ponto tem, à partida, a mesma probabilidade de selecção (daí ser uma amostragem uniforme) e, para cada amostragem, é construído um modelo hipotético e parametrizado de pontos. O tamanho da amostra depende dos pontos necessários para a determinação dos parâmetros do modelo. A consideração de um maior número de pontos além dos estritamente necessários à construção do modelo revela-se uma má opção, na medida em que a probabilidade de ter uma amostra consistente com o modelo construído diminui com o aumento do número de pontos escolhidos. ^[15]

Seguidamente, é avaliada a qualidade do modelo hipotético em todo o conjunto de pontos, através de uma função de custo. Para isso, é contabilizado o número de *inliers*, isto é, o conjunto de pontos que estão de acordo com o modelo, dentro de uma determinada tolerância de erro. O modelo que tiver o maior número de *inliers* é considerado o que melhor se adequa à amostra a considerar. ^[15]

Suponha-se o exemplo de uma estimativa de um ajuste de uma linha recta a um conjunto bidimensional de pontos. Dado este conjunto, o objectivo será estimar a linha que minimiza uma regressão ortogonal, com a condição que nenhum dos pontos válidos para a construção da linha se desvie além de um limite t pré-definido. Os problemas para esta estimativa encontram-se em classificar os pontos como válidos – *inliers* – e em estimar correctamente o ajuste de linha. Com o algoritmo de RANSAC, dois dos pontos são seleccionados aleatoriamente, definindo o modelo de uma linha. A robustez desta será definida pelo número de *inliers* que possui, dentro do limite estabelecido. Esta selecção é repetida N vezes e linha mais robusta é considerada como melhor ajuste. Os pontos considerados *inliers* fazem parte do *conjunto consensus* ^[16].

Numa linguagem mais exacta: ao ter um modelo com um mínimo de n pontos como requisito de estabelecimento de parâmetros livres, e um conjunto de pontos P , tal que o número de P é maior que n , é seleccionado aleatoriamente um subconjunto $S1$ de n pontos do conjunto P , de modo a definir o modelo usado – $M1$. Através deste, determina-se um subconjunto $S1^*$ de pontos de P dentro de um erro de tolerância. $S1^*$ é denominado o *conjunto consensus* de $S1$. Se $S1^*$ é maior que um limiar pré-definido T

(determinado em função da estimativa de números de erros e que estipula o tamanho máximo da amostra a considerar), utiliza-se $S1^*$ para determinar um novo modelo $M1^*$. Caso o conjunto seja menor que o limiar, selecciona-se aleatoriamente um subconjunto $S2$ e o processo é repetido; se, após um determinado número de ensaios, não há um consenso com t ou mais membros, termina-se o processo com o número encontrado ou dá-se início a um novo processo, desde o princípio (e considera-se que este falhou).^[14]

Há dois pontos a destacar neste processo: em primeiro, no caso de haver um problema relacionado com a selecção de pontos de S , há uma selecção determinística, em vez da habitual aleatoriedade. Em segundo, ao haver um conjunto S^* que seja compatível com o desejado, e um modelo M^* inicializado, pode haver adição de quaisquer novos pontos de P compatíveis, criando um novo modelo baseado num conjunto de pontos mais alargado.^[14]

5.4.2. Variáveis do processo

São considerados três parâmetros fulcrais para uma correcta aplicação do algoritmo: a tolerância de erro, usada na determinação de compatibilidade de pontos num modelo; o número de subconjuntos testados na aplicação; e um valor de limiar t , que representa o número de pontos compatíveis usados para classificar um modelo como viável^[14].

i. Tolerância de erro para estabelecimento de compatibilidade:

O desvio dos dados em relação a um modelo proposto é dependente dos erros associados aos dados e ao modelo. Se este último é, simplesmente, uma função dos dados, pode estabelecer-se limites analíticos para uma tolerância de erro. Contudo, não pode ser aplicado em todos os casos, daí que os limites possam ter que ser estabelecidos experimentalmente. As amostras de desvio podem ser induzidas por perturbação dos dados, medindo os erros implícitos. O factor de tolerância encontrar-se-á, nestes casos, compreendido entre valores de desvio padrão.

O desvio registado é, geralmente, função dos dados, e, por isso, a tolerância deverá ser diferente para cada conjunto avaliado. Contudo, verifica-se que a sua variação pode ser

desprezada face aos erros, assumindo-se, então, uma tolerância uniforme para todos os conjuntos, sem que isso prejudique a robustez do modelo ^[14].

ii. *Número máximo de tentativas para encontrar um conjunto consensus*

Sendo w a probabilidade de um ponto seleccionado ser *inlier*, então $\varepsilon = 1 - w$ será a probabilidade ser outlier.

São necessárias N selecções, cada uma de n pontos, em que $(1 - w^s)^N = 1 - p$, ou seja,

$$N = \frac{\ln(1 - \rho)}{\ln(1 - (1 - \varepsilon)^\rho)} \quad (5.27)$$

em que ρ é o grau de confiança e ε a fracção de falsas correspondências (outliers). ^[14]

A decisão de interromper a selecção de novos subconjuntos de P pode ser baseada no número de tentativas programadas N , necessárias para seleccionar um subconjunto de n pontos adequados ao modelo. Sendo $E(N)$ é o valor esperado de N , $b = w^n$ e $a = (1 - b)$

$$E(N) = b + 2b \times (1 - b) + 3b \times (1 - b)^2 + \dots + ib \times (1 - b)^{i-1} \quad (5.28)$$

Ou seja,

$$E(N) = b \times [1 + 2a + 3a^2 + \dots + ia^{i-1} + \dots] \quad (5.29)$$

Considerando uma série geométrica do tipo, com razão r ,

$$\sum_{n=0}^{\infty} r^n = \frac{a}{(1 - a)} = a + a^2 + a^3 + \dots + a^i \quad (5.30)$$

e a sua derivada:

$$\frac{1}{(1 - a)^2} = 1 + 2a + 3a^2 + \dots + ia^{i-1} \quad (5.31)$$

a expressão para $E(N)$ pode ser então simplificada para

$$E(N) = \frac{1}{b} = w^{-n} \quad (5.32)$$

A expressão para o desvio padrão de N é dada por

$$SD(N) = \sqrt{E(N^2) - E(N)^2} \quad (5.33)$$

A expressão de $E(N^2)$ é dada por

$$E(N^2) = \sum_{i=0}^{\infty} (b * i^2 * a^{i-1}) \quad (5.34)$$

e pode ainda ser simplificada pela segunda derivada de a , obtendo

$$E(N^2) = \frac{(2-b)}{b^2} \quad (5.35)$$

e, conseqüentemente,

$$SD(N) = \sqrt{1 - w^n} \times \frac{1}{w^n} \quad (5.36)$$

Geralmente, o desvio padrão tem um valor aproximado ao valor esperado de N , o que significa que pode ser usado este valor para classificar um conjunto. ^[14]

iii. Limiar de distância

O limite t representa um ponto de partida para determinar se um dado subconjunto foi encontrado, permitindo o estabelecimento de um *conjunto consensus* que torne o algoritmo exequível. Este cálculo requer uma distribuição de probabilidade de uma distância de um *inlier* de um modelo.

Teoricamente, é assumido que o erro de medição é uma Gaussiana com média zero e desvio padrão σ , de modo a poder estipular um valor de limiar t . Na prática, este valor é escolhido empiricamente, após análise da tolerância de erro.

Num formalismo matemático, o quadrado da distância, d^2 , é a soma dos quadrados das gaussianas e segue uma distribuição do tipo χ_m^2 , com m graus de liberdade. A probabilidade de que o valor de uma variável de χ_m^2 seja menor que o quadrado de uma variável k é dada pela função cumulativa

$$F_m(k^2) = \int_0^{k^2} \chi_m^2(\xi) d\xi \quad (5.37)$$

Considerando então $t^2 = F_m^{-1}(\alpha)\sigma^2$, serão considerados *inliers* os pontos de valor inferior a t . Normalmente, α tem valores de 0.95, de forma que haja 95% de probabilidade de um ponto ser *inlier*.^[16]

A proporção dos pontos que não estão de acordo com o modelo – *outliers* – depende do quantidade e tipo de ruído presente e do número de modelos a analisar. A consideração de um modelo teórico depende, tal como na primeira variável, de uma tolerância de erro, e o modelo só é validado e tiver um número suficiente de pontos em concordância dentro dessa tolerância de erro.

Além destes parâmetros, considera-se também um outro, T , que determina o tamanho máximo do *conjunto consensus*, e expresso por $T = (1 - \epsilon) \cdot n$.

Caso o valor do conjunto seja superior a este parâmetro, o cálculo é terminado, dando início a outro conjunto.

5.4.3. *Optimização*

A velocidade do processamento do algoritmo depende maioritariamente de dois factores internos: primeiro, do número de amostras a ser construídas, dependente do nível de

contaminação destas, de modo a assegurar uma boa estimativa, e, em segundo, do tempo dispendido a avaliar a qualidade de cada modelo.

A ideia desta alternativa de modelo é implementar um algoritmo de avaliação dividido em duas partes: primeiro, é realizado um teste estatístico para avaliar a robustez dos pontos que compõem um conjunto inicial d , menor que n , constituído por pontos aleatórios. Numa recta final, essa avaliação é estendida ao conjunto final n , caso os pontos em d verifiquem correspondência. Então, apenas é necessário testar um pequeno número de pontos, pertencentes ao total, que seja muito menor que n , permitindo concluir, com um maior grau de confiança, se pertence ou não ao modelo. São assim eliminados, praticamente à priori, alguns modelos que não seriam aceites, evitando avaliar todos os pontos no seu conjunto, garantindo a natureza da solução ^[17].

O desempenho do algoritmo é condicionado também pelo aumento da amostra ou pelo número de modelos construídos a partir do mesmo conjunto de dados. Normalmente, os outliers encontram-se dispersos ao longo da amostra, enquanto que os *inliers* se concentram em nichos específicos. Esta constatação pode ser vantajosa e o algoritmo pode ser processado partindo de uma selecção de conjuntos baseados na relação espacial dos pontos; o primeiro ponto é escolhido aleatoriamente e os adjacentes a este são analisados imediatamente a seguir ^[15].

Além destas considerações, pode ter-se em conta a probabilidade de selecção de uma determinada característica, p_i , dada por

$$p_i = \frac{n_i}{N_m} \quad (5.38)$$

Assim, se cada conjunto for uniformemente amostrado, a probabilidade de uma correspondência candidata ser seleccionada é

$$\frac{n_i}{N_m} \times \frac{1}{n_i} = \frac{1}{N_m} \quad (5.39)$$

Impondo como base esta estratégia, ter-se-á a garantia que, num conjunto de amostras de todas as possíveis correspondências, nenhuma característica será repetida na amostra ^[13].

Referências Bibliográficas

- [1] Magnus, Jan R. . Matrix differential Calculus with applications in statistics and econometrics. 3rd. JOHN WILEY & SONS, 2007.
- [2] Descoteaux, Maxime. "A Multi-Scale Geometric Flow for Segmenting Vasculature in MRI: Theory and Validation." 3 May 2004 1-28. 13 April 2007 <[www-sop.inria.fr/odyssee/team/Maxime.Descoteaux/docs/cvamia_2004.pdf](http://www.sop.inria.fr/odyssee/team/Maxime.Descoteaux/docs/cvamia_2004.pdf)>.
- [3] Jiri, Hladuvka. "Exploiting Eigenvalues of the hessian matrix for volume decimation." 29 July 2007
<<http://citeseer.ist.psu.edu/cache/papers/cs/17934/ftp:zSzzSzfcp.tuwien.ac.atzSzpubzSzTRzSz00zSzTR-186-2-00-19Paper.pdf/exploiting-eigenvalues-of-the.pdf>>.
- [4] Aylward, Stephen R.. "Initialization, Noise, Singularities, and Scale in Height Ridge Traversal for Tubular Object Centerline Extraction." IEEE Transactions on medical imaging Vol 21 No 2 February 2002
<<http://citeseer.ist.psu.edu/cache/papers/cs/26235/http:zSzzSzwww.isi.uu.nlzSzMeetingszSz..zSzTGVzSzAylward.pdf/aylward02initialization.pdf>>.
- [5] Jahne, Bernd. Digital Image processing. 5th. Springer
- [6] Sutton, Michael A.. "Advances in two dimensional and Three-Dimensional Computer Vision." Springer-Verlag Berlin Heidelberg 2000 11 August 2007
- [7] Ertürk, Sarp. Digital Image Processing. Wiley-Interscience, 2003.
- [8] Forsyth, David A. . Computer Vision: A Modern Approach. 1st Ed. Prentice Hall, 2003.
- [9] Bovik, Al. Handbook of Image and Video Processing. 1st Ed. Canada: ACADEMIC PRESS, 2000.
- [10] Hartley, Richard , and Andrew Zisserman. Multiple View Geometry in Computer Vision. 2nd ed. Cambridge: Cambridge University Press, 2003.
- [11] Shapiro, Linda. Computer Vision. 1st ed. Prentice Hall, 2000.
- [12] Oliveira-Martins, Maria Teresa. Algebra Linear e Geometria Analítica. 1st. Coimbra: Vicentimagem, 2002.
- [13] Zhang, Wei. "Generalized RANSAC framework for relaxed correspondence problems." 1-7. <doi.ieeecomputersociety.org/10.1109/3DPVT.2006.67>.
- [14] Fischler, Martin. "Random Sample Consensus: a paradigm for model fitting with applications to image analysis and automated cartography." Communications of the ACM vol 24 No 6 June 1981 381-395 19 March 2007 <www.ai.sri.com/pubs/files/836.pdf>.
- [15] Cantzler, H.. "Random Sample Consensus (RANSAC)." 1-4. 19 March 2007 <homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/CANTZLER2/ransac.pdf>.

[16] Hartley, Richard , and Andrew Zisserman. Multiple View Geometry in Computer Vision. 2nd ed. Cambridge: Cambridge University Press, 2003.

[17] Chum, O.. "Randomized RANSAC with tdd test." BMVC 2002 448 - 457. 19 March 2007 <http://www.bmva.ac.uk/bmvc/2002/papers/50/full_50.pdf>.

6. Descrição do software

Numa primeira fase do projecto, o algoritmo foi desenvolvido para uma sequência artificial de imagens, criada a partir de uma retinografia. Foi determinado, para cada *frame*, o conjunto de pontos característicos da imagem. A partir daqui, foram determinadas as correlações entre os pontos de duas *frames* consecutivas, para estabelecimento de correspondências e posterior estimativa da transformação existente.

É definida, a partir de alguns dados inseridos pelo utilizador, uma distância máxima, que diz respeito ao movimento máximo esperado entre imagens. Este factor é calculado através do estabelecimento de relações entre valores de diâmetro de retina e disco óptico e resoluções da imagem e do conhecimento prévio da amplitude de movimentos sacádicos.

A função de correlação e o algoritmo de RANSAC são adaptações de funções já existentes e criadas por Peter Kovesi, da *University of Western Australia – School of Science & software engineering*. A biblioteca de funções das quais fazem parte os algoritmos anteriores está disponível *online*, em <http://www.csse.uwa.edu.au/>.

6.1. Determinação da amplitude máxima de movimentos entre *frames*

Com base numa retinografia de trinta e cinco graus adquirida por um sistema TOPCON, foi determinado o diâmetro da retina, cujo valor é de 637 píxeis. Por sua vez, o diâmetro do disco óptico, na mesma retinografia, é de 96 píxeis. Uma simples equação permite-nos obter o diâmetro, em graus, do disco óptico:

$$x_1 = \frac{96 \times 35}{637} \quad (6.1)$$

x_1 tem um valor aproximado de 5,27°. Este valor corresponde, então, ao diâmetro do disco óptico.

Assumindo uma amplitude de movimentos de 600 graus por segundo para os movimentos sacádicos do olho, e admitindo um número de *frames* por segundo escolhido pelo utilizador, a amplitude, em graus, de cada *frame*, é dada por

$$x_2 = 600 / \textit{numero de frames} \quad (6.2)$$

x_2 assume um valor de 600° dividido pelo número de *frames*.

Sabendo que o disco óptico tem um diâmetro de $1500\mu m$, ou seja, $1500 \times 10^{-6} m$,

$$x_3 = \frac{600 \times 1500 \times 10^{-6}}{5,27 \times \textit{frames}} \quad (6.3)$$

x_3 terá um valor de $0,17 / \textit{frames}$.

Convertendo este valor para polegadas (uma polegada corresponde a 0,0254 metros),

$$x_4 = \frac{0,17}{0,0254 \times \textit{frames}} \quad (6.4)$$

Através do valor de dpi, fornecido pelo utilizador, é possível converter o valor da amplitude de movimento, expresso em polegadas, para píxeis, conhecendo então o valor do deslocamento máximo entre duas *frames* consecutivas.

$$6,72 \times \frac{\textit{valor dpi}}{\textit{frames}} \quad (6.5)$$

Este deslocamento máximo será, então, dependente do número de *frames* e do valor de dpi, multiplicado por um factor de 6,72.

6.2. Sequência artificial de vídeo

Numa primeira fase do projecto, o software foi desenvolvido para um vídeo criado artificialmente. Ou seja, a partir de uma retinografia, desenvolveu-se uma sequência temporal de movimentos aleatórios, em 20 *frames*, e foi com base nessa sequência que o software foi desenvolvido.

Inicialmente, é pedido ao utilizador que assinale um ponto na retinografia e é estabelecido o tamanho de cada imagem da sequência (200 píxeis de altura e 80 píxeis de largura). Para cada *frame* são estabelecidos deslocamentos aleatórios, que serão guardados numa matriz, para posterior comparação. Designando as coordenadas do ponto assinalado pelo utilizador como coordenadas centrais da imagem, é actualizado o centro de cada imagem posterior, por simples adição dos movimentos aleatórios ao ponto central. É seleccionada a secção da imagem original correspondente às dimensões pretendidas, de acordo com a posição do centro, e, a partir desta, são calculados os pontos característicos e a transformação correspondente. O desenvolvimento do algoritmo é análogo ao da sequência real, descrito em seguida.

6.3. Sequência real de vídeo

Após a implementação do algoritmo desenvolvido na sequência artificial, este foi melhorado para a aplicação em vídeos reais e encontra-se descrito em seguida.

6.3.1. Considerações iniciais

Este software está preparado para que o utilizador possa definir o vídeo no qual pretende aplicar o *tracking* e o nome com que pretende guardar o resultado final da aplicação. São definidos pelo utilizador os parâmetros de entrada o número de pontos por polegada (dpi) e o número de *frames* por segundo (fps) com que o vídeo é apresentado. O primeiro surge da necessidade de estabelecer um valor para a distância máxima de movimentos aceitáveis entre *frames* consecutivas. O segundo parâmetro foi adicionado após se constatar que o número de *frames* por segundo pode variar. Apesar de este parâmetro ter sido sempre de 24 *frames* por segundo ao longo do desenvolvimento prático neste projecto, pode variar ao usar outros vídeos, pelo que o utilizador deve ter esta opção em atenção.

O programa selecciona o vídeo, em formato “.avi” definido “1ê” (ou seja, processa a informação de cada *frame*) as *frames* que constam no vídeo, para posterior aplicação das funções. Posteriormente, cria a variável na qual será guardado, *frame a frame*, o vídeo final, resultante do processo de detecção do movimento.

Após as definições iniciais, é então iniciado o processamento do vídeo. Cada imagem contida nas *frames* do vídeo original é transformada para escala de cinzentos, para que possa ser determinado o seu gradiente. Porém, para que o tempo necessário para a detecção dos pontos característicos seja reduzido, a imagem que será processada será limitada à região de interesse. Por intermédio de uma função intrínseca do Matlab – *regionprops* –, é determinado um rectângulo cujas fronteiras definem uma região de interesse da imagem, ou seja, uma área constituída maioritariamente por píxeis não-nulos. As coordenadas desta fronteira são dadas pelos valores de uma *bounding box* – característica integrante da função *regionprops*. A partir destas coordenadas é então definida uma nova imagem, cujos limites correspondem às fronteiras da *bounding box*, e que será sujeita ao processo de estabelecimento de correspondências.

No entanto, provavelmente devido à forma de aquisição do vídeo são detectadas 2 *bounding box*, em que uma delas corresponde à imagem na sua totalidade. Esta constatação leva a supor a existência de píxeis de valor não nulo na margem da imagem original. Uma vez que estes não possuem qualquer interesse para a detecção, é-lhes atribuído um valor nulo, eliminando assim a *bounding box* correspondente.

É, igualmente, de ter em atenção que a aquisição do vídeo pode não ter as condições ideais, isto é, pode conter *frames* que não tenham imagem, e toda a região é constituída por píxeis de valor nulo. Nestas condições, não é detectada nenhuma região de interesse. Foi então estipulada uma área mínima de 25 píxeis quadrados (um quadrado de 5×5 píxeis) para que seja considerada uma região de interesse para processamento; as imagens de que não respeitem este tamanho mínimo não são processadas, e o algoritmo passa imediatamente à *frame* seguinte do vídeo original.

6.3.2. Detecção dos pontos característicos da imagem

Para a detecção dos pontos, é criada uma máscara aplicada à imagem. Apesar de esta já estar limitada a uma região de interesse, a máscara assegura que a procura destes pontos se limita efectivamente à imagem de píxeis não nulos e garante que os pontos de interesse detectados não estarão nas fronteiras da imagem, ou seja, nas zonas onde há transição entre a imagem e o plano de fundo. Este processo vem, então, melhorar o tempo de processamento.

Dá-se, por vezes, o caso de a imagem não conter regiões de interesse significativo, a nível clínico. Este tipo de imagens não possui quaisquer características anatómicas (vasos

sanguíneos, por exemplo); são apenas manchas difusas sem qualquer significado para a detecção. Caso fossem detectados pontos nestas regiões, estes poderiam ser facilmente confundidos entre si e levariam a um falso *tracking*. Matematicamente, estas regiões são caracterizadas por ter um desvio padrão baixo, num intervalo de valores mais restrito, comparativamente às verdadeiras regiões de interesse. Nestas últimas, o desvio padrão assume valores muito distintos e pertencentes a um intervalo mais vasto. Esta constatação levou a concluir que o cálculo do desvio padrão da imagem poderia constituir um factor de distinção entre a relevância das imagens. Estabeleceu-se, então, um valor mínimo de desvio padrão de quinze píxeis. Este valor foi determinado empiricamente, por análise e comparação de imagens. Assim, as *frames* cuja área de interesse tiver um desvio padrão inferior ao estipulado são eliminadas do processo de detecção de pontos de interesse e o algoritmo passa automaticamente à *frame* seguinte do vídeo original.

Caso as *frames* contenham então uma área de interesse com características significativas, é iniciado o processo de detecção dos pontos característicos. Conforme já referido, o cálculo dos valores próprios da matriz Hessiana permite uma detecção destes pontos. No entanto, como a imagem é considerada uma função de valores discretos, é calculada uma aproximação às derivadas, pelo cálculo do gradiente e do gradiente do gradiente, nas direcções x e y . A partir dos valores obtidos, é construída, para cada píxel da imagem delimitada pela máscara, a matriz Hessiana. Cada uma destas matrizes terá, à partida, dois valores próprios; para cada uma, é seleccionado o máximo destes valores. Nesta fase do processo, tem-se então, uma matriz de dados constituída pelas coordenadas dos píxeis e o respectivo valor próprio máximo. Estes valores são ordenados por ordem crescente e são determinados os dois por cento mais elevados, conduzindo à selecção dos píxeis de relevância na imagem. Ou seja, se tiver, por exemplo, mil valores próprios ordenados por ordem crescente (calculados a partir de mil píxeis), são seleccionados e guardados numa matriz de dados os vinte píxeis que possuem os vinte valores próprios mais elevados. Esta última matriz é constituída pelas coordenadas dos píxeis e pelo valor do valor próprio máximo.

6.3.3. Correspondência de pontos entre duas frames consecutivas

Após o processo de detecção dos pontos característicos, é efectuada uma renomeação das matrizes obtidas até esta fase. Este processo servirá para estabelecer uma distinção entre os dados que se referem a cada imagem, de forma a estabelecer uma ordem de comparação

entre *frames*. As variáveis que definem a primeira *frame* processada são guardadas e são determinados os pontos característicos da *frame* seguinte. Os valores desta última são guardados com nomes diferentes das da primeira *frame*, assegurando assim a comparação entre as matrizes de pontos para possíveis estabelecimento de correspondências. Estas variáveis vão sendo actualizadas no decorrer do processamento das *frames* do vídeo, estabelecendo sempre uma *frame* anterior e uma *frame* actual e tornando possível a análise de correspondências entre duas *frames* consecutivas.

O processo de correlação é então iniciado, estipulando a distância máxima do movimento permitida, definida inicialmente pelo utilizador. Além deste parâmetro, é também definido um raio máximo para a janela de correlação, isto é, o tamanho da janela à volta da qual a correlação é estabelecida. Empiricamente, escolheu-se o valor de 11. As imagens, definidas como parâmetros de entrada para a função de correspondência, são sujeitas a um filtro de média, para homogeneizar a luminosidade entre as imagens, acelerando o processo de comparação (uma vez que assegura que as vizinhanças dos píxeis a comparar estão dentro dos mesmos níveis de luminosidade). É então preparada a matriz de correlação dos pontos. Para isso, o programa certifica-se dos tamanhos das matrizes de pontos, assegurando-se que estes são os correctos e são seleccionados os conjuntos de pontos que podem ser analisados para estabelecimento de correlação, isto é, aqueles que distam das margens um valor mínimo do raio da janela de correlação. Este processo eliminará, à partida, alguns pontos da correspondência. Para cada um destes pontos na primeira imagem, é definida uma janela de correlação normalizada e seleccionados os pontos da segunda imagem que se encontram dentro dos limites definidos pela distância máxima de movimento. Ou seja, para cada índice de linha e coluna de cada ponto da primeira imagem, é definida a janela em redor do píxel e são guardados numa matriz todos os índices dos possíveis pontos correspondentes na segunda imagem, dentro de um limite de distância de movimento. Em seguida, para cada um destes pontos da segunda imagem, é criada a janela de correlação, que será comparada com a da primeira imagem. É avaliado o valor de correlação entre ambas, que é guardado numa matriz de correlação. Este processo é repetido para todos os pontos da primeira imagem. No final, a matriz de correlação será constituída nas linhas pelos índices dos píxeis da primeira imagem, e nas colunas pelos índices dos píxeis da segunda imagem. Os seus elementos serão os valores resultantes do cálculo da correlação.

A partir desta matriz, serão seleccionadas as correspondências assumidas como verdadeiras. Para isso, são seleccionados os valores máximos de cada linha e de cada coluna da matriz de

correlação e verificada a sua consistência. Isto é, se os valores máximos da linha e da coluna se encontrarem na mesma posição, há uma correspondência em ambas as direcções e, logicamente, entre píxeis. O resultado final deste processo é uma correspondência entre os pontos de duas imagens consecutivas, obtendo as matrizes dos índices dos pontos correspondidos entre si.

6.3.4. Implementação do algoritmo de RANSAC

Para obter uma transformação robusta entre imagens, foi aplicado um algoritmo de eliminação de erros e cálculo simultâneo da transformação ocorrida entre imagens – RANSAC. Pondo de parte uma possível homografia entre imagens (uma vez que não faria sentido a consideração do factor de profundidade da imagem), resta a hipótese do cálculo de uma transformação *linear conformal*, a mais simples considerada pelo Matlab, em que são ponderados factores de escala, rotação e translação, sendo necessários dois pares de pontos correspondentes. Cada um destes parâmetros será avaliado para a transformação entre duas imagens consecutivas. Há ainda a ter em conta que o efeito de rotação e escala serão, à partida, nulos (uma vez que não há rotação do globo e a escala não varia). Assim, esta matriz de transformação será a fonte de informação acerca do deslocamento sofrido entre pontos, com a grande vantagem de ser actualizada à medida que as falsas correspondências vão sendo eliminadas.

Para inicialização do algoritmo de RANSAC, são definidos como parâmetros de entrada um valor de limiar t e as matrizes das coordenadas dos pontos correspondentes. O valor de limiar representa a distância limite entre os pontos a ajustar ao modelo da transformação, isto é, a distância máxima para ser considerado como *inlier*. Como é óbvio, este valor deve ser muito baixo, a variar entre 0,001 e 0,01, para que a transformação seja o mais fiel possível. O valor assumido para t será então 0,001. as matrizes de pontos terão um tamanho $3 \times n$, em que n representa o número de pontos considerados correspondentes, as duas primeiras linhas referem-se às coordenadas dos pontos e a terceira é adicionada para permitir a multiplicação de matrizes no processo decorrente, uma vez que a matriz de transformação é de tamanho 3×3 , como já visto anteriormente.

O tamanho das matrizes é então verificado na aplicação de software, certificando também, que haverá um número mínimo de pontos – dois – para estabelecer a transformação.

Na estimativa da transformação, especificamente, α , β , γ e δ são dados como parâmetros de entrada o modelo da transformação pretendida, uma função de distância, a degenerescência entre pontos, uma matriz construída pelos conjuntos de pontos correspondentes, o valor mínimo de pontos necessários para calcular a transformação e o valor de limiar t . O modelo da transformação é construído a partir da simples execução da função intrínseca do Matlab que permite o seu cálculo, através dos pontos correspondentes; a função de distância determina as matrizes necessárias para o cálculo de transferência de pontos, isto é, constrói duas matrizes que consideram a transformação linear nos pontos estabelecidos como correspondentes. Esta função vem permitir a avaliação da distância euclidiana e a selecção dos pontos que se encontram dentro do valor limite t . A degenerescência dos pontos serve apenas para certificação da validade dos pontos (não-nulos).

O algoritmo de RANSAC já descrito teoricamente é iniciado pela estipulação da probabilidade de escolha de uma amostra livre de outliers – 0,99 –, do número máximo de tentativas – 1000 – e do número máximo de tentativas de selecção de um conjunto correcto de pontos – 100.

O conjunto de pontos aleatório é seleccionado por avaliação do tamanho da população, isto é, pelo tamanho do conjunto de pontos correspondentes, e pelo tamanho mínimo de pontos para estabelecer a transformação (dois pontos). É então verificada a consistência da população e seleccionado um conjunto aleatório.

Este conjunto é avaliado para degenerescência, ou seja, os pontos que o constituem são analisados para verificar que não são nulos. É igualmente avaliada a distância entre os pontos, para eliminação de *outliers* e tornar o modelo mais robusto. No final deste procedimento, determinou-se um modelo da transformação e o respectivo número de *inliers* que o constitui.

Esta prática de selecção de pontos e verificação de modelo é repetida até que N , dado pela expressão $N = \frac{\log(1-p)}{\log(outliers)}$, seja inferior à unidade. Isto só é possível porque este valor é actualizado no final de cada estimativa. O modelo escolhido será aquele cujo número de *inliers* seja maior.

Após ser então encontrado o modelo adequado, a matriz final de transformação será ajustada para os índices dos pontos considerado *inliers* e os elementos (1,3) e (2,3) desta serão representativos do deslocamento em x e y , respectivamente.

O algoritmo guarda, então, numa matriz, estes valores de deslocamento, ao mesmo tempo que os apresenta na janela de comandos do Matlab.

Adicionalmente, é criada uma matriz que avalia o resultado do deslocamento. Se este tiver um valor inferior à distância máxima, calculada no início do algoritmo, o valor correspondente na matriz de resultado é 1; caso o deslocamento seja superior, é atribuído o valor 0 na matriz de resultados.

6.4. Ficheiros do programa

O programa desenvolvido é, então, constituído pelos seguintes ficheiros:

- `Trackig_point.m`: este é o ficheiro a partir do qual decorre todo o processo de detecção de movimento da imagem.
- `Pointdetection.m`: este ficheiro corresponde à função de detecção de pontos característicos da imagem. Avalia o desvio padrão da imagem e calcula a matriz Hessiana e os respectivos valores próprios, para todos os píxeis que constituem a imagem.
- `Matchbycorrelation.m`: este ficheiro é a função de correspondência de pontos, na qual é calculada a correlação entre os pontos característicos de duas imagens consecutivas.
- `Ransacfitlinearconformal.m`: este ficheiro permite a implementação da função RANSAC e a estimativa da transformação entre duas imagens. É nele que são implementadas as funções RANSAC e `linearconformalmatrix`.
- `Ransac.m`: este ficheiro constitui a função de aplicação do algoritmo de RANSAC.
- `Randsample.m`: este algoritmo permite a determinação dos conjuntos aleatórios de pontos, nos quais a criação do modelo de transformação é baseada.
- `Linearconformalmatrix.m`: este ficheiro calcula a matriz de transformação entre dois conjuntos de pontos correspondentes.

Estes ficheiros podem ser encontrados no CD de anexo a este relatório.

7. Análise de resultados

7.1. Sequência artificial de imagens

Na primeira fase do projecto, o software foi desenvolvido e implementado numa sequência artificial de 10 *frames*. Comparando os valores de deslocamento induzido na imagem – Movimentos Provocados – com o deslocamento calculado a partir das transformações entre imagens consecutivas – Movimento estimado –, obteve-se os seguintes resultados.

Tabela 7.1. Resultados e comparação dos movimentos para uma sequência de vídeo artificial

Movimentos Provocados		Movimento estimado		Resultado
0	0	0	0	0
-7	9	-7	9	1
2	-7	2	-7	1
10	1	10	1	1
5	0	5	-7,11E-15	1
7	3	7	3	1
-6	8	-6	8	1
7	-5	7	-5	1
-3	-2	-3	-2	1
-3	0	-3	-1,72E-14	1

Verifica-se uma uniformidade de valores entre o movimento induzido e o deslocamento calculado, o que leva a concluir que o algoritmo é eficaz na detecção de movimento entre imagens. Além disso, a última coluna avalia o resultado do deslocamento: o valor 1 é atribuído caso os valores de deslocamento se encontrem abaixo do valor de deslocamento máximo. Como, nesta sequência, este valor foi de 20 píxeis, o resultado foi considerado válido.

O tempo dispendido para cada operação encontra-se discriminado no CD de anexo ao relatório, na pasta de resultados, com o título “Resultados Profiler da sequência artificial de vídeo”. Neste ficheiro encontram-se discriminados todos os tempos para cada operação executada no algoritmo.

7.2. Sequência real de vídeo

Numa segunda fase do projecto, o software desenvolvido para a sequência artificial de imagens foi adaptado a uma sequência em vídeo, previamente adquirida com uma câmara Sony 3CCD ExwaveHAD, acoplada a uma lâmpada de fenda TAKAGI S10-15 e a um *frame grabber* Coreco Imaging PC-RGB. Esta sequência foi sujeita a um processo de segmentação, realizado paralelamente noutra projecto, de forma a seleccionar a região de interesse de cada imagem. O vídeo resultante deste processo serve de entrada ao software desenvolvido. Os valores usados para o número de *frames* por segundo e para o número de dpi foram 24 e 72, respectivamente. Daqui resulta uma distância máxima de 20 píxeis entre *frames*.

O resultado desta aplicação às 10 primeiras *frames* processadas (para o vídeo 3) é apresentado na tabela seguinte. Note-se que o vídeo original que serviu de base a este processo tem 151 *frames* a processar. No CD de anexo a este relatório, encontra-se a tabela completa de deslocamentos obtidos.

Tabela 7.2. Resultados de deslocamento obtidos para as 10 primeiras *frames*

Displacements		Result
0	0	0
1	18	1
-2	-22	0
1	2	1
-1	-2	1
1	1,26E-14	1
-4,35E-14	3,10E-14	1
2,50E-15	-1	1
1	1	1
1	-1,59E-14	1

Na primeira *frame*, foi marcado um ponto inicial, de posição fixa. (escolhido numa bifurcação de vasos), ao qual foram adicionados os deslocamentos obtidos. Esta nova posição foi, sequencialmente, marcada nas *frames* seguintes e avaliada a sua correspondência com a marcada na primeira *frame*. Verificou-se que, de facto, o ponto continuava assinalado na mesma característica da imagem. As imagens seguintes representam, então, alguns destes deslocamentos.

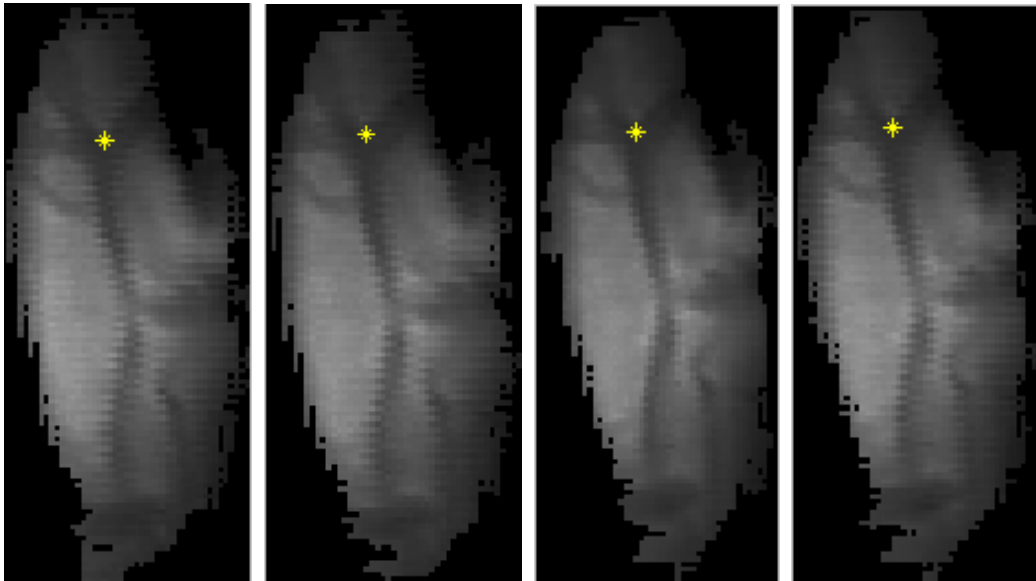


Figura 7.1. Representação dos deslocamentos obtidos para o vídeo 3.

Note-se que as imagens foram ampliadas (e a sua resolução alterada), o que pode dar a entender que o ponto não se encontra exactamente no mesmo sítio.

Analisando os dados relativos ao tempo de processamento de cada fase do processo, pode constatar-se que a implementação de software útil, isto é, a determinação simples do deslocamento (sem que haja apresentação de imagem ou criação de vídeo) demora cerca de 50 segundos para um vídeo de 151 *frames*, com uma região de interesse de 107×45 píxeis (vídeo 3). Já no caso de um vídeo de tamanho 365×85 , com 151 *frames* (vídeo 4), o processo integral de detecção, correspondência de pontos e determinação da transformação demora cerca de 290 segundos. Daqui se conclui que o processo é tanto mais rápido quanto menos for a área de interesse para a detecção de pontos característicos.

Este software foi desenvolvido tendo em conta que todas as *frames* do vídeo original são lidas na fase inicial do processo e só posteriormente é que são processadas uma a uma, para que se possa determinar a extensão do vídeo e o número máximo de *frames* às quais será aplicado o processo de detecção.

Em ambos os vídeos testados, verificou-se que os processos que exigem maior tempo de computação são a detecção de pontos e o cálculo das correspondências (no estabelecimento de correspondências entre *frames* consecutivas). Para o vídeo 3, os processos demoram cerca de 15 e 13 segundos, respectivamente. Isto deve-se a um tempo de cálculo de 11 segundos

para o cálculo da matriz de correspondências e de 3,8 segundos para o cálculo do máximo dos valores próprios da matriz Hessiana. Já no caso do vídeo 4, o tempo de cálculo da matriz de correlação demora cerca de 180 segundos e a detecção de pontos característicos demora cerca de 45 segundos.

No caso de se pretender um registo dos dados em formato vídeo, são necessárias funções adicionais, para criar um vídeo e gravar as *frames*. Este processo implica um tempo de computação consideravelmente longo, superior a 128 segundos (este valor corresponde ao tempo mínimo de aquisição de *frames* individuais).

No CD de anexo a este relatório, encontram-se os registos dos tempos dispendidos em cada uma das situações, para todas as funções que constituem o algoritmo.

8. Conclusões e trabalho futuro

Ao longo deste projecto, foram implementadas ferramentas matemáticas para a detecção do movimento sequencial de uma aquisição vídeo.

Entendendo uma imagem como uma função bidimensional, é possível calcular as suas derivadas direccionais, valores próprios, funções de correlação e, através dos resultados destes processos, estimar o deslocamento entre duas *frames* consecutivas.

O software desenvolvido está preparado para ser sensível a um deslocamento máximo, *frames* nulas (sem qualquer tipo de imagem ou sem características relevantes). O algoritmo determina o movimento, em cada direcção, de um conjunto de imagens integradas numa sequência vídeo.

Comparando os resultados obtidos para dois vídeos, verificou-se que o tempo de computação é proporcional à área útil de imagem. É, então, de todo o interesse do operador restringir ao máximo a região onde será aplicado o algoritmo e garantir boas condições de luminosidade (para que as características da imagem se encontrem em boas condições de detecção), para que o tempo dispendido no cálculo do movimento seja o mínimo possível. Além disso, quanto menos ruído e artefactos (como reflexos) a imagem tiver, mais rápido é o processamento.

O objectivo seria alcançar um tempo suficiente rápido para que a determinação do deslocamento pudesse ser conciliada com a aquisição vídeo. No entanto, era de conhecimento prévio que este objectivo dificilmente seria alcançado. Os métodos a aplicar exigiam, à partida, uma interpretação da imagem, que, por si só, consumiria demasiado tempo. No entanto, fazendo uma análise comparativa com outros os resultados obtidos em algoritmos anteriormente desenvolvidos (e sucintamente apresentados no capítulo 5), o tempo alcançado é considerado bastante satisfatório, tendo em conta a complexidade do processo.

Num desenvolvimento futuro do projecto, espera-se ser possível aumentar a velocidade do processo de cálculo do movimento. Isto irá permitir a conjugação da aquisição do vídeo com

o processo de segmentação e de detecção de movimento, de modo a poder interagir com um sistema de disparos laser e automatizar o processo.

A aplicação prática deste projecto vem apenas reforçar um novo estatuto na medicina, demonstrando todas as vantagens que uma conjugação entre diferentes ramos da ciência pode trazer. A versatilidade das ciências computacionais permite estabelecer facilmente uma nova aliança com a medicina, permitindo uma resposta mais rigorosa a uma realidade cada vez mais exigente a nível clínico. Particularmente, o desenvolvimento de um sistema laser que possa conjugar a detecção de movimentos oculares com a aquisição da imagem traz claros benefícios aos pacientes que dele possam usufruir. A prática cirúrgica, por exemplo, poderá ser muito mais rigorosa e exacta, permitindo a realização de cirurgias muito mais eficazes. Além disso, a monitorização de pacientes e o respectivo diagnóstico clínico poderão ser muito mais eficientes, alcançando resultados mais claros, uma vez que o próprio processo de avaliação da imagem permite detectar atempadamente as alterações anatómicas, e de uma forma mais rigorosa.

Será o alargamento de horizontes da medicina às restantes ciências, aliando a criatividade ao conhecimento, que vai marcar a diferença em cada novo desafio.

Anexo A. Relações básicas entre píxeis

1. Vizinhanças de um píxel

Uma vizinhança de quatro de um píxel de localização (x,y) considera os seus quatro vizinhos horizontais e verticais

$$(x+1, y), (x-1, y), (x, y+1), (x, y-1) \quad (8.1)$$

Uma vizinhança de oito considera, além desta vizinhança, os píxeis adjacentes nas diagonais [1]

$$(x+1, y+1), (x+1, y-1), (x-1, y-1), (x-1, y+1) \quad (8.2)$$

2. Escolher um filtro de suavização

O filtro pode ser escolhido tendo em conta um modelo de um *edge* e usando alguns critérios para obter a melhor resposta ao modelo. Esta abordagem torna-se complexa em duas dimensões, uma vez que os filtros podem adquirir curvatura. Por isso, o problema e a consequente escolha do filtro são formulados unidimensionalmente, e assumindo, posteriormente, uma simetria rotacional do filtro.

O modelo usual para definir o filtro é uma função escada $U(x)$, conjugada com ruído gaussiano estacionário, $n(x)$:

$$\begin{aligned} edge(x) &= AU(x) + n(x) \\ U(x) &= \begin{cases} 0, & x < 0 \\ 1, & x > 0 \end{cases} \end{aligned} \quad (8.3)$$

O valor A informa-nos do contraste do *edge* [2].

3. Enriquecimento de arestas por diferenciação de valores discretos

A diferenciação discreta é uma técnica de melhoramento de arestas locais. É usado para evidenciar arestas numa imagem, por diferenciação, quer na vertical ou na horizontal.

Matematicamente, para uma imagem original $a \in R^x$, a imagem \mathbf{b} melhorada pode ser obtida por:^[4]

Diferenciação horizontal

$$b(i, j) = a(i, j) - a(i, j+1) \quad (8.4)$$

ou

$$b(i, j) = 2a(i, j) - a(i, j-1) - a(i, j+1) \quad (8.5)$$

Esta diferenciação é obtida pela convolução

$$b := a \oplus r, \text{ onde } r_{(i,j)}(x) = \begin{cases} 1 & \text{se } x = (i, j) \\ -1 & \text{se } x = (i, j+1) \\ 0 & \text{outros} \end{cases} \quad (8.6)$$

ou, no segundo caso,

$$b := a \oplus s, \text{ onde } s_{(i,j)}(x) = \begin{cases} 2 & \text{se } x = (i, j) \\ -1 & \text{se } x \in \{(i, j-1), (i, j+1)\} \\ 0 & \text{outros} \end{cases} \quad (8.7)$$

Diferenciação vertical

$$b(i, j) = a(i, j) - a(i+1, j) \quad (8.8)$$

ou

$$b(i, j) = 2a(i, j) - a(i-1, j) - a(i+1, j) \quad (8.9)$$

Nestes casos, a convolução é dada por:

$$b := a \oplus t, \text{ em que } t_{(i,j)}(x) = \begin{cases} 1 & \text{se } x = (i, j) \\ -1 & \text{se } x = (i+1, j) \\ 0 & \text{outros} \end{cases} \quad (8.10)$$

ou

$$b := a \oplus u, \text{ em que } u_{(i,j)}(x) = \begin{cases} 2 & \text{se } x = (i, j) \\ -1 & \text{se } x \in \{(i-1, j), (i+1, j)\} \\ 0 & \text{outros} \end{cases} \quad (8.11)$$

Aproximação de gradiente

$$b(i, j) = |a(i, j) - a(i+1, j+1)| + |a(i, j) - a(i, j+1)| \quad (8.12)$$

ou

$$b(i, j) = |a(i, j) - a(i+1, j+1)| + |a(i-1, j) - a(i, j+1)| \quad (8.13)$$

Estabelecendo as convoluções:

$$b := |a \oplus t| + |a \oplus r| \quad (8.14)$$

e

$$b := |a \oplus v| + |a \oplus w| \quad (8.15)$$

os templates serão definidos como

$$v_{(i,j)}(x) = \begin{cases} 1 & \text{se } x = (i, j) \\ -1 & \text{se } x = (i+1, j+1) \\ 0 & \text{outros} \end{cases} \quad (8.16)$$

e

$$w_{(i,j)}(x) = \begin{cases} 1 & \text{se } x = (i, j) \\ -1 & \text{se } x = (i-1, j+1) \\ 0 & \text{outros} \end{cases} \quad (8.17)$$

4. Detecção direccional de arestas

Este método é baseado no uso de derivadas direccionais, identificando os píxeis que são possíveis arestas da imagem e designando n direcções para estes. A imagem é então convolucionada com $n/2$ máscaras, cada uma delas com 2 direcções associadas. A direcção do píxel será aquela cujo resultado tenha a maior magnitude após a convolução. A magnitude é dada pela expressão ^[4]

$$m(x) = \max_{0 \leq i \leq \frac{n}{2}-1} |a_i(x)| \quad (8.18)$$

Então, para $i \in \left(0, 1, \dots, \frac{n}{2}-1\right)$, ter-se-á, para uma imagem original $a \in R^X$:

$$a_i := (0, i) + f(a \oplus m_i) \quad (8.19)$$

onde

$$f(r) = \left(|r|, \frac{n}{2} \cdot \chi < 0(r) \right) \quad (8.20)$$

A resultante será dada por

$$b := \left(V \Big|_1 \right)_{i=0}^{\frac{n}{2}-1} a_i \quad (8.21)$$

Note-se que esta imagem contém em cada ponto a magnitude máxima de cada *edge* e a sua direcção associada ^[4].

5. Detecção hierárquica

Neste tipo de detecção, é desenvolvida uma estrutura piramidal, agrupando as vizinhanças de determinados píxeis num único, com valor igual à medição de algumas características locais, formando uma nova imagem, que terá uma resolução menor e menos píxeis. Este processo repete-se iterativamente, de forma a obter uma pirâmide hierárquica de imagens. Para detectar um *edge*, é aplicado um operador a uma imagem intermédia da pirâmide (de resolução intermédia). Se a magnitude da resposta exceder um determinado limiar, então a detecção processa-se numa imagem hierarquicamente superior, de maior resolução, e o operador é aplicado aos píxeis desse nível, e assim sucessivamente, até atingir o nível estabelecido pelo limiar. ^[4]

Ou seja, para a_k imagens numa pirâmide, $k \in \{0, 1, \dots, l-1\}$, com a imagem original $a_{l-1} \in \mathbb{R}^{X_{l-1}}$, a imagem $a_k \in \mathbb{R}^{X_k}$ é definida:

$$a_k(i, j) = \frac{1}{4} \cdot [a_{k+1}(2i, 2j) + a_{k+1}(2i+1, 2j) + a_{k+1}(2i, 2j+1) + a_{k+1}(2i+1, 2j+1)] \quad (8.22).$$

Tipicamente, o nível de detecção inicial é $k = \left\lceil \frac{2l}{3} \right\rceil$.

6. Detecção por *clusters* – *K-forms*

Esta técnica codifica a informação da diferença de intensidade local numa imagem. As suas componentes são números expressos na forma ternária ou decimal. Estas diferenças são calculadas e designadas com valores de 0, 1 ou 2, dependendo do seu valor do seu diferencial. A *K-form* é uma função linear desses valores, em que k determina o tamanho da vizinhança. Formalmente, numa vizinhança de 8 do píxel a_0

a_4	a_3	a_2
a_5	a_0	a_1
a_6	a_7	a_8

Para todos os píxeis da vizinhança, ou seja, para $l \in \{1, \dots, 8\}$ e para um valor de limiar T , é definido um $p: \mathbb{R} \rightarrow \{0, 1, 2\}$ tal que

$$T: p(r) = \begin{cases} 0 & \text{se } r < -T \\ 1 & \text{se } |r| \leq T \\ 2 & \text{se } r > T \end{cases} \quad (8.23)$$

A notação usada para definir a estrutura de k é f_b^{nc} , onde b representa a base (decimal ou terciária), n representa o tamanho da vizinhança, com uma orientação c (esta variável será substituída pela inicial da orientação vertical ou horizontal).^[4]

7. Fronteiras binárias

Um ponto na fronteira de um objecto numa imagem binária é definido como um ponto no qual a sua vizinhança de quatro (ou de oito) intersecta o objecto de que faz parte e o seu complementar. Estas fronteiras são classificadas pela sua conectividade e pela sua localização no objecto.

As transformações que envolvem uma binarização da imagem são, geralmente, métodos de *thinning*, que não preservam a homotopia (isto é, a deformação de uma aplicação entre espaços topológicos, que são conectados ou contínuos) da imagem e são normalmente aplicadas como complementos a outros processos de segmentação.

As técnicas empregues fazem uso dos conceitos de vizinhança, para aumentar ou reduzir a região (por reunião ou intersecção de conjuntos). Após esta operação, é intersectado com o seu complementar original, de modo a produzir a imagem binária^[4].

8. Detectores de gradiente

Estes operadores são aplicados ao longo das direcções principais do gradiente, num píxel específico e recorrem ao método das diferenças finitas. Nos detectores de Roberts e de Prewit assumem-se pesos iguais para os píxeis, enquanto que no de Sobel os píxeis mais próximos do centro têm maior importância.

É esperado que qualquer alteração significativa produza efeito num conjunto de píxeis. Por exemplo, o contorno de um objecto pode resultar numa cadeia de pontos, caso a derivada da imagem seja grande. Em alguns casos de ruído, estas derivadas são localizadas, o que significa que suavizar uma imagem tende a reunir e a servir de suporte a mudanças, suprimindo alguns dos efeitos indesejados da imagem.

O tempo para o cálculo da magnitude do gradiente é longo, o que motiva a procura de algumas aproximações, tais como o cálculo de $\max|\Delta x|, |\Delta y|$ ou de $|\Delta x| + |\Delta y|$. A primeira evita a computorização de raízes quadradas e potências, que levaria muito tempo de cálculo, quando se pretende um processo extremamente rápido ^[3].

9. Detector de Roberts

Este detector é mais um exemplo de um realce de características da imagem por técnicas de diferenciação.

Usando as mesmas notações para as imagens, a imagem final será definida pela convolução

$$b := \left((a \oplus s)^2 + (a \oplus t)^2 \right)^{1/2} \quad (8.24)$$

sendo os modelos

$$s_{(i,j)}(x) = \begin{cases} 1 & \text{se } x = (i, j) \\ -1 & \text{se } x = (i+1, j+1) \\ 0 & \text{outros} \end{cases} \quad (8.25)$$

e

$$t_{(i,j)}(x) = \begin{cases} 1 & \text{se } x = (i, j) \\ -1 & \text{se } x = (i+1, j-1) \\ 0 & \text{outros} \end{cases} \quad (8.26)$$

O resultado da imagem final é dado por ^[4]

$$b(i, j) = \left((a(i, j) - a(i+1, j+1))^2 + (a(i, j+1) - a(i+1, j))^2 \right)^{1/2} \quad (8.27)$$

10. Detector de Prewitt

Este detector calcula um vector de gradiente de um *edge* em cada ponto da imagem original, e o realce das arestas é alcançado partindo do cálculo da magnitude dos vectores.

A aproximação ao vector gradiente é feita pela convolução de duas máscaras com a imagem original, em que cada uma delas representa a derivada direccional em x ou em y . ^[4]

Algebricamente, ter-se-á os *templates* definidos por

$$s_{(i,j)}(x) = \begin{cases} -1 & \text{se } x = (i-1, j-1), (i-1, j), (i-1, j+1) \\ 1 & \text{se } x = (i+1, j-1), (i+1, j), (i+1, j+1) \\ 0 & \text{outros} \end{cases} \quad (8.28)$$

e

$$t_{(i,j)}(x) = \begin{cases} -1 & \text{se } x = (i-1, j-1), (i, j-1), (i+1, j-1) \\ 1 & \text{se } x = (i-1, j+1), (i, j+1), (i+1, j+1) \\ 0 & \text{outros} \end{cases} \quad (8.29)$$

A imagem final b será obtida pelas convoluções

$$b := \left(\left[(a \oplus s)^2 + (a \oplus t)^2 \right]^{1/2} \right) \quad (8.30)$$

e, direccionalmente por

$$d := \arctan 2 \left((a \oplus t) \Big|_{\text{domínio}(b \parallel_{>0})}, (a \oplus s) \Big|_{\text{domínio}(b \parallel_{>0})} \right) \quad (8.31)$$

11. Detector de Sobel

O Detector de Sobel pertence às técnicas de realce não lineares de diferenciação de valores discretos.

A imagem construída pela magnitude do gradiente é obtida pela convolução ^[4]

$$m := \left[(a \oplus s)^2 + (a \oplus t)^2 \right]^{1/2} \quad (8.32)$$

com os templates definidos por

$$s_{(i,j)}(x) = \begin{cases} -1 & \text{se } x = (i-1, j-1), (i-1, j+1) \\ 1 & \text{se } x = (i+1, j-1), (i+1, j+1) \\ -2 & \text{se } x = (i-1, j) \\ 2 & \text{se } x = (i+1, j) \\ 0 & \text{outros} \end{cases} \quad (8.33)$$

e

$$t_{(i,j)}(x) = \begin{cases} -1 & \text{se } x = (i-1, j-1), (i+1, j-1) \\ 1 & \text{se } x = (i-1, j+1), (i+1, j+1) \\ -2 & \text{se } x = (i, j-1) \\ 2 & \text{se } x = (i, j+1) \\ 0 & \text{outros} \end{cases} \quad (8.34)$$

A direcção é dada por

$$d := \arctan 2 \left((a \oplus s) \Big|_{b>0}, (a \oplus t) \Big|_{b>0} \right) \quad (8.35)$$

equivalente a

$$d(i, j) = \arctan\left(\frac{u}{v}\right) \quad (8.36)$$

Numa máscara generalizada 3×3 ,

a_3	a_2	a_1
a_4	(i,j)	a_0
a_5	a_6	a_7

O gradiente será dado por ^[4]

$$m(i, j) = (u^2 + v^2)^{1/2}, \text{ onde} \quad (8.37)$$

$$u = (a_5 + 2a_6 + a_7) - (a_1 + 2a_2 + a_3)$$

$$v = (2a_0 + a_1 + a_7) - (a_3 + 2a_4 + a_5)$$

12. Detector Laplaciano

O operador Laplaciano, expresso por

$$\Delta f = \nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \quad (8.38)$$

é também usado como detector de *edge* e pertence a uma classe de filtros isotrópicos, cuja resposta é independente da direção das discontinuidades da imagem – invariância à rotação.

Este operador é usado como base de avaliação de uma magnitude do gradiente: ter-se-á uma maior magnitude no ponto onde a segunda derivada seja nula e o gradiente elevado.

Numa aproximação para valores discretos, considera-se

$$\nabla^2 f(x, y) = -f[(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1)] + 4f(x, y) \quad (8.39)$$

Trata-se de um filtro linear, já que as derivadas são operações lineares. ^[1] A sua aplicação acentua as regiões de descontinuidade e esbate as regiões com pequenas variações. Esta aplicação conduz a imagens que têm as arestas destacadas num plano de fundo isento de características e escuro. As características que se encontram no plano de fundo podem ser recuperadas após este processo adicionando as imagens original e Laplaciana (imagem resultante da aplicação do operador Laplaciano). Assim,

$$g(x, y) = \begin{cases} f(x, y) - \nabla^2 f(x, y), & \text{caso o coeficiente no centro da máscara seja negativo} \\ f(x, y) + \nabla^2 f(x, y), & \text{caso o coeficiente no centro da máscara seja positivo} \end{cases}$$

A partir daqui, e por combinação da expressão para o Laplaciano, pode obter-se os coeficientes da máscara

$$\begin{aligned} g(x, y) &= f(x, y) - [f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1)] + 4f(x, y) \\ &\Leftrightarrow \\ g(x, y) &= 5f(x, y) - [f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1)] \end{aligned} \quad (8.40)$$

No entanto, ao ser aplicado o Laplaciano numa imagem com elevada quantidade de ruído, os seus valores serão mais elevados que as arestas e os resultados apresentam valores de arestas elevados, o que implica um processo de *thinning* ^[3].

13. Laplaciano do operador Gaussiano

Este operador foi proposto por Marr e Hildreth, para segmentação ou correspondência de imagens, recorrendo a um filtro gaussiano para eliminação de ruído.

Muito sucintamente, este detector processa-se com a convolução de uma imagem com uma função gaussiana bidimensional, e, seguidamente, calculando o Laplaciano da imagem resultante, em vez da habitual determinação por derivação parcial. As arestas são identificadas como sendo os zeros do Laplaciano. Além disso, este operador não requer um limite explícito, possibilitando ainda uma extracção de arestas numa gama alargada de frequências

espaciais, e a combinação de operadores em várias escalas, para uma detecção diferencial de características.

Matematicamente, a convolução em 2D dá-se por

$$I * G(i, j) = \sum_n \sum_m I(n, m)G(i - n, j - m) \quad (8.41)$$

Sendo a função gaussiana bidimensional definida por

$$G_\sigma(x, y) = \sigma^2 e^{-\frac{(x^2+y^2)}{\sigma^2}} \quad (8.42)$$

Neste caso, a ordem dos processos é indiferente, o que significa que pode ser calculado o Laplaciano da gaussiana e depois aplicar à imagem, como uma máscara de convolução [3], [5]:

$$\nabla^2 G_\sigma = \left(\frac{r^2 - 2\sigma^2}{\sigma^4} \right) e^{\left(\frac{-r}{2\sigma^2} \right)} \quad (8.43)$$

em que

$$r = \sqrt{x^2 + y^2} \quad (8.44)$$

14. Detector de Canny

Canny propôs um operador que é a soma de quatro exponenciais complexas, aproximadas à primeira derivada de uma Gaussiana. Para isso, partiu da suposição que a detecção é realizada por convolução da imagem com uma função $f(x)$ e as arestas detectadas correspondem ao valor máximo do resultado desta operação. São especificados três critérios para determinar até que ponto este algoritmo é eficaz na detecção. Assim, a operação deve ter uma baixa probabilidade de falha na detecção das arestas e de falsos positivos, combinada com a sua localização, em que as arestas marcadas deverão ser o mais próximo possível do centro dos verdadeiras. Estes parâmetros são dependentes da quantidade de ruído da imagem, o que implica que um melhor desempenho passa por maximizar a relação sinal – ruído, dada pela expressão

$$SNR = \frac{A \left| \int_{-w}^0 f(x) dx \right|}{n_0 \sqrt{\int_{-w}^w f^2(x) dx}} \quad (8.45)$$

Nesta condição, é calculado o inverso da distância entre o verdadeiro *edge* e o que é marcado pelo operador, recorrendo ao valor do desvio padrão na posição do operador. No entanto, Canny chegou à conclusão que há uma contradição entre localização e detecção, isto é, num valor mais elevado de relação sinal-ruído, a detecção é menos exacta. Para solucionar, recorre-se a um produto entre os dois critérios, de modo a poder combiná-los, chegando assim à solução óptima de $f(x)$.

Além das condições de localização e detecção, é essencial a garantia que há apenas uma resposta para cada *edge*.

Como acréscimo a estas considerações, é também implementada uma optimização estocástica para melhorar o detector. Este processo procura, de forma iterativa, os filtros possíveis para aplicação, criando um *edge* por adição de números Gaussianos aleatórios a uma amostra de valores de um step *edge*, e convolucionando cada filtro com este ponto. Por avaliação da resposta e alteração sucessiva dos parâmetros do filtro, este vai sendo optimizado para uma melhor resposta de detecção [3].

Assim, é assumido que o detector é uma convolução de um filtro f que suaviza o ruído e detecta o *edge*. A questão essencial desta teoria é determinar o filtro que optimiza a relação entre os três critérios: relação sinal ruído, localização e distância média entre os zeros. Estes podem ser expressos matematicamente, sendo f o filtro, como

$$\text{Localização: } L = \frac{A |f'(0)|}{n_0 \sqrt{\int_{-w}^w f'^2(x) dx}} \quad (8.46)$$

$$x_{zc} = \pi \left(\frac{\int_{-\infty}^{\infty} f'^2(x) dx}{\int_{-\infty}^{\infty} f''^2(x) dx} \right)^{\frac{1}{2}} \quad (8.47)$$

O valor da relação sinal-ruído deverá ser o maior possível, uma vez que é pretendido que o ruído, que se encontra no denominador, seja o menor possível. O valor da localização representa uma analogia entre a distância entre a localização detectada do *edge* e a sua verdadeira localização; por isso, este valor deverá ser pequeno, para assegurar uma maior autenticidade de resultados. A segunda equação, de $x_{\bar{x}}$, representa a distância média entre os zeros da derivada de f , o que permite assegurar que o mesmo *edge* não terá demasiadas respostas numa pequena região.

A implementação do algoritmo para encontrar um filtro f que maximize a relação sinal ruído será então semelhante a:

- 1) Criação de uma máscara unidimensional gaussiana convolucionada com a imagem I , na direcção das linhas e das colunas: I_x e I_y . (s) será o desvio padrão desta gaussiana;
- 2) Criação de uma máscara para a primeira derivada da gaussiana, nas direcções x e y : G_x e G_y ;
- 3) Convolução de I_x e I_y com G_x e G_y , respectivamente. Como resultados, ter-se-á I'_x e I'_y .

A magnitude será dada, em cada píxel, por

$$M(x, y) = \sqrt{I'_x(x, y)^2 + I'_y(x, y)^2} . \quad (8.48)$$

Após a convolução da imagem com o filtro, é estimada a direcção do *edge* a partir do gradiente da imagem filtrada e a magnitude desse gradiente passa pelo processo de supressão de valores não máximos. Os limiares necessários são estabelecidos baseados em estimativas do ruído da imagem ^[3].

Referências bibliográficas

- [1] Gonzalez, Rafael. Digital Image Processing Using Matlab. New Jersey: Prentice Hall, 2004.
- [2] Forsyth, David A. . Computer Vision: A Modern Approach. 1st Ed. Prentice Hall, 2003.
- [3] Mubarak, Shah. "Edge Detection." Fundamentals of Computer Vision. 1997. 6 April 2007 <<http://www.cs.ucf.edu/courses/cap6411/book.pdf>>.
- [4] Ritter, Gerhard X.. Handbook of Computer Vision Algorithms in Image Algebra. 1st Ed. CRC Press, 2001
- [5] Parker, J.. "Advanced Edge Detection Techniques: The Canny and the Shen-Castan Methods." Techniques in Computational Vision. 7 April 2007 <<http://pages.cpsc.ucalgary.ca/~parker/501/edgedetect.pdf>>.

Anexo B. Vectores e valores próprios

Diz-se que λ é um vector próprio de uma matriz $A \in M_{n \times n}$ se existir um vector não nulo tal que

$$Ax = \lambda x, x \neq 0 \quad (2.1)$$

Este vector denomina-se vector próprio associado ao valor próprio λ . A todo o vector próprio está associado um único valor próprio que verificam esta condição. A cada valor próprio λ está associado um conjunto de vectores, que constituem um subespaço designado subespaço próprio associado ao valor próprio λ de A . Este subespaço é representado por

$$V_\lambda = \{x \in \mathbb{R}^n : Ax = \lambda x\} = N(A - \lambda I) \quad (2.2)$$

em que I representa a matriz identidade (uma matriz cujos valores da diagonal são 1 e os restantes são nulos). A equação expressa no subespaço é equivalente ao sistema

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = \lambda x_1 \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = \lambda x_2 \\ \dots \\ a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n = \lambda x_n \end{cases} \quad (2.3)$$

Como o vector de x terá de ser não nulo, a solução será calculada pela equação

$$|A - \lambda I| = 0 \quad (2.4)$$

Esta equação equivale a

$$\begin{vmatrix} a_{11} - \lambda & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} - \lambda & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} - \lambda \end{vmatrix} = 0 \quad (2.5)$$

Esta equação denomina-se equação característica de A e o primeiro elemento designa-se polinómio característico de A , de grau n em λ .

Na aplicação a matrizes bidimensionais (como é o caso da matriz Hessiana), o cálculo do determinante

$$\begin{vmatrix} a_{11} - \lambda & a_{12} \\ a_{21} & a_{22} - \lambda \end{vmatrix} = 0 \quad (2.6)$$

é efectuado por

$$(a_{11} - \lambda)(a_{22} - \lambda) - a_{12}a_{21} = 0 \quad (2.7)$$

Referências Bibliográficas

Oliveira-Martins, Maria Teresa. Algebra Linear e Geometria Analítica. 1st. Coimbra:
Vicentimagem, 2002.

Anexo C: Software desenvolvido

O anexo C é apresentado em CD, como parte integrante deste relatório. Dele fazem parte o código desenvolvido em Matlab, com todas as funções necessárias ao seu funcionamento, e os resultados obtidos, tais como os vídeos resultantes do processo e os tempos obtidos (estes encontram-se em formato PDF). Além destes, o CD tem também os vídeos que serviram de base à implementação do algoritmo.