



Universidade de Coimbra
Faculdade de Ciências e Tecnologia

Construção de uma câmara de alta sensibilidade e baixa resolução para super resolução

Relatório de Projecto
Mestrado Integrado em Engenharia Biomédica

Narciso António Vaz Beça
Coimbra, 2008



Construção de uma câmara de alta sensibilidade e baixa resolução para super resolução

Projecto realizado no Instituto Biomédico de Investigação da Luz e Imagem (IBILI), da Faculdade de Medicina da Universidade de Coimbra, sob orientação do Prof. Doutor Nuno Ferreira e sob a supervisão do Engenheiro Francisco Caramelo, ambos docentes nesta instituição.

Narciso António Vaz Beça

Coimbra, 2008

Agradecimentos

Em primeiro lugar quero dirigir-me ao Professor Miguel Morgado, que com o seu ímpar bom senso me ajudou na orientação e escolha deste Projecto, revelando-se sem dúvida certo. Obrigado pela persistência em querer fazer sempre parte da solução e não do problema.

Muito agradeço ao Professor Nuno Ferreira e ao Professor Francisco Caramelo pela paciência, dedicação e compreensão. A sua orientação e coordenação aliadas ao profissionalismo e objectividade, tornaram este Projecto um desafio aliciante.

Um agradecimento especial ao Engenheiro Samuel Neves pelas 'dicas' na linguagem C/C++.

Ao Sr. Abreu pela disponibilidade da sua bancada de trabalhos.

Muito Obrigado aos colegas que mais directamente colaboraram para chegar até aqui.

Aos meus irmãos pela devoção.

Por último, queria agradecer à minha mãe pelo apoio incondicional desde sempre.

Índice

Agradecimentos	V
Índice	VI
Índice de figuras	VII
Resumo	10
Organização	11
1 – Introdução	12
1.1 - Objectivos	12
2 – Aspectos teóricos	14
2.1 - Conceitos	14
2.2 - Enquadramento histórico	22
2.3 - O Algoritmo testado	26
3 - Métodos	32
3.1 - Sensor	32
3.2 - Câmara	43
3.3 - Plotter	46
3.4 - Suporte	49
3.5 - Software	52
4 - Resultados	62
4.1 - Câmara	62
4.2 - Algoritmo	67
5 - Discussão e Conclusões	69
Referências bibliográficas	72
ANEXOS	76

Índice de figuras

Figura 1 - Resolução espacial (FWHM). Gaussianas correspondentes a funções de resposta do aparelho: os dois picos são distinguíveis quando a distância entre eles é maior do que a FWHM [22].	14
Figura 2 - Resolução espacial. Duas fontes pontuais distanciadas de 1cm.	15
Figura 3 - Variação de resolução espacial numa imagem digital da Lena com os mesmos níveis de cinzento (256). a) 256x256 pixels, b) 128x128 pixels, c) 64x64 pixels.	15
Figura 4 - Falsos contornos provocados pelo uso de um número insuficiente de níveis de cinzento: a) 128, b) 64, c) 32, d) 8, e) 4, f) 2. [21]	16
Figura 5 - Arrastamento provocado pelo deslocamento do sensor/câmara.	17
Figura 6 - Linha superior: imagens aumentadas de 128x128, 64x64 e 32x32 pixels para 1024x1024 pixels usando interpolação nearest neighbor. Linha inferior: a mesma sequência usando interpolação bilinear. (da esquerda para a direita). [21]	19
Figura 7 - Aumento de uma imagem (1) de 25x20 pixels para 50x20 (2), 24x19 (3) e 26x21 pixels (4) usando interpolação bicúbica. A perda de definição é menor quando se converte a imagem para múltiplos inteiros da resolução da imagem original.	20
Figura 8 - Imagem de alta resolução (HR) obtida pelo algoritmo de super resolução (SR) processando as imagens de baixa resolução (LR) adquiridas nas posições p_i . O número de imagens LR necessárias é de N^2 , para uma melhoria de resolução por um factor de N.	27
Figura 9 - Fórmula matemática para cálculo da matriz do sistema.	27
Figura 10 - Esquema de aquisição de imagens LR.	29
Figura 11 - Construção da matriz do sistema do algoritmo de super resolução (SR). Para cada posição p_j da imagem de baixa resolução (LR) o respectivo valor do pixel L_j , é repartido por todos os pixels S_i correspondentes na imagem de alta resolução (HR), vistos pelo pixel L_j . [11]	30
Figura 12 - Placa comum de um rato óptico (esquerda), Agilent 2610 (direita) 33	
Figura 13 - Diagrama mecânico ADNS 2610 - 8 pins (visto de cima)	34

Figura 14 - Operação de ‘escrita’ constituída por dois bytes. O primeiro contém a informação da direcção dos dados (MSB=1) e registo pretendido, o segundo contém os dados. Esta operação tem a direcção do PC para o sensor. [25]....	35
Figura 15 - Operação de ‘leitura’ constituída por dois bytes. O primeiro contém a informação da direcção dos dados (MSB=0) e registo pretendido, o segundo contém os dados. Esta operação tem a direcção do PC para o sensor no primeiro byte e do sensor para o PC no segundo. [25].....	36
Figura 16 - Chamada ao Registo <i>Pixel Data</i> para obtenção de imagem (pixel 1). Após efectuada a leitura 324 vezes obtém-se a imagem completa capturada pelo sensor.	37
Figura 17 - Grelha de disposição dos pixels no CCD olhando para a sua superfície.....	39
Figura 18 - Diagrama mecânico da porta paralela e configuração dos <i>pins</i>	40
Figura 19 - Plano de ligações e respectivas soldaduras. a) diagrama do sensor, b) sensor, c) porta paralela, d) ligações/soldaduras no sensor, e) ligações/soldaduras na porta paralela (a azul os pins do sensor), f) cabo USB utilizado para fonte de alimentação do sensor (pin 7).....	42
Figura 20 - Óptica aplicada ao sensor para utilização no rato óptico.	43
Figura 21 - Óptica sobre uma linha de 0,5mm de espessura. O campo de visão é muito pequeno, impedindo a focagem de grandes objectos.....	44
Figura 22 - <i>Webcams</i> utilizadas para retirar ópticas. a) Samsung Pleomax, b) Logitech QuickCam Messenger.	44
Figura 23 - Ópticas aplicadas aos sensores. a) Samsung, b) Logitech.....	45
Figura 24 - <i>Graphtec Plotter</i> MP3200.....	47
Figura 25 - Suporte para a câmara. a) peça principal, b) aplicação na <i>plotter</i> , c) peças constituintes do suporte.	49
Figura 26 - Suporte definitivo aplicado na <i>plotter</i>	51
Figura 27 - Imagens quadradas de calibração com respectivas dimensões.	51
Figura 28 - a) PC com duas portas paralelas, b) Placa PCI / porta paralela	53
Figura 29 - Interface gráfica (GUI) em Matlab para controlo do equipamento.	55
Figura 30 - GUI com a opção ‘Ligar’ e possíveis respostas.	56
Figura 31 - GUI com a opção ‘Funções’ e selecção de ‘Calibrar’.....	56
Figura 32 - Opção ‘Calibrar’ com visualização da imagem de calibração.....	57

Figura 33 - Opção 'Processar' com as escolhas de dimensão de imagem 5,4cm e aumento de resolução espacial por um factor de 4.....	58
Figura 34 - Imagem de teste utilizada nos testes do sistema.	59
Figura 35 - Sentido do movimento e capturas das imagens de sub pixel para um exemplo de aumento de resolução espacial por um factor de 4. 'Pixel' é a área coberta por um pixel do sensor, 'Sub Pixel' representa os movimentos efectuados pela <i>plotter</i> com distâncias segundo X e Y menores que a dimensão do pixel do sensor. Os números representam os índices dos pixels da imagem de HR.....	60
Figura 36 - Ferramenta de teste e equipamento de apoio.....	61
Figura 37 - Primeira imagem caracterizada capturada pela aplicação.....	63
Figura 38 - Imagem de uma folha de papel branca. Actualização dos valores dos pixels segundo linhas periódicas diagonais, provocadas pelo mecanismo <i>Shutter</i> . São também evidentes diferentes eficiências entre pixels.	63
Figura 39 - Captura de uma imagem (esquerda), após tratamento (direita).....	64
Figura 40 - Imagem resultante após tratamento sem normalização para a profundidade de cor 8 bit/pixel na escala de cinzentos (256 níveis).....	65
Figura 41 - Fantoma (esquerda) utilizado nos testes de aplicação do algoritmo de super-resolução e exemplo de uma imagem de baixa resolução adquirida pela câmara, numa única posição (direita). O quadrado sólido a negro do Fantoma tem de lado 7 mm. Cada pixel da imagem de baixa resolução tem a dimensão de 3 mm.	66
Figura 42 - Resultado da reconstrução de imagem num aumento de resolução 5x (versão IDL).....	67

Resumo

Com base no crescente interesse e investigação na área da super resolução, este trabalho ambiciona desenvolver um sistema que permita capturar imagens de baixa resolução para testar um algoritmo de super resolução. Técnica que se baseia na construção de uma única imagem de alta resolução a partir de uma série de imagens de baixa resolução.

Para tal, foi desenvolvido um mecanismo que executa diversas aquisições do mesmo objecto deslocadas entre si por distâncias inferiores às do tamanho do pixel de uma amostra de baixa resolução.

O mecanismo é constituído por uma câmara de baixa resolução, uma *plotter*, um suporte para a câmara e uma aplicação que integra e controla todo o sistema ligado a um computador pessoal.

A câmara foi construída com base num sensor de resolução espacial de 18x18 pixels constituente de um rato óptico e com a óptica de uma webcam.

Foi dimensionado e desenvolvido em ligas leves um suporte que tem como função fixar a câmara à cabeça da *plotter* e que permite colocar a câmara numa posição horizontal e ajustá-la verticalmente para diferentes focagens.

A *plotter* é responsável pelos pequenos movimentos nos quais se baseia a técnica e onde é colocada a imagem/objecto com a ajuda do seu tabuleiro.

A comunicação com a câmara e a *plotter* é efectuada através de duas portas paralelas instaladas no computador, uma para cada equipamento.

O Programa da aplicação que controla o sistema foi desenvolvido na linguagem C, do qual resultou uma (biblioteca dinâmica) DLL que serve uma interface gráfica desenvolvida no ambiente MATLAB.

Na aplicação é configurável a dimensão da imagem de teste e o aumento de resolução pretendidos.

Organização

No primeiro capítulo é feita uma breve introdução ao Projecto onde são salientados os objectivos a que se propõe e sua justificação, bem como, a motivação do próprio trabalho.

Alguns conceitos básicos teóricos sobre super resolução, aspectos históricos julgados relevantes para o enquadramento e a explicação do algoritmo de super resolução utilizado, têm lugar no capítulo dois.

O capítulo três aborda os métodos utilizados (i.e. a sua descrição e justificação), os equipamentos escolhidos e a forma como foram integrados, a descrição do software desenvolvido para o funcionamento de todo o sistema e, ainda, as contrariedades e dificuldades encontradas.

Os resultados obtidos em relação a todo o equipamento e, também, em relação ao próprio algoritmo de super resolução são apresentados no capítulo quatro.

O quinto e último capítulo é constituído pela discussão dos resultados obtidos e pelas conclusões.

1 – Introdução

1.1 - Objectivos

A super resolução é uma das áreas de crescente interesse e investigação actual, já que permite superar as limitações inerentes aos sistemas de imagem no que diz respeito à resolução espacial (ver secção 2.1) e aumentar o desempenho de variadas aplicações de processamento digital de imagem. O objectivo desse conjunto de técnicas de processamento é melhorar a resolução espacial e, com isso, melhorar a interpretação e a identificação dos pormenores nas imagens, preservando as informações originais sem acrescentar detalhes falsos (artefactos) à imagem de alta resolução.

A resolução espacial numa imagem é um importante factor no desempenho de diversos algoritmos de visão/processamento computacional. A detecção, reconhecimento e identificação de detalhes são procedimentos que, por isso, poderão ter os seus desempenhos melhorados com o uso da super resolução.

A análise de imagem que é efectuada, correntemente, em várias áreas do conhecimento tem um problema comum associado à utilização de imagens de baixa resolução (seja isto devido a limitações orçamentais ou mesmo técnicas), o que dificulta a visualização precisa de detalhes dos fenómenos patentes nas imagens.

Este trabalho pretende desenvolver um sistema que permita capturar imagens de baixa resolução (LR - “low resolution”) para testar o desempenho de um algoritmo de super resolução (SR - “super resolution”). Esta técnica baseia-se na construção de uma única imagem de alta resolução (HR - “High resolution) a partir de uma série de imagens LR.

O mecanismo tem de executar diversas aquisições do mesmo objecto, deslocadas entre si por distâncias inferiores às do tamanho do pixel de uma amostra LR. Esta técnica é, usualmente, designada de *substepping*, e o

processamento consiste na sobreposição das amostras e na determinação da contribuição de cada pixel para a imagem final.

Assim, o presente trabalho tem como objectivo principal o desenvolvimento de uma ferramenta de teste útil para validar algoritmos de super resolução baseados na técnica de *substepping*, oferecendo-nos uma utilização fácil e possibilitando a troca da imagem de teste desejada, com uma precisão específica e com um óptimo grau de repetibilidade, promovendo desta forma grande fiabilidade nos resultados. A mobilidade neste tipo de equipamento é também um requisito, pois o seu fácil transporte e integração constituem-se de relevante interesse. Para isso, a ferramenta deverá permitir fazer as aquisições de imagens de baixa resolução de um objecto sendo, por isso, fundamental que consiga deslocar a câmara em distâncias inferiores ao tamanho do pixel, ou deslocar o objecto em si.

2 – Aspectos teóricos

2.1 - Conceitos

São apresentados de seguida alguns conceitos básicos relativos à aquisição, tratamento e melhoramento de imagem, essenciais para uma melhor compreensão de todos os processos, métodos e conclusões que são abordados neste trabalho.

Comece-se, então, pela definição básica de resolução espacial. A resolução espacial pode ser definida como a distinção entre dois pontos muito próximos permitindo a sua identificação, isto é, capacidade de reproduzir detalhes finos.

A resolução espacial é definida como a largura a meia altura de uma gaussiana (FWHM - full width at half maximum) correspondente à função de resposta do aparelho, ver figura 1 e 2.

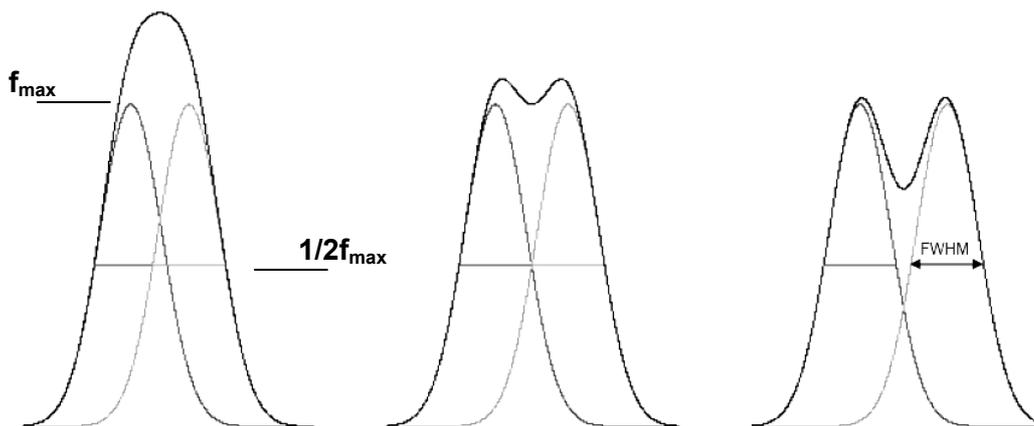


Figura 1 - Resolução espacial (FWHM). Gaussianas correspondentes a funções de resposta do aparelho: os dois picos são distinguíveis quando a distância entre eles é maior do que a FWHM [22].

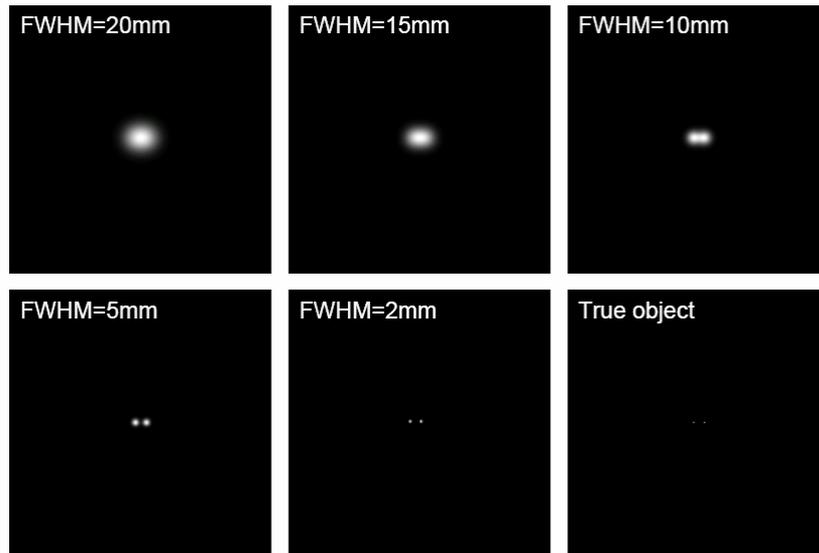


Figura 2 - Resolução espacial. Duas fontes pontuais distanciadas de 1cm. Resultados da resposta do aparelho para valores distintos de FWHM [22].



Figura 3 - Variação de resolução espacial numa imagem digital da Lena com os mesmos níveis de cinzento (256). a) 256x256 pixels, b) 128x128 pixels, c) 64x64 pixels.

A qualidade de uma imagem digital está associada às resoluções espacial e radiométrica. A resolução espacial está relacionada com o tamanho do pixel, o qual depende do número e dimensão dos elementos dos sensores, mas de facto não são a mesma coisa. O número e dimensão dos elementos do sensor são constantes, enquanto que, a resolução espacial pode ser alterada com a distância do sensor ao objecto. A resolução radiométrica consiste na quantificação da radiação electromagnética desde o espectro Ultravioleta até ao Infravermelho passando obviamente pelo Visível onde se destaca a Fotometria que se dedica à medição da luz visível tal como é perceptível ao

olho humano. Desta forma, esta resolução está aliada aos níveis de quantização empregues para amostrar a luminosidade reflectida pela cena, o que depende do número de bits usados para armazenar a informação de brilho (ou cor) de cada pixel. É importante distinguir entre elemento de resolução, que é o tamanho do pixel, e resolução espacial, que está relacionada com a dimensão dos menores detalhes visíveis na imagem, como definido acima.

Em particular a resolução de níveis de cinzento corresponde à menor variação de nível de cinzento que é possível distinguir numa imagem. Dois níveis de cinzento muito próximos conseguem distinguir-se se a sua diferença de nível de cinzento for maior do que a resolução [21].

Falsos contornos são visíveis em áreas de variação lenta de níveis de cinzento, devido ao uso de um número insuficiente de níveis de cinzento, ver figura 4.

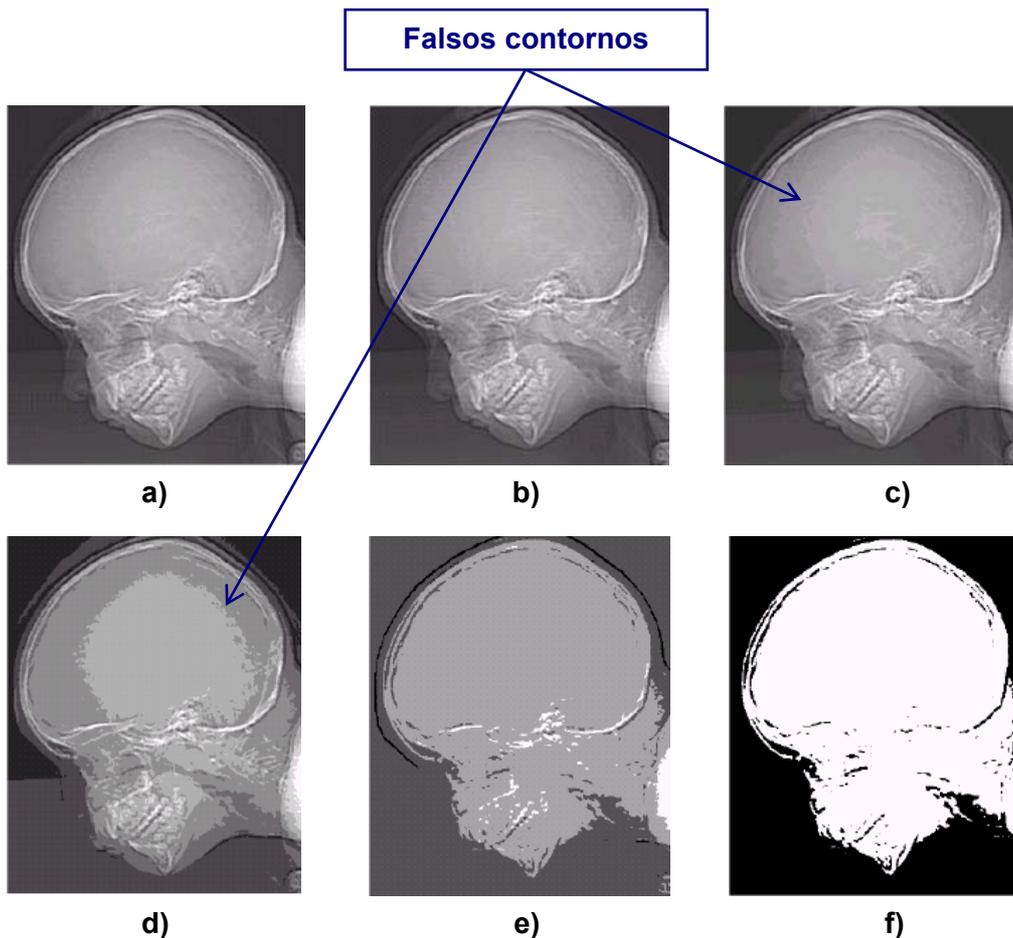


Figura 4 - Falsos contornos provocados pelo uso de um número insuficiente de níveis de cinzento: a) 128, b) 64, c) 32, d) 8, e) 4, f) 2. [21]

Existem alguns factores com bastante nível de ocorrência que promovem a degradação da qualidade de uma imagem digital.

São exemplos desses factores:

- Aberração das lentes da câmara, que provocam um esborratar (blurring) circular da imagem.
- Focagem incorrecta. Ocorre quando o sensor não está bem posicionado com o plano de focagem, atendendo aos conceitos teóricos da Óptica geométrica [27].
- Deslocamento do sensor/câmara ou do objecto durante a aquisição da imagem, causando o denominado *arrastamento*.
- Deslocamento de um objecto da cena, que causa um esborratar equivalente ao *arrastamento*, ver figura 5.



Figura 5 - Arrastamento provocado pelo deslocamento do sensor/câmara.

- Insuficiência ou excesso de iluminação.

Entre algumas áreas que podem requerer uma melhor resolução espacial, podem incluir-se a astronomia, geologia, monitorização ambiental, medicina e defesa.

Apesar de existirem sensores comerciais de alta resolução espacial, muitas vezes, o custo associado impossibilita a sua utilização na aquisição dessas imagens.

A SR tem provado ser bastante útil onde é possível obter diferentes aquisições de uma mesma imagem, incluindo imagens médicas, aplicações de vídeo e na Astronomia [10, 19, 26].

As técnicas de interpolação (vizinho mais próximo, bilinear e bicúbica) diferem da super resolução. Nas primeiras, apenas uma imagem é utilizada como fonte de informação para que uma imagem de maior resolução seja produzida, algo que não acontece utilizando SR para produzir uma imagem de maior resolução.

Interpolação é a estimação de um determinado parâmetro para o qual não há informação espacial disponível, num intervalo de dados conhecidos.

O método do vizinho mais próximo (*Nearest-neighbor*) selecciona simplesmente o valor do ponto mais próximo para o valor a estimar, sem ter em conta o valor de outros vizinhos.

Por outro lado, a interpolação bilinear é feita com base em vários vizinhos mais próximos em relação ao ponto a estimar, o que não deixa de ser uma aproximação polinomial de ordem 1, em que as diferentes distâncias desses vizinhos têm contribuições distintas (mais próximo, mais peso) [8].

Usando por exemplo os quatro vizinhos mais próximos de um ponto (x,y) , sendo (x',y') as coordenadas de um ponto na imagem ampliada e $v(x',y')$ o nível de cinzento a atribuir, então $v(x',y')=ax'+by'+cx'y'+d$, onde os quatro coeficientes são determinados a partir das quatro equações com quatro incógnitas que se podem escrever usando os quatro vizinhos do ponto (x,y) .

A interpolação bicúbica embora polinomial, insere-se num tipo de interpolação designada *spline*, que determina a estimação dos valores usando polinómios de baixa ordem em cada intervalo e escolhe os termos desses polinómios que melhor se ajustam aos pesos para obter um resultado mais suave. Comparativamente à interpolação polinomial típica, a *spline* introduz um

erro menor e é muito mais fácil a sua implementação quando na primeira são utilizados polinómios de ordem maior [9, 28].

Independentemente da sua complexidade, as interpolações não deixam de ser aproximações: a interpolação bilinear utiliza os 4 vizinhos mais próximos para estimar o valor, enquanto que a bicúbica usa os 16 mais próximos.

São apresentados em seguida alguns exemplos práticos da aplicação dos tipos de interpolação descritos (figuras 6 e 7).

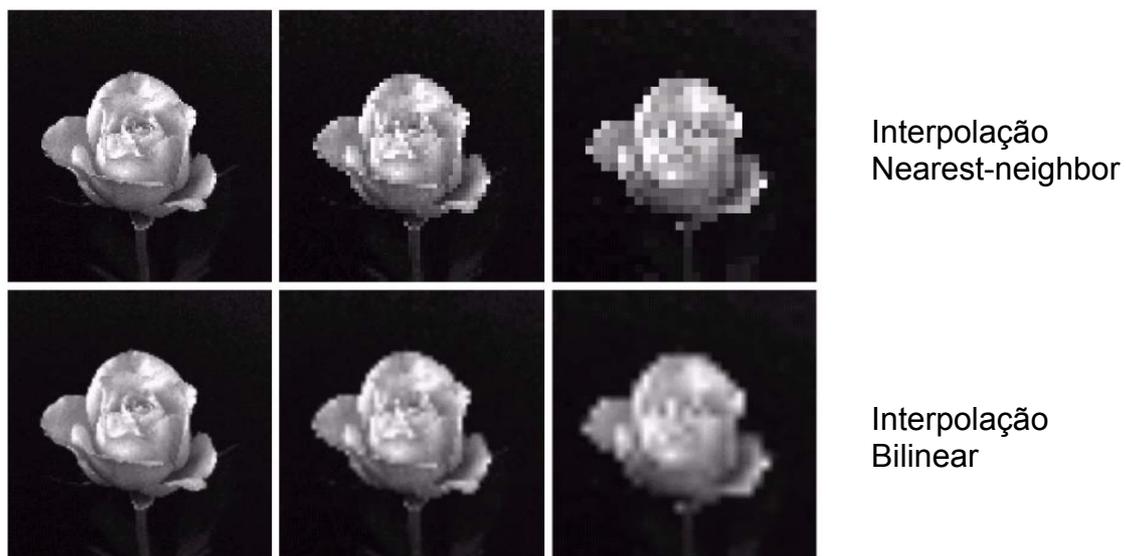


Figura 6 - Linha superior: imagens aumentadas de 128x128, 64x64 e 32x32 pixels para 1024x1024 pixels usando interpolação nearest neighbor. Linha inferior: a mesma sequência usando interpolação bilinear. (da esquerda para a direita). [21]

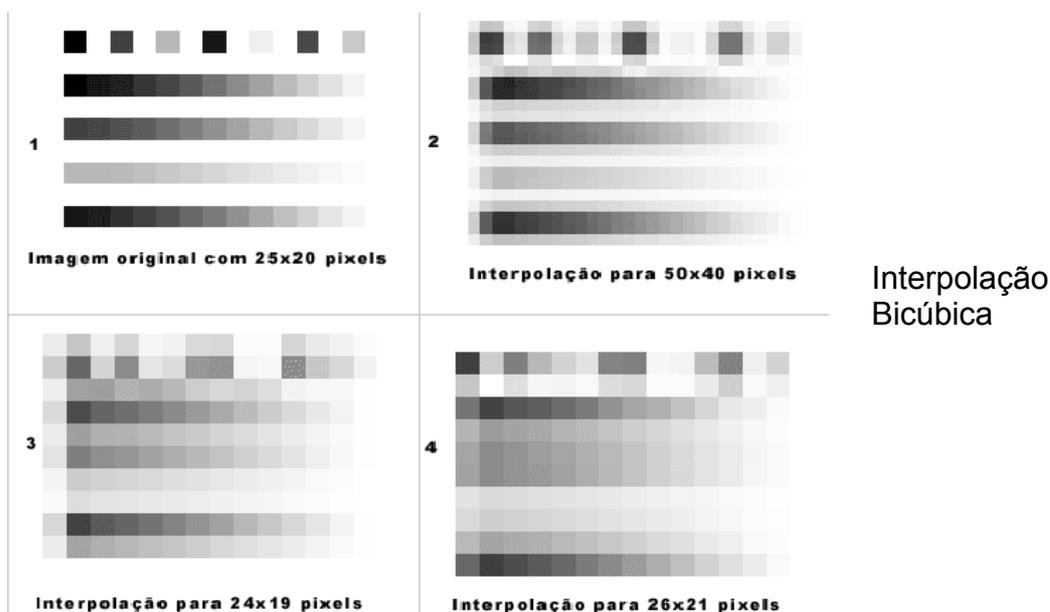


Figura 7 - Aumento de uma imagem (1) de 25x20 pixels para 50x20 (2) e 26x21 pixels (4), redução da mesma imagem para 24x19 (3), usando interpolação bicúbica. A perda de definição é menor quando se converte a imagem para múltiplos inteiros da resolução da imagem original.

De entre as múltiplas abordagens para melhorar a resolução de uma imagem destacamos três das quais mais frequentemente se utilizam:

1 - Realce de detalhes existentes na imagem. Esta operação é realizada através do aumento das componentes de altas frequências já presentes no sinal, através de técnicas como restauração [1] e/ou como técnicas de interpolação adaptativa [2, 3]. Nestes casos, são utilizados filtros adaptativos que aumentam as componentes de alta-frequência atenuadas no processo de amostragem; em alguns casos, aplica-se posteriormente um pós-processamento para reduzir os artefactos inseridos nas imagens interpoladas.

2 - Construção de uma única imagem de alta resolução a partir de uma série de imagens de baixa resolução. É uma das técnicas mais estudadas em SR. Alguns trabalhos abordam esta questão como um problema de reconstrução de imagens [4] ou também como fusão de dados [5]. Esta abordagem permite recuperar detalhes reais (não interpolados) das sequências de imagens e adicioná-los na imagem estimada. Os algoritmos de SR com base nesta técnica [6] geralmente passam por três fases distintas: i) aquisição

das imagens de baixa resolução ou estimação do deslocamento de cada imagem em relação a outra; ii) criação de uma grelha de alta resolução, onde os valores da mesma são interpolados a partir das imagens adquiridas; iii) remoção de ruídos dependendo do modelo de observação.

3 - Super resolução baseada em aprendizagem. O objectivo é estimar a informação de detalhes que não está presente na imagem original através de amostras de treino. Aqui são propostos algoritmos que usam algumas imagens de referência com o objectivo de treinar o algoritmo e assim produzir uma imagem com melhor qualidade do que a imagem original. Para construir a imagem de SR, o algoritmo extrai detalhes que não existem na imagem de baixa resolução a partir de um conjunto de imagens de treino [7].

2.2 - Enquadramento histórico

Tsai e Huang (1984) foram os primeiros a introduzir a técnica de SR a partir da fusão de múltiplas imagens de baixa resolução constituindo a reconstrução de imagem de alta resolução [10]. O seu trabalho foi motivado pela necessidade de melhorar a qualidade de imagens provenientes do satélite norte-americano, Landsat. O modelo proposto por eles baseava-se na translação de movimentos e resolvia o problema da aquisição e da restauração, mas não considerava os efeitos da degradação do sinal e do ruído. O método por eles desenvolvido explora a relação entre transformada rápida de cosseno (FCT) e a transformada de Fourier (FT) aplicada às imagens subamostradas.

Peleg e Keren (1987) destacam que é possível produzir uma imagem com resolução melhorada a partir de imagens de LR usando deslocamentos conhecidos entre as imagens. O processo da obtenção da imagem de HR, proposto pelos autores, inicia-se com uma imagem superamostrada, determinada como sendo uma aproximação inicial [13]. Este processo é iterativo, onde os novos valores para cada pixel são determinados, nesta imagem, após se verificar o menor erro originado pela diferença entre os níveis de brilho dos pixels das imagens de LR e os seus correspondentes na imagem de HR. Varia-se a intensidade de brilho dos pixels que são analisados na imagem super amostrada, no intervalo $[-1, 1]$, realizando-se combinações neste bloco, com o intuito de estabelecer qual o valor atribuído ao brilho que produz a menor diferença entre as imagens.

Peleg e Irani (1991) sugeriram melhorias no método apresentado por Peleg e Keren (1987). Os autores adoptaram a aquisição de imagens como base para a determinação de uma imagem de melhor resolução. Desenvolveram algoritmos iterativos para se obter um melhoramento na resolução espacial das imagens, com precisão subpixel, tanto em imagens de LR em níveis de cinzento como em imagens de cor [14]. Com os

deslocamentos da imagem e o processo de registo de imagens conhecidos, pode realizar-se a junção entre as imagens de LR, desde que pertençam à mesma sequência que compõe a cena, usando técnicas de interpolação obtendo-se assim uma imagem de melhor resolução espacial.

Gross e Ur (1992) desenvolveram uma abordagem para se conceber uma imagem de melhor resolução, através do conhecimento *a priori* dos deslocamentos existentes nas imagens de LR que fazem parte da mesma cena [15]. Apesar de abordar um método que utiliza as frequências dos sinais produzidas pela imagem, esse processo possui a mesma base do desenvolvido por Peleg e Irani (1991), pois utiliza a junção entre as imagens de LR, através da interpolação entre as mesmas. A determinação do valor de brilho do novo pixel na imagem de HR considera o menor erro possível, após serem analisados todos os erros originados pela variação dos valores de brilho dos pixels no intervalo $[-1, 1]$, além da aplicação de um filtro para eliminar o *blurring* causado anteriormente. As experiências foram elaboradas usando imagens simuladas e imagens reais, mas sempre com os deslocamentos conhecidos *a priori*.

Wiman (1992) trabalhou no aumento da resolução espacial de uma imagem digital (tons de cinzento) através de super amostragem. Desenvolveu um método para a obtenção de uma imagem de HR a partir de imagens de LR digitalizadas em *scanners*. Nesta abordagem, o processo de obtenção da imagem de HR baseia-se em equações que calculam os níveis de cinzento dos pixels na imagem criada com base nas adquiridas, média aritmética e deslocamentos conhecidos (adoptado como sendo de um pixel) [18].

Hathaway & Meyer (1999) desenvolveram um software denominado VISAR (Vídeo Image Stabilization and Registration) [16], utilizado para melhorar a resolução espacial de imagens e com a intenção de se realizarem análises de imagens científicas, vídeos na área da medicina, vídeos de circuitos fechados realizados por câmaras de segurança para identificação de intrusos, imagens que identificam motoristas infractores, aparelhos electrónicos, para além de uma série de outras situações que envolvem uma

sequência de imagens. Estas imagens de uma cena podem ser geradas por câmaras de vídeo ou fotográficas. Este software, inicialmente, era de uso restrito aos laboratórios do FBI, sendo posteriormente autorizada a sua comercialização. O software serve de apoio para o sistema que captura a cena (sequência de imagens) possibilitando a determinação de novos detalhes, levando em consideração a intensidade de luz existente e a velocidade no momento da aquisição. O software gera imagens de HR removendo os efeitos produzidos pelos deslocamentos e efeitos de zoom ocorridos durante a aquisição com base nos frames ou imagens que compõem a ocorrência, resultando na minimização de ruídos existentes nas imagens.

Baker & Kanade (1999) apresentaram uma solução para criar imagens de HR, a partir de sequências de imagens não-planares, não-rígidas e sujeitas a auto-oclusões, já que a grande maioria dos trabalhos desenvolvidos nessa linha utiliza imagens planares e rígidas [20]. Os autores procuraram melhorar a resolução espacial das imagens, ou partes das mesmas, que envolviam expressões faciais, pois há grande dificuldade na construção do modelo 3-D da face humana. Com base no algoritmo do fluxo óptico (algoritmo para a determinação dos parâmetros de aquisição e correção de imagens), conseguem produzir-se, simultaneamente, os parâmetros para correção e aquisição das imagens e a de melhor resolução.

Kang (2000) desenvolveu um software que visa formar imagens de HR a partir de imagens de LR. Para isso, determinou o número mínimo de imagens que deveriam ser adquiridas, tendo em consideração a minimização de custos em relação à aquisição das imagens e equipamentos utilizados, o tempo de processamento, o mínimo de ruídos gerados e como resultado final, uma imagem que pode ser interpretada de forma fiável [17]. O algoritmo implementado é capaz de determinar os parâmetros de deslocamentos e rotação entre as imagens de LR através do método de correspondência baseado na área por mínimos quadrados. Posteriormente, determina-se o aumento no tamanho da imagem e o valor do ruído mínimo que se deseja obter, ou seja, a qualidade da imagem resultante em relação ao erro médio que pode ocorrer entre a imagem de super resolução determinada e a imagem que

foi identificada no software como a de referência. A fase para a determinação da imagem de HR é executada iterativamente, até se obter a convergência em relação à qualidade desejada.

Joyeux (2001) aborda o processo de reconstrução de sequências de imagens danificadas com ênfase na restauração de filmes antigos em preto e branco [19]. O objectivo dos autores seria minimizar a quantidade de ruído que aparece na cena, com base na sequência de imagens que compõem o filme. Os trabalhos foram realizados utilizando processos no domínio das frequências para remover ruídos e *blurring* encontrados nas imagens, através de filtros passa-baixo e passa-alto, além das séries de Fourier.

2.3 - O Algoritmo testado

A elaboração deste trabalho, que culminou na construção de um instrumento de teste, permitiu a avaliação de um algoritmo SR desenvolvido no IBILI [11]. Este algoritmo foi inicialmente desenhado com o objectivo de melhorar a resolução espacial em imagens obtidas em SPECT (Single Photon Emission Computed Tomography), técnica que consiste em obter imagens bi-dimensionais ou tri-dimensionais (adquiridas em diferentes planos) através da medição da actividade dos isótopos previamente administrados no organismo de um doente [12].

Existem duas versões do algoritmo: uma desenvolvida no ambiente IDL e outra em MATLAB.

Tal algoritmo, assenta nos pressupostos teóricos da forma 2 de melhoramento da resolução de uma imagem. A partir de n^2 capturas (correspondendo a movimentos sub pixel na imagem original de referência) é calculada uma imagem cuja resolução é 'n' vezes superior à resolução original. A imagem assim obtida possui aproximadamente a mesma definição que uma hipotética imagem que fosse capturada com um detector com uma resolução "n" vezes superior ao usado realmente. O esquema da figura 8 representa a forma como são processadas as imagens LR adquiridas para se obter a imagem final HR.

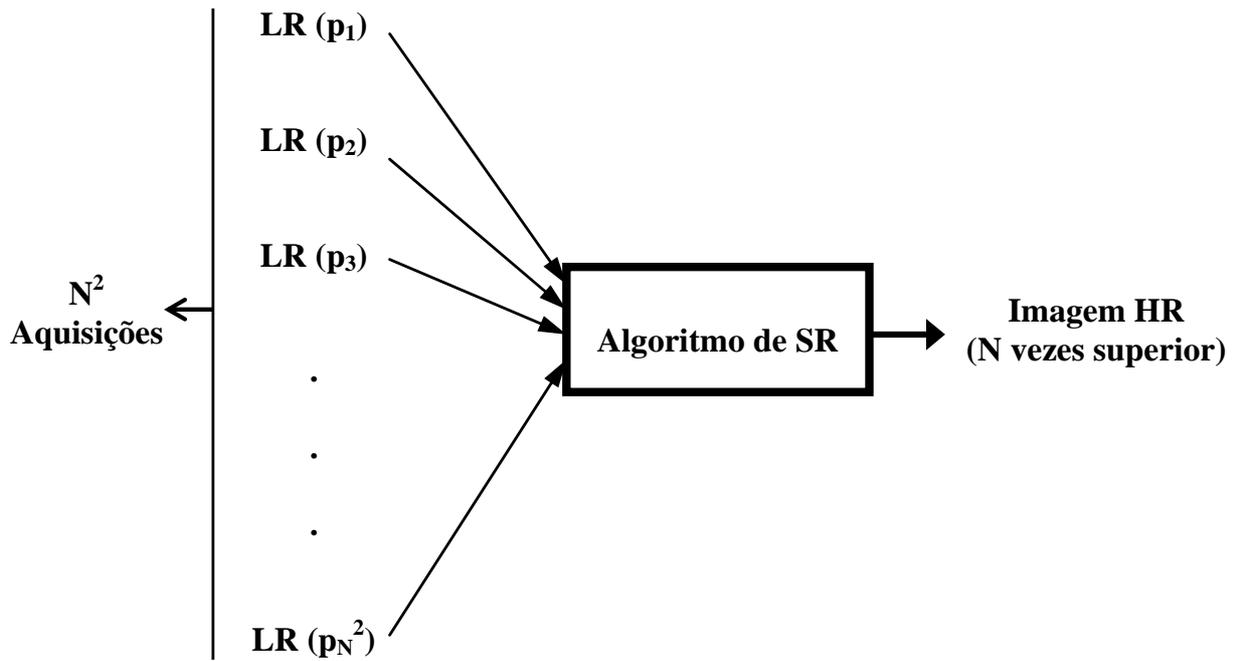


Figura 8 - Imagem de alta resolução (HR) obtida pelo algoritmo de super resolução (SR) processando as imagens de baixa resolução (LR) adquiridas nas posições p_i . O número de imagens LR necessárias é de N^2 , para uma melhoria de resolução por um factor de N .

Onde:

$LR(p_i)$ = Imagem de baixa resolução ($B \times B$) na posição p_i

HR = Imagem de alta resolução ($M \times M$) $\equiv (N \times B) \times (N \times B)$

N = Factor de resolução = M/B

$$\begin{pmatrix} a_{11} & \cdots & a_{1m} \\ \vdots & \ddots & \vdots \\ a_{N^2 1} & \cdots & a_{N^2 m} \end{pmatrix} \begin{pmatrix} S_1 \\ S_1 \\ \vdots \\ S_m \end{pmatrix} = \begin{pmatrix} {}^1L_1 \\ {}^1L_2 \\ \vdots \\ {}^1L_b \\ {}^2L_1 \\ \vdots \\ {}^{N^2}L_1 \end{pmatrix} \Leftrightarrow AX = Y$$

Figura 9 - Representação algébrica do problema de super resolução. A matriz A , designada por matriz sistema, engloba o processo de aquisição e as características das imagens LR. A matriz X , composta pelos elementos S_i , representa a imagem HR. A matriz Y é composta pelas imagens adquiridas.

Sendo:

A = Matriz do sistema

X = Imagem de HR

Y = Imagens de LR

S_i = Pixel i da imagem de HR

L_j = Pixel i da imagem de LR na posição j

O problema matemático que se coloca na SR é a obtenção da matriz X, o que pode ser facilmente obtido, em casos de ausência de ruído, pela inversão da matriz sistema. No entanto, a existência de ruído torna geralmente o problema mal condicionado e a inversão da matriz impossível. Por essa razão, é corrente o recurso a métodos alternativos de resolução da equação representada na figura 9 sendo o algoritmo SR apresentado um desses métodos iterativos.

A matriz sistema, A, tem em conta a relação entre o tamanho do pixel do sensor responsável pelas aquisições das imagens de LR e o deslocamento efectuado (suposto de menor valor ao da dimensão do pixel do sensor). Esta relação, encontra-se melhor caracterizada no esquema da figura 10:

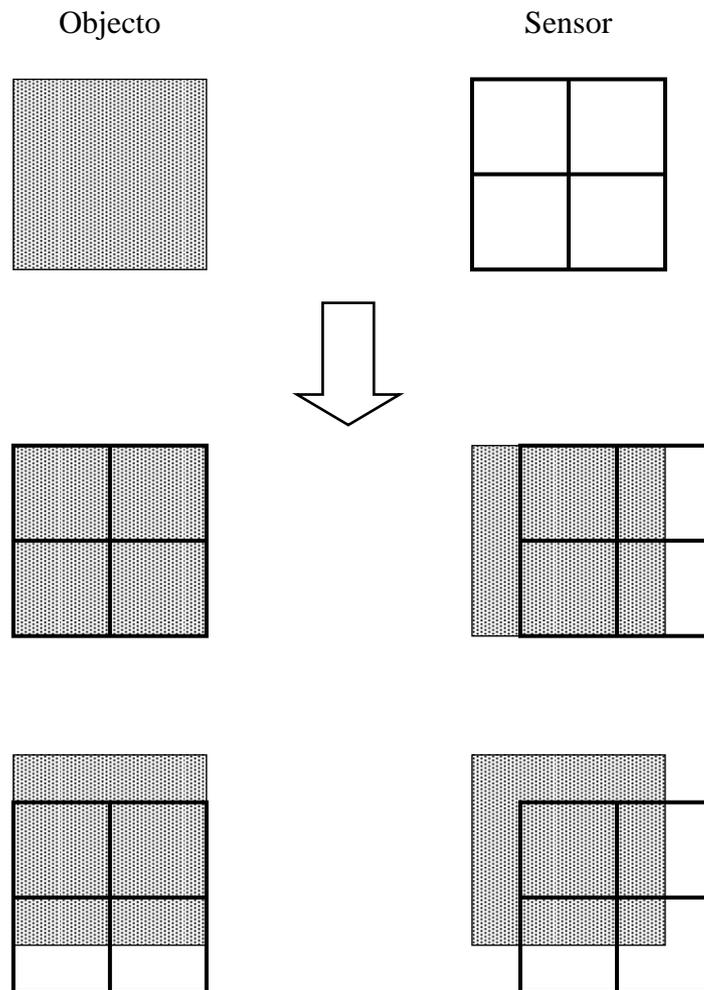


Figura 10 - Esquema de aquisição de imagens LR.

Assim, a determinação da matriz sistema assenta no cálculo da contribuição que um dado pixel, S_i , da imagem HR tem para um dado pixel, L_i , da imagem LR (ver figura 11).

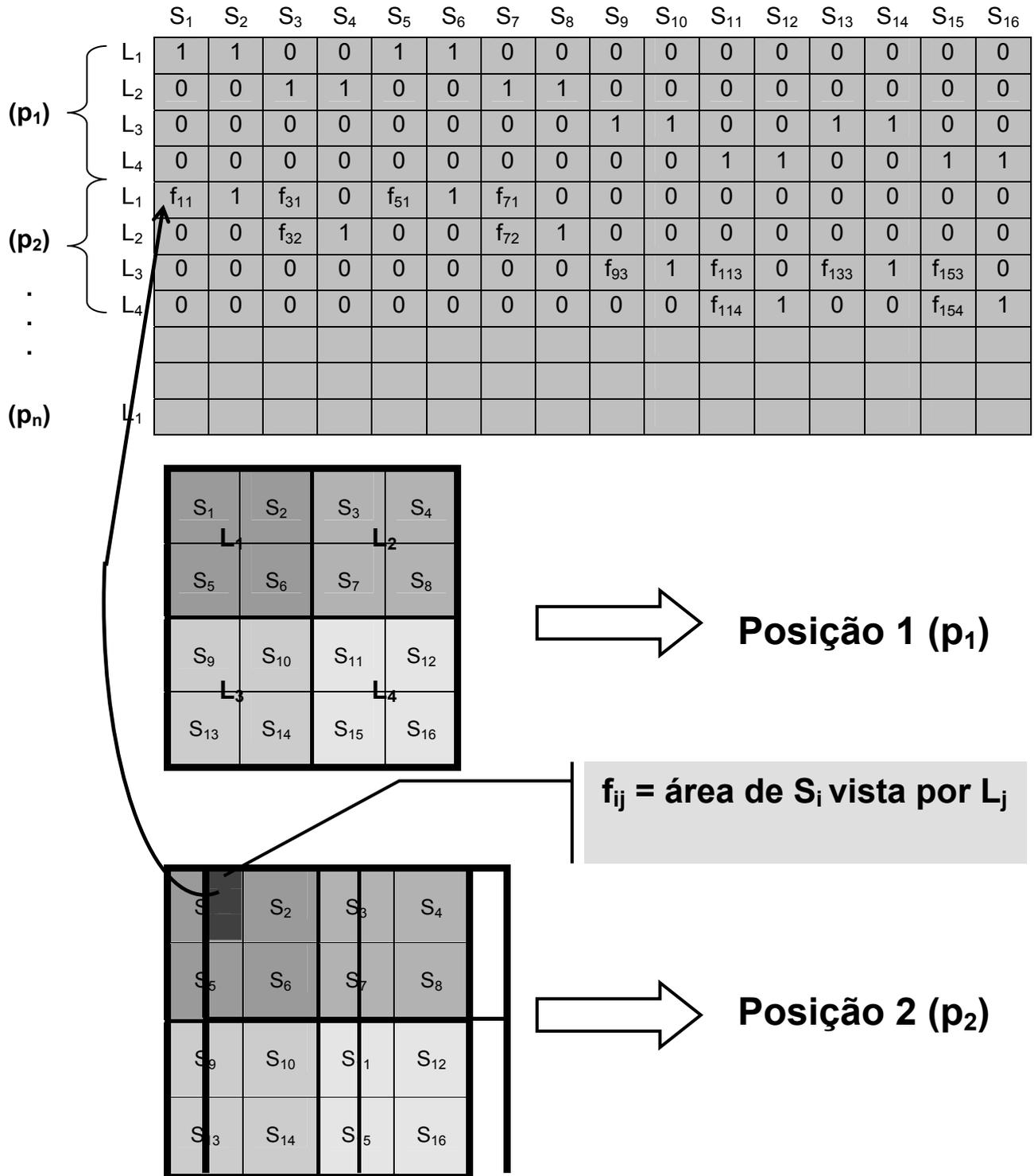


Figura 11 - Construção da matriz do sistema do algoritmo de super resolução (SR). Para cada posição p_j da imagem de baixa resolução (LR) o respectivo valor do pixel L_j , é repartido por todos os pixels S_i correspondentes na imagem de alta resolução (HR), vistos pelo pixel L_j . [11]

A matriz do sistema é então preenchida conforme as aquisições das imagens de LR existentes, pois convém salientar que têm de ser efectuadas pelo menos n^2 capturas para assegurar a fidelidade do resultado no aumento de resolução para 'n' vezes da imagem original. Por isso, apresentará as contribuições de quantas capturas disponíveis.

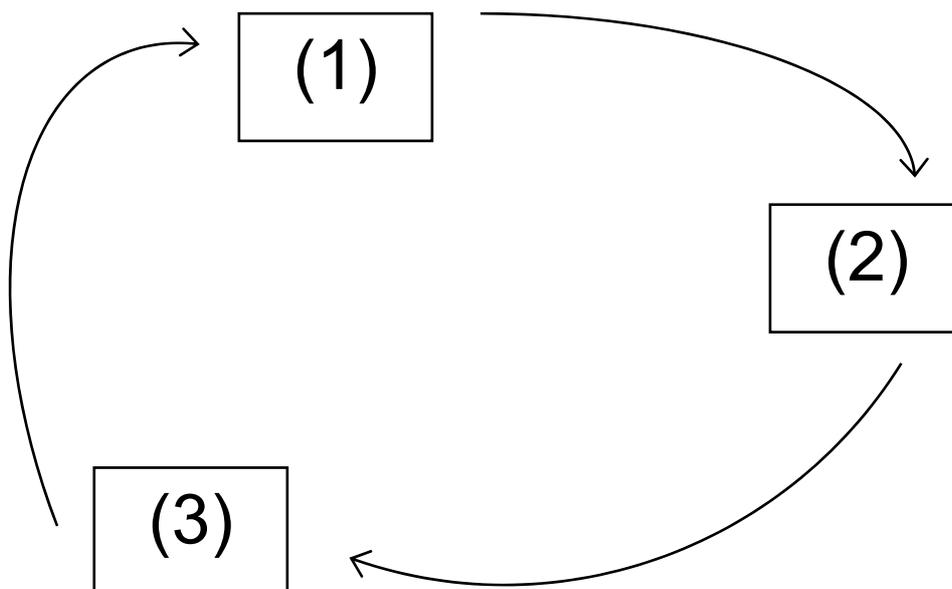
O algoritmo tem a configuração seguinte:

1 - O valor do pixel i da imagem de LR na posição j é repartida por todos os pixels na imagem de HR vista pelo pixel i . Esta repartição é pesada pelos valores actuais dos pixels da imagem de HR.

2 – Para cada posição j da imagem de LR é obtida uma estimativa do pixel correspondente na imagem de HR. Deste modo, para cada pixel da imagem de HR é feito um ajuste das estimativas obtidas.

3 – Os novos valores dos pixels da imagem de HR são determinados pela média das estimativas obtidas em 2. Estes valores são então usados em 1 como novos valores dos pesos.

Criando assim um método que constitui um ciclo fechado:



3 - Métodos

3.1 - Sensor

Antes de se proceder a uma descrição mais detalhada dos métodos utilizados para a concretização dos objectivos propostos neste trabalho, será relevante nesta fase, sem repetir os conceitos básicos, apresentar alguns fundamentos já descritos anteriormente que são extremamente importantes para justificar os métodos e direcções escolhidos para levar a bom termo a execução de todo o projecto.

Reportando às noções elementares para aquisição de imagens digitais, que se encaixam nas características de imagens de LR que pretendemos capturar, convém salientar e lembrar que essas imagens devem ser obtidas por um sensor de fraca resolução espacial ($>0,5\text{mm}$), para permitir deslocamentos sub pixel exequíveis. É compreensível que quanto maior for o tamanho do pixel mais fácil será encontrar uma solução que proporcione tais movimentos de precisão sub pixel, independentemente do método utilizado para realizar os deslocamentos.

Posto isto, é necessário analisar que tipo de equipamentos preenchem os requisitos relativos aos deslocamentos e, que sejam fáceis de encontrar e de baixo custo. O sensor tem de apresentar uma resolução na ordem dos milímetros (até à décima de milímetro). Logo, um sensor com uma resolução espacial 30x30 pixels representaria uma escolha aceitável. Actualmente, há uma disponibilidade muito grande de equipamentos de aquisição de imagem digital. A proliferação da venda de máquinas fotográficas digitais, *webcams* e telemóveis com essas funcionalidades é uma evidência irrefutável. No entanto, a qualidade dos produtos existentes no mercado supera largamente os nossos requisitos. Mesmo uma *webcam* rudimentar consegue reproduzir imagens com uma resolução na ordem dos 320x280 pixels, podendo ir até aos 640x480

pixels o que cai fora das nossas exigências, pois apresentam uma qualidade excessiva para a precisão dos movimentos que vamos ter. Por maioria de razão, as máquinas fotográficas digitais são ainda uma pior alternativa. Já os telemóveis gozam de uma implementação muito restrita e bastante difícil de interagir na elaboração de uma interface de comunicação. Assim, a hipótese de explorar a potencialidade do equipamento existente num rato óptico, também ele muito comum e bastante acessível, torna-se a escolha adequada.

Os sensores que equipam este tipo de instrumentos revelam características bastante interessantes no que diz respeito à capacidade de capturar imagem e sua resolução. A informação disponível acerca do funcionamento de dois deles, aponta para sensores de 32x32 pixels e 18x18 pixels. A título de curiosidade existem hoje ratos ópticos constituídos por sensores de 128x128 pixels.

Após analisarmos algumas possíveis soluções optámos pelo sensor, de 18x18 pixels *Agilent ADNS 2610*, com CCD monocromático e profundidade de cor 6 bit/pixel (figura 12 à direita). A questão que se impunha resolver era a de capturar de forma controlada imagens usando este sensor.

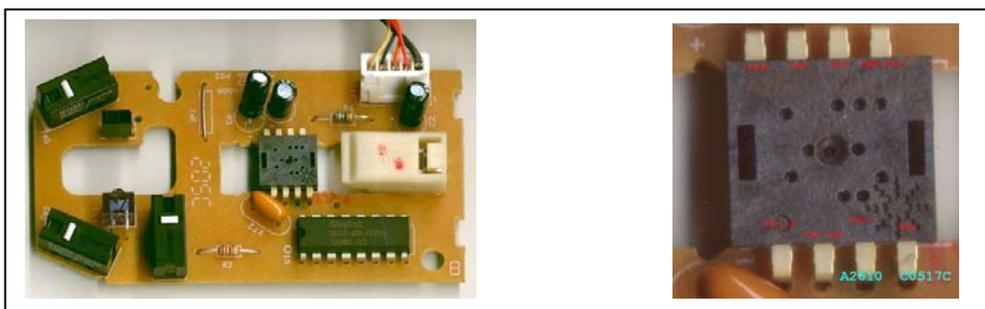


Figura 12 - Placa comum de um rato óptico (esquerda), Agilent 2610 (direita)

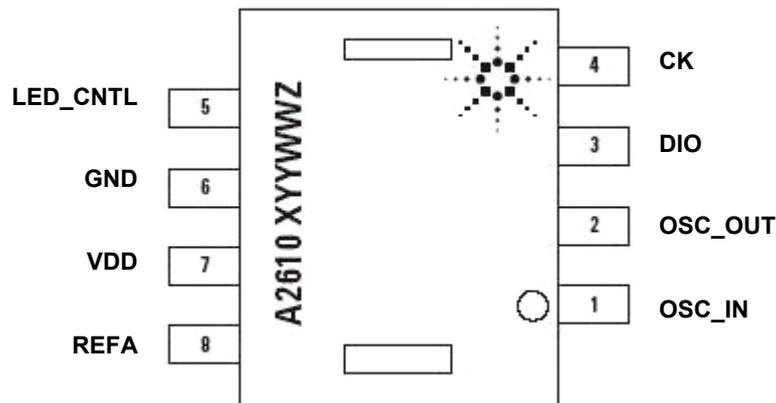


Figura 13 - Diagrama mecânico ADNS 2610 - 8 *pins* (visto de cima)

O *chip* (Agilent ADNS 2610) é composto por 8 *pins* dos quais são usados 4, a saber:

- DIO - Entrada e saída de dados
- CK - Entrada de clock (porta paralela neste caso)
- GND - Terra do sistema
- VDD - Fonte de alimentação (5V DC)

Após a análise e interpretação do manual do *chip* ADNS 2610 [25], é importante e necessário apresentar as principais funções deste sensor e quais são utilizadas para ser possível a visualização das imagens por ele capturadas. A enumeração de todas as operações e comandos previstos pelo equipamento não faria sentido, visto que o sensor é parte integrante de um rato óptico e grande parte desses comandos são para o bom funcionamento deste.

Em primeiro lugar, há que referir que embora o sensor tenha sido dimensionado para comunicar com a porta série de um computador pessoal (PC), a comunicação implementada é feita através da porta paralela. Esta opção assenta essencialmente na experiência prévia da sua utilização pelo autor em actividades anteriores. Na sincronização da transferência de dados é

preciso ter em conta o sinal de clock (ck) e a comunicação é efectuada byte a byte em vez de bit a bit como seria pela porta série.

A captura da imagem baseia-se nas operações de escrita e leitura de dados. Leitura entende-se como a recepção da informação que vem do sensor. Escrita é o envio de comandos apropriados.

A comunicação implementada com a porta paralela processa-se byte a byte e a construção da mensagem processa-se começando no bit mais significativo (MSB) até ao menos significativo.

A escrita, pressupõe dois bytes. O primeiro contém o endereço do registo para o qual se quer enviar, e o segundo que contém os dados a enviar. O endereço tem o primeiro bit a 1 que indica a direcção (escrita).

A leitura segue uma estrutura semelhante à escrita com dois bytes onde o primeiro é o endereço, o único byte a ser enviado, e o segundo os dados a receber. O endereço tem o primeiro bit a 0 que indica a direcção (leitura).

Para que o chip tome conhecimento que se está a enviar um bit é necessário pôr o clock a 0 e de seguida a 1 (transição ascendente), como se mostra no esquema das figuras 14 e 15.

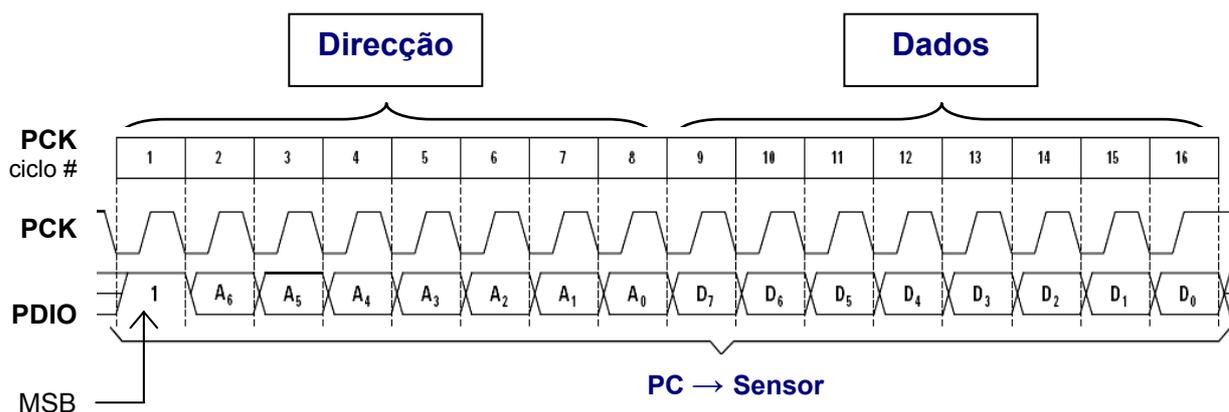


Figura 14 - Operação de 'escrita' constituída por dois bytes. O primeiro contém a informação da direcção dos dados (MSB=1) e endereço do registo pretendido, o segundo contém os dados. Esta operação tem a direcção do PC para o sensor. [25]

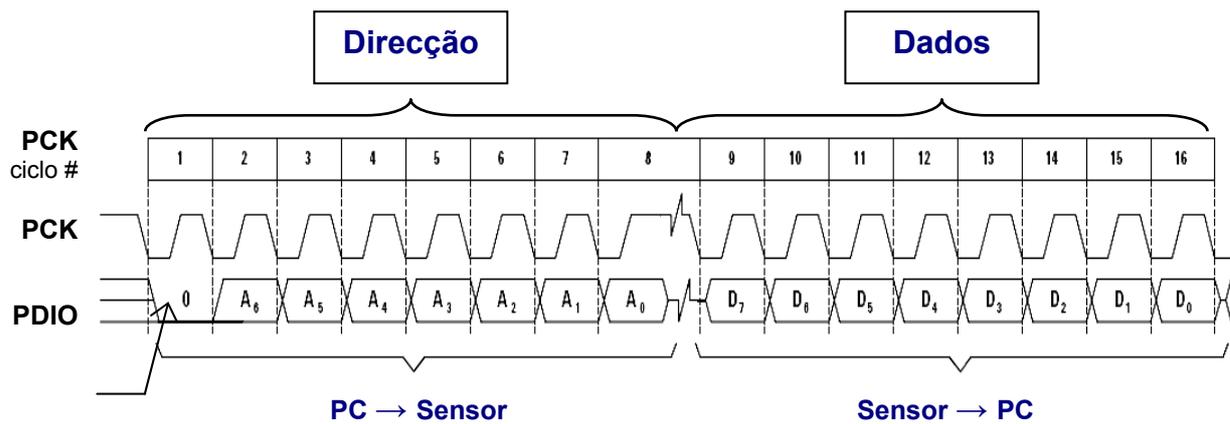


Figura 15 - Operação de 'leitura' constituída por dois bytes. O primeiro contém a informação da direcção dos dados (MSB=0) e endereço do registo pretendido, o segundo contém os dados. Esta operação tem a direcção do PC para o sensor no primeiro byte e do sensor para o PC no segundo. [25]

A configuração do primeiro bit faz todo o sentido visto que, na prática, está a lidar-se com um sensor que tem um pin bidireccional. Isto é, procede à transferência de dados pela mesma ligação física e, por isso, é necessário conceder um atributo para que se possa distinguir entre dados recebidos e enviados. Este mecanismo é abordado detalhadamente mais adiante nesta secção.

Para além do sincronismo com o sinal de ck da porta paralela existe ainda a configuração dos compassos de espera entre sucessivas leituras ou escritas para o registo do *chip*. O intervalo de tempo entre essas sucessivas operações é de 250ns e de 100µs entre o byte que contém o endereço e o que contém os dados, detalhes importantes para evitar dados incorrectos durante a comunicação.

O sensor tem uma característica aliada à sua função como detector de movimentos quando instalado no rato óptico: desliga-se quando não há movimento. O comando *Forced Awake* é extremamente útil, pois permite configurar o sensor para estar constantemente ligado se o comando for activado. Isto faz-se colocando o bit menos significativo (primeiro) do registo de configuração a 1. A relevância desta função na tarefa de capturar uma imagem é incontestável pois afasta a possibilidade do *chip* se desligar durante a actualização dos pixels que constituem uma imagem completa.

Para melhor se perceber como na realidade se processa a composição dos dados veja-se a figura 16 que apresenta o exemplo de um pedido e consequente resposta. Para que se obtenha uma imagem completa basta escrever para o registo *Pixel Data* (disponibiliza a informação do valor dos pixels que constituem a imagem completa actual), com o endereço 0x08 em hexadecimal, activar o sinal de *Forced Awake* e ler 324 vezes verificando se o bit *Data_Valid* está a 1, para garantir a integridade da imagem capturada.

Pedido								
Bit	Endereço							
	7	6	5	4	3	2	1	0
Campo	1	0	0	0	1	0	0	0
Bit	Dados							
	7	6	5	4	3	2	1	0
Campo	0	0	0	0	0	0	0	0

Resposta								
Bit	Endereço							
	7	6	5	4	3	2	1	0
Campo	0	0	0	0	1	0	0	0
Bit	Dados							
	7	6	5	4	3	2	1	0
Campo	1	1	0	0	0	0	0	0

Figura 16 - Acesso ao Registo *Pixel Data* para obtenção de imagem (pixel 1). Após efectuada a leitura 324 vezes obtém-se a imagem completa capturada pelo sensor.

No primeiro byte do pedido é indicada a direcção correspondente a escrita, primeiro bit a 1, a configuração do registo *Pixel Data*, '1000' (0x08 em binário) resultando o byte '10001000'. O segundo byte pode ser qualquer, neste caso tudo a zeros, '00000000'. Isto permite que a memória do *chip* seja limpa e a leitura da imagem se inicie no primeiro pixel. Quanto à resposta, no byte correspondente ao endereço é indicada a direcção de leitura colocando o primeiro bit a 0 para o mesmo registo *Pixel Data* formando-se o byte '00001000'. O segundo byte contém os dados relativos ao primeiro pixel da imagem que o CCD está a capturar, neste caso, informação dada pelo bit 7 colocado a 1. A verificação do bit *Data_Valid* aplicado a 1 indica que os dados são válidos. Os 6 bits restantes dizem respeito ao valor de cada pixel com a profundidade de cor já referida de 6 bit/pixel, que corresponde ao intervalo de 0 a 63 níveis de cinzento, zero totalmente preto e 63 totalmente branco. O exemplo apresentado na figura 10 tem o valor do nível 0, totalmente preto, o byte '11000000'.

Para que consigamos obter uma nova imagem, é necessário esperar 5s de modo a que o sensor actualize os valores dos pixels. Caso contrário, dependentemente do tempo decorrido, a imagem poderá apresentar valores de pixels relativos a momentos distintos de aquisição.

A imagem adquirida tem a forma apresentada na figura 17 que contém a grelha da distribuição dos pixels constituintes, de 'cabeça para baixo' se olharmos para a sua superfície.

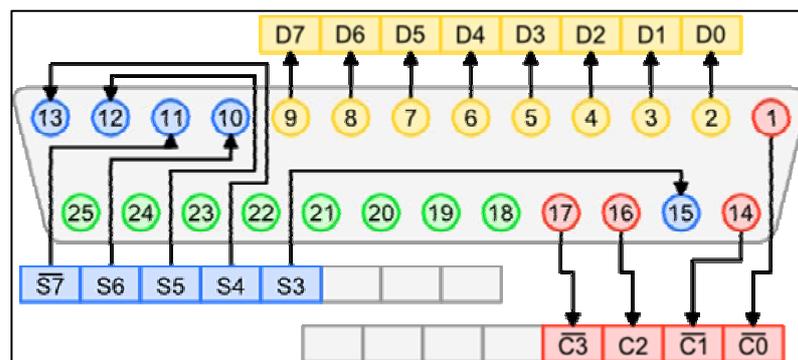
18	36	54	72	90	108	126	144	162	180	198	216	234	252	270	288	306	324
17	35	53	71	89	107	125	143	161	179	197	215	233	251	269	287	305	323
16	34	52	70	88	106	124	142	160	178	196	214	232	250	268	286	304	322
15	33	51	69	87	105	123	141	159	177	195	213	231	249	267	285	303	321
14	32	50	68	86	104	122	140	158	176	194	212	230	248	266	284	302	320
13	31	49	67	85	103	121	139	157	175	193	211	229	247	265	283	301	319
12	30	48	66	84	102	120	138	156	174	192	210	228	246	264	282	300	318
11	29	47	65	83	101	119	137	155	173	191	209	227	245	263	281	299	317
10	28	46	64	82	100	118	136	154	172	190	208	226	244	262	280	298	316
9	27	45	63	81	99	117	135	153	171	189	207	225	243	261	279	297	315
8	26	44	62	80	98	116	134	152	170	188	206	224	242	260	278	296	314
7	25	43	61	79	97	115	133	151	169	187	205	223	241	259	277	295	313
6	24	42	60	78	96	114	132	150	168	186	204	222	240	258	276	294	312
5	23	41	59	77	95	113	131	149	167	185	203	221	239	257	275	293	311
4	22	40	58	76	94	112	130	148	166	184	202	220	238	256	274	292	310
3	21	39	57	75	93	111	129	147	165	183	201	219	237	255	273	291	309
2	20	38	56	74	92	110	128	146	164	182	200	218	236	254	272	290	308
1	19	37	55	73	91	109	127	145	163	181	199	217	235	253	271	289	307

Figura 17 - Grelha de disposição dos pixels no CCD olhando para a sua superfície

A descrição das ligações físicas do sensor à porta paralela beneficia de uma breve introdução à constituição e possíveis conexões da porta paralela. Esta porta foi concebida na década de 70 como interface de comunicação para impressoras, sendo também conhecida por porta de impressora ou porta Centronics. (empresa que a fabricou e cujo nome é dado ao protocolo que hoje é padrão nos sistemas operativos que a utilizam para comunicação com outros periféricos [23]).

Tradicionalmente, num PC comum existem três endereços pré definidos para portas paralelas: LPT1, LPT2 e LPT3, que em hexadecimal são representadas por 0x378, 0x278 e 0x3BC respectivamente.

Com base nos mesmos argumentos utilizados na apresentação do sensor vamos debruçar-nos apenas nas ligações que são necessárias, pois na realidade não se trata de uma impressora, pretende-se sim que seja possível criar uma interface de comunicação entre o PC e o sensor. Assim sendo, a figura 18 mostra o diagrama mecânico de uma porta paralela e como se pode constatar, várias ligações são especificamente pensadas no intercâmbio com impressoras [24].



Pin	Descrição	Direcção	Pin	Descrição	Direcção
1	Strobe	Out			
2	Data Bit 0	Out	14	Auto Feed	In
3	Data Bit 1	Out	15	Error	In
4	Data Bit 2	Out	16	Initialise	In
5	Data Bit 3	Out	17	Select	In
6	Data Bit 4	Out	18	Ground	-
7	Data Bit 5	Out	19	Ground	-
8	Data Bit 6	Out	20	Ground	-
9	Data Bit 7	Out	21	Ground	-
10	Acknowledge	In	22	Ground	-
11	Busy	In	23	Ground	-
12	Paper End	In	24	Ground	-
13	Select In	In	25	Ground	-

Figura 18 - Diagrama mecânico da porta paralela e configuração dos *pins*

Referenciando os diagramas expostos nas figuras 13 e 18 especifiquemos agora as ligações feitas entre o sensor e a porta paralela. O pin 3 do sensor é de entrada e saída de dados (bidireccional), requerendo uma

ligação especial à porta paralela. Esta possui pins diferentes para entrada e saída de dados. Os utilizados foram o pin 12 para entrada e o pin 5 para saída. A ligação é feita então com o auxílio de um díodo que tem a propriedade de conduzir corrente eléctrica num sentido e bloqueá-la no outro. Isto evita que quando se está a enviar dados do sensor através da porta paralela não vão para o pin errado, ou seja, permite que sejam direccionados directamente para o pin 12 e sejam bloqueados para o pin 5. Aquando do envio da porta paralela para o sensor o díodo permite a condução para o pin bidireccional do sensor.

O pin 4 do sensor vai ligar ao pin 9 da porta de onde vem a informação do clock para sincronização de transferência de dados.

Relativamente à fonte de alimentação do *chip* ela é feita através do cabo USB original do rato óptico de onde foi retirado. O fio que transporta a corrente é ligado directamente ao pin 7 do sensor.

Os pins *ground* de cada aparelho são ligados um ao outro para fechar o circuito, pin 7 do sensor ao pin 25 da porta paralela.

Esta tarefa está representada na figura 19 onde se mostram as soldaduras efectuadas para concretizar as ligações descritas.

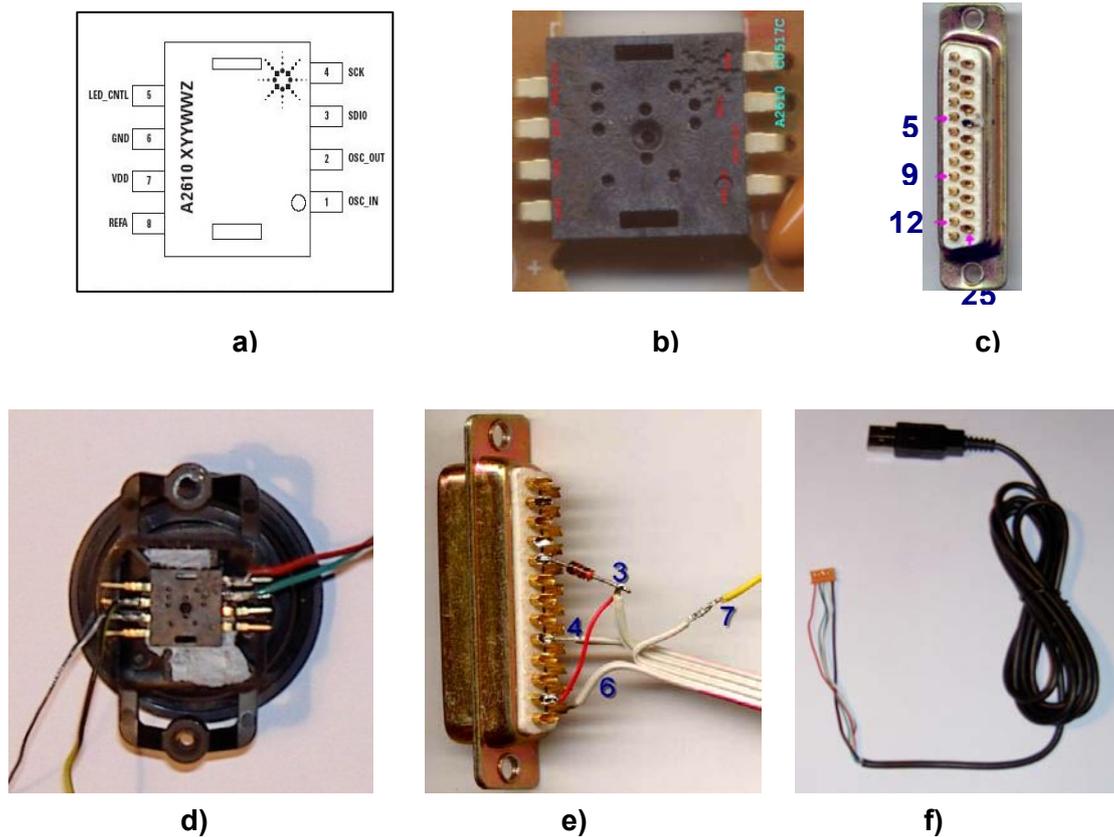


Figura 19 - Plano de ligações e respectivas soldaduras. a) diagrama do sensor, b) sensor, c) porta paralela, d) ligações/soldaduras no sensor, e) ligações/soldaduras na porta paralela (a azul os pins do sensor), f) cabo USB utilizado para fonte de alimentação do sensor (pin 7).

3.2 - Câmara

A óptica que foi utilizada na construção da câmara digital, é uma peça fundamental que permite a focagem e definição do campo de visão da imagem que se pretende adquirir.

Devido à função que se pretende obter de um rato, ou seja, detectar o mais ínfimo movimento para melhor precisão, o sensor é extremamente sensível. Aliada a essa sensibilidade está a óptica fixa na base do rato, onde encaixa o próprio sensor como se mostra na figura 20.

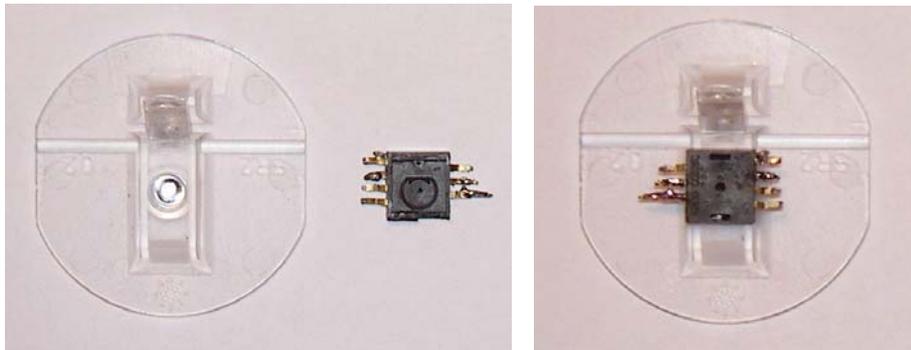


Figura 20 - Óptica aplicada ao sensor para utilização no rato óptico.

Uma das grandes contrariedades encontradas na execução da construção da câmara foi sem dúvida a função da óptica. Com a óptica mostrada na figura 20 não se conseguem focar imagens com as características adequadas àquilo que se pretende. O seu campo de visão é extremamente pequeno, vários testes foram feitos sobre texto escrito e não se conseguiu focar de maneira a que uma simples letra do tipo 'Arial' e tamanho 12 fosse visualizada. A figura 21 mostra como é focada uma linha com a espessura de 0,5mm, quando a óptica é colocada sobre a linha. Embora na imagem não seja aparente, a linha ofusca por completo o círculo que define a óptica.

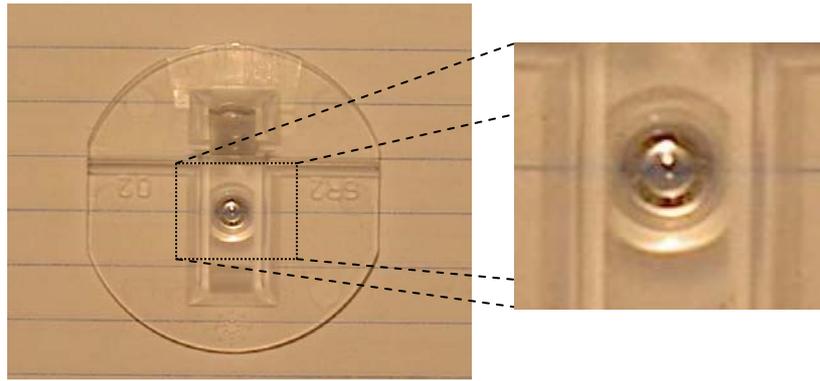


Figura 21 - Óptica sobre uma linha de 0,5mm de espessura. O campo de visão é muito pequeno, impedindo a focagem de grandes objectos.

A solução encontrada para resolver tal problema foi reutilizar uma *webcam* avariada. A ideia era aproveitar a óptica nela instalada e aplicá-la ao sensor em questão. As máquinas escolhidas podem ser observadas na figura 22: uma Logitech QuickCam Messenger e uma Samsung Pleomax, das quais foram retiradas as respectivas ópticas e, posteriormente, aplicadas a dois sensores ADNS 2610, fixadas com uma goma de características adesivas muito parecidas com as da plasticina. Estas constituem assim os dois exemplares de câmaras que foram construídas, ver figura 23.



Figura 22 - *Webcams* utilizadas para retirar ópticas. a) Samsung Pleomax, b) Logitech QuickCam Messenger.

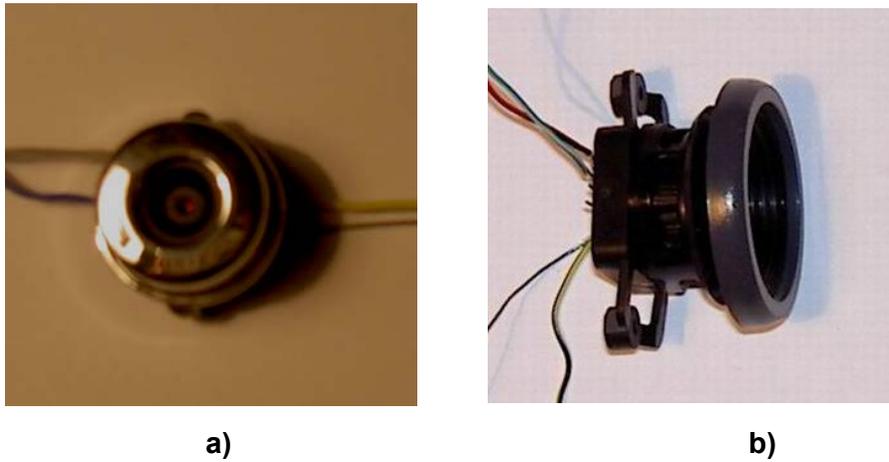


Figura 23 - Ópticas aplicadas aos sensores. a) Samsung, b) Logitech

Após algumas experiências, a óptica Logitech apresentou uma distorção nas imagens bastante exagerada, devido ao ângulo de abertura ser elevado para o sensor *Agilent*. Por este motivo, foi completamente abandonada ficando apenas a ser utilizada a câmara com a óptica Samsung. Esta câmara permite capturar imagens com dimensões de 2x2 cm a uma distância de 10 cm do objecto.

3.3 - *Plotter*

A resolução e dimensões das imagens adquiridas pela câmara aliada ao facto de se pretenderem movimentos sub pixel implicam deslocamentos na ordem dos milímetros e décimos de milímetro, dependendo do aumento da resolução que se pretenda obter.

A opção de fabricar de raiz *hardware* com mecanismo capaz de responder a tais exigências foi abandonada, devido não só à dificuldade inerente à sua implementação, mas também à limitação do tempo disponível para a execução do trabalho. Em alternativa, procurámos uma solução com equipamento já existente nada dispendioso e de implementação acessível. Daí utilização de uma *plotter* de tabuleiro à qual foi adaptado à sua cabeça de impressão um suporte fixo para deslocamento da câmara anteriormente descrita.

A *plotter* é uma *Graphtec Plotter MP3200*, de resolução 16640x10720 pontos com o tamanho do passo 0,025mm e de erro de precisão $\pm 0,3\%$, bastante aceitável para uma peça com duas décadas, ver figura 24.

Possui duas interfaces de comunicação, uma com o protocolo RS-232 (porta série) e outra com protocolo Centronics (porta paralela), características que fazem da MP3200 uma boa opção para se alcançarem os objectivos.



Figura 24 - *Graphtec Plotter MP3200.*

Ao nível mecânico esta *plotter* é uma boa escolha. É imperativo comandá-la de maneira a que a precisão dos movimentos seja a que pretendemos.

Quase todas as *plotters* usam uma linguagem de controlo desenvolvida pela HP, sendo mesmo a primeira linguagem para controlo de *plotters*, de nome HP-GL (*Hewlett-Packard Graphics Language*) que se baseia em comandos compostos por códigos de duas letras seguidos de parâmetros opcionais e finalizados com ponto e vírgula (;). A quantidade de comandos possíveis é enorme, mas alguns exemplos podem ser referidos para melhor compreensão da simplicidade da linguagem. Não esquecer que estamos a tratar de instruções para traçar ou desenhar que exigem movimentos precisos, ver tabela 1:

HP-GL	
COMANDO	DESCRIÇÃO
AA100, 100, 60;	'ARC ABSOLUTE' - desenha a partir da posição actual um arco de centro na posição 100, 100 (x,y) com um ângulo de 60°
CI250;	'CIRCLE' - desenha um círculo com um raio de 250 na posição onde o comando foi recebido
PU50, 50;	'PEN UP' - levanta a caneta e move a cabeça para a posição 50, 50 (x, y)
SP1;	'SELECT PEN' - selecciona a caneta 1
PA2000, 1500;	'PLOT ABSOLUTE' - move a cabeça para a posição 2000, 1500 (x, y)
PR1000, 1500;	'PLOT RELATIVE' - move a cabeça com incremento 1000 em x e 1500 em y

Tabela 1 - Exemplos de comandos em HP-GL

Mais simples se torna quando a função que se espera ter da *plotter* é apenas efectuar movimentos, sejam eles em absoluto (relativos ao *offset* '0, 0'), sejam incrementais (relativos ao *offset* actual). Assim, são precisos apenas dois comandos que nos permitam posicionar a cabeça da *plotter* na posição exigida, PA (PLOT ABSOLUTE) e PR (PLOT RELATIVE).

3.4 - Suporte

Dentro das opções de se movimentar a câmara ou a imagem, a decisão ficou pela primeira fazendo-se um levantamento do material necessário para apetrechar a câmara na cabeça de impressão da *plotter*.

Este suporte tem de garantir a fixação rígida e precisa da câmara numa posição horizontal paralela à cena e permitir que se mova na vertical para possível necessidade de focagem, afastando ou aproximando a câmara ao alvo pretendido.

A peça principal responsável pela ligação da estrutura dimensionada à cabeça da *plotter* é feita em chapa zincada com a forma ergonómica da cabeça à qual é aparafusada sem danificar a última. Na chapa estão ainda soldados dois parafusos que permitem a colocação do resto da estrutura, ver figura 25a).

Todas as peças que constituem o suporte estão apresentadas na figura 25 bem como a sua aparência depois de montado e aplicado na *plotter*, suportando a câmara.

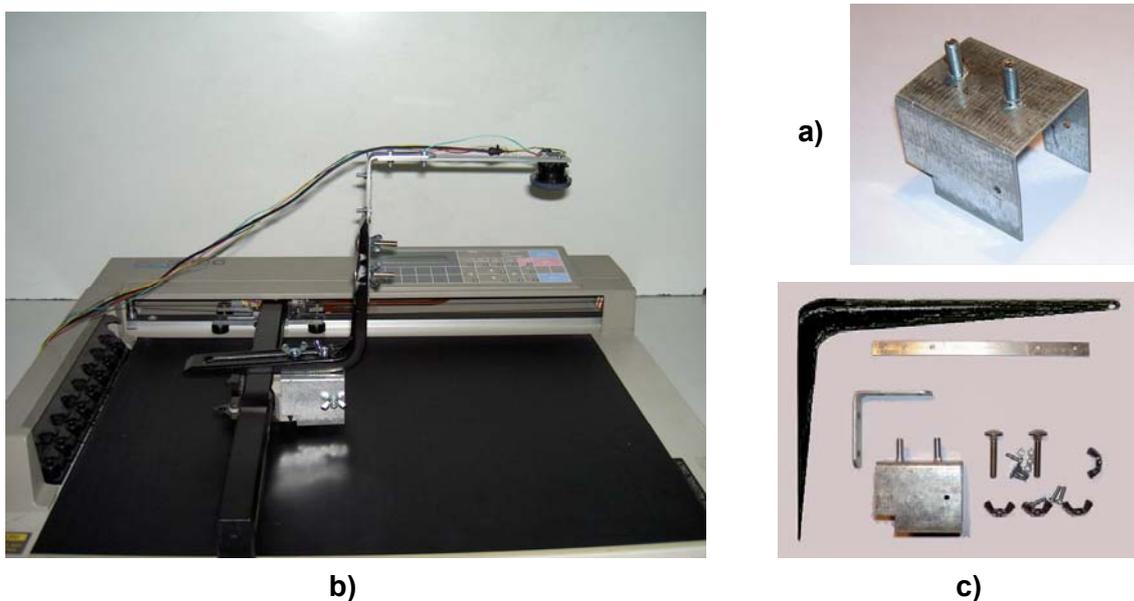


Figura 25 - Suporte para a câmara. a) peça principal, b) aplicação na *plotter*, c) peças constituintes do suporte.

Falhas na estrutura não demoraram muito a revelar-se. O ajuste da altura da câmara para focar imagens de maior dimensão era uma das limitações. Além disso, revelou ser pesada demais o que por vezes impedia a *plotter* de executar o movimento correcto.

Brevemente se passou a projectar uma nova estrutura que eliminasse ou pelo menos minimizasse essas limitações. Foram escolhidas ligas mais leves para ser possível aumentar as dimensões do suporte em altura e ao mesmo tempo reduzir o seu peso.

Com o apoio de uma mola e de uma calha é agora também mais fácil movimentar com rigor na vertical a haste horizontal que sustenta a câmara. A peça principal que agarra a cabeça foi a única mantida. A Figura 26 mostra a segunda e derradeira versão do suporte que se revelou muito versátil, preciso e funcional. Vários testes foram feitos para se ter garantias de que o peso da nova estrutura em nada afectava o rigor dos movimentos, colocando-se uma escala no tabuleiro da *plotter* e fazendo deslocar em curtas e longas distâncias a cabeça com e sem suporte. Utilizaram-se imagens quadradas de calibração com o objectivo de se posicionar na altura correcta a câmara e com isto saber que dimensões têm os pixels constituintes das imagens a capturar, ver figura 27.

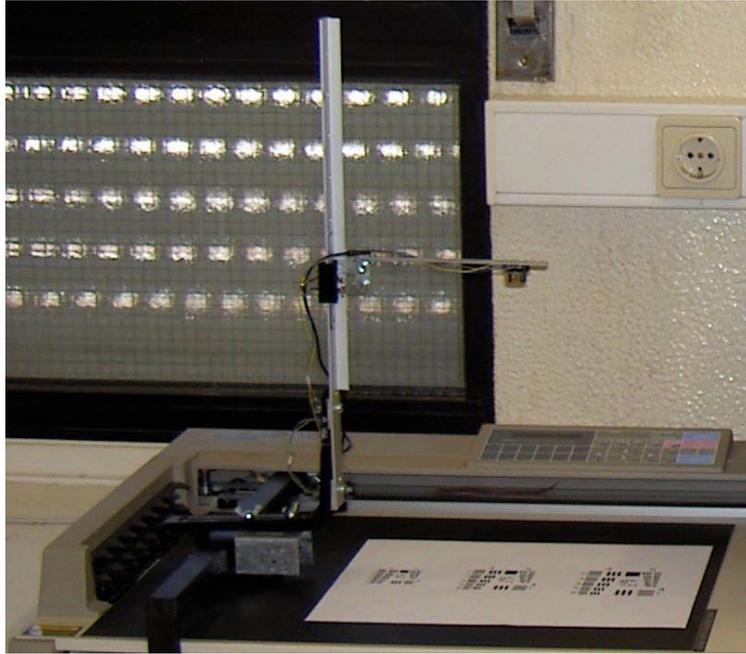


Figura 26 - Suporte definitivo aplicado na *plotter*.

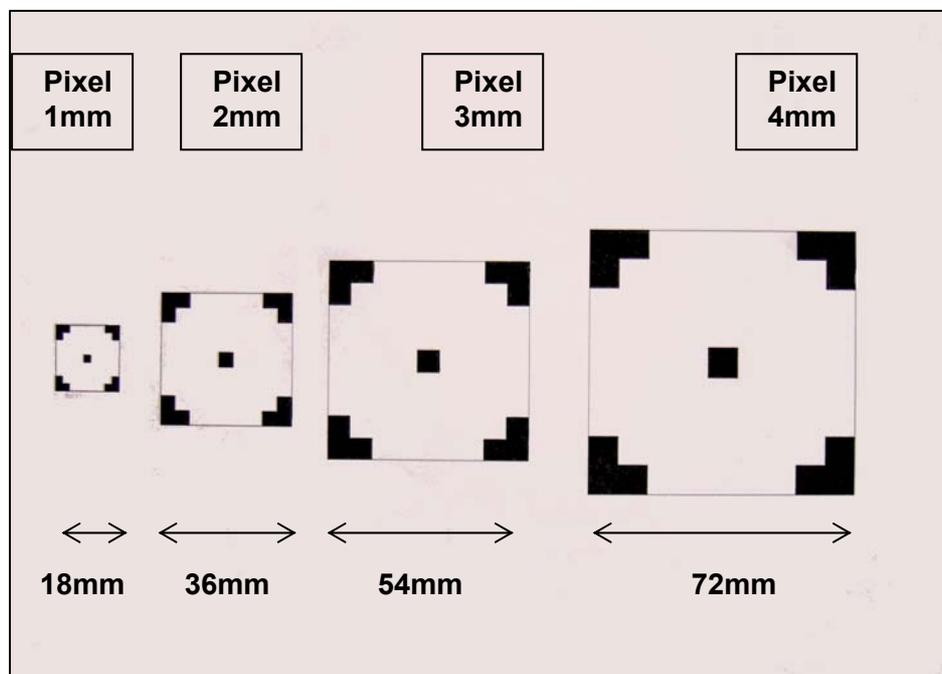


Figura 27 - Imagens quadradas de calibração com respectivas dimensões.

3.5 - Software

Nesta fase é necessário fazer uma breve análise dos requisitos que são exigidos e que foram usados na integração dos sistemas que funcionam em sinergia, constituindo apenas uma ferramenta. Ter apenas uma câmara e uma *plotter* independentes não chega e não serve para atingir o pretendido.

Foi utilizado um PC com um processador de 1.6GHz de clock, 1GB de memória RAM e com a particularidade de ter sido adicionada uma porta paralela extra, como mostra a figura 28, de modo a que seja possível comunicar com os dois periféricos ao mesmo tempo. Isto fez-se introduzindo uma placa de protocolo PCI (Protocol Control information) de porta paralela na Placa Mãe ou Principal do computador, proporcionando a capacidade de activar as portas LPT1 e LPT2, como já referido.

O programa que controla o equipamento foi desenvolvido na linguagem C, compilado pelo *Microsoft Visual C++ Express Edition*, do qual resultou uma DLL (Dynamic-link library). Esta Biblioteca de ligação dinâmica serve uma interface gráfica desenvolvida em MATLAB.

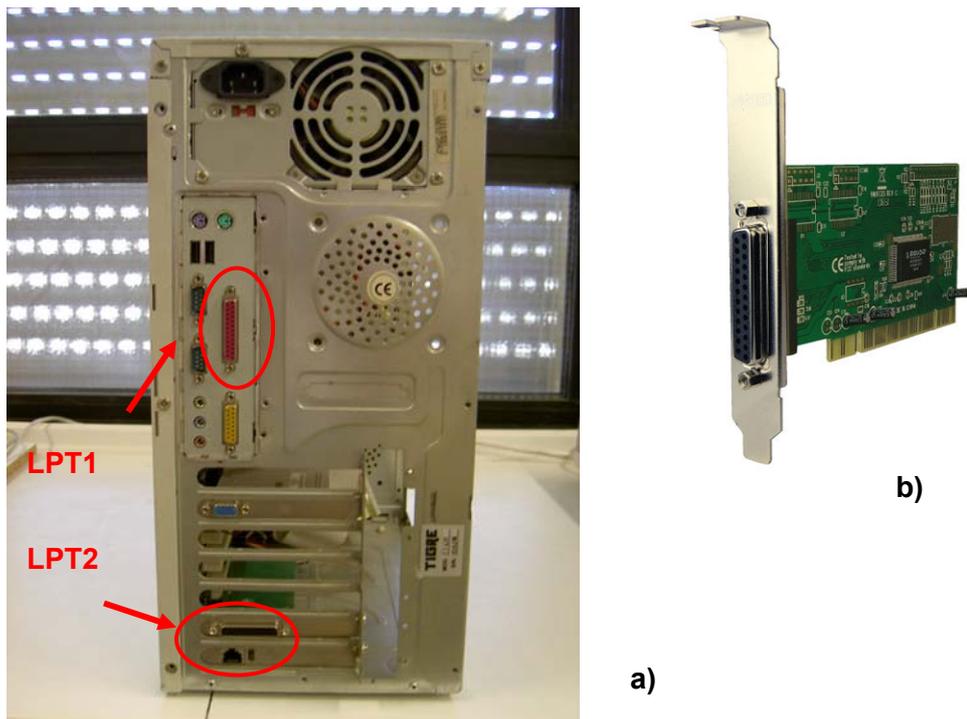


Figura 28 - a) PC com duas portas paralelas, b) Placa PCI / porta paralela

No início da implementação do software de apoio ao equipamento, a ideia era desenvolver toda a plataforma em MATLAB, acima de tudo por se revelar um programa versátil e prático devido aos pacotes de ferramentas que prevêm a comunicação através de diversos protocolos. Pode conceber-se facilmente o tratamento da entrada e saída de dados, sejam eles obtidos com o protocolo RS-232, Centronics ou mesmos com as versões mais recentes de USB. Contudo, revelou ser uma impossibilidade comunicar com MATLAB através da porta paralela adicional (LPT2), fosse ligado a câmara ou a *plotter*. Isto deveu-se ao facto de a porta paralela adicional não utilizar um dos endereços padrão para portas paralelas (acima mencionados).

A criação da DLL na linguagem C foi a solução prática encontrada. O Windows através da sua interface para programação de aplicações (API) torna relativamente fácil a interação com o protocolo Centronics. Com a linguagem C e com Funções do Windows desse protocolo, pode comunicar-se directamente com a porta LPT que se desejar, bastando referenciar o seu endereço.

Relativamente à *plotter*, a comunicação é feita utilizando as Funções do protocolo; como precisamos apenas de dois comandos para que os movimentos se processem como desejado, basta abrir a ligação no endereço da porta onde a *plotter* estiver ligada e enviar directamente o comando. Foi estipulado que a LPT1 serviria a câmara e a LPT2 a *plotter*.

A DLL tem então duas Funções de nome *MovePlotterRelative* e *MovePlotterAbsolute* para comunicar com a porta LPT2, responsáveis pelos movimentos relativos à posição actual e pelos movimentos absolutos relativamente à origem respectivamente, com dois parâmetros inteiros de entrada (valores x e y do movimento).

O sensor não comunica com base em Centronics e por isso é necessário criar todas as Funções para interagir com o *chip*. Baseado nas características e no princípio de funcionamento do sensor já apresentado a DLL nomeia ainda mais três Funções para comunicação com a LPT1: Ler, Escrever e Scan. É ainda utilizada uma outra Biblioteca 'inout32.dll' para que se tenham permissões do sistema operativo (Windows) no envio e recepção directos de dados pela porta paralela.

A Função Ler trata os dados vindos do ADNS 2610, ou seja os bytes que contém a informação solicitada. Tem como parâmetros de entrada um byte e a porta associada.

Embora as operações de escrita não sejam só solicitações ao sensor, estas estão a cargo da Função Escrever, que através dos dois bytes de parâmetros de entrada e a porta, escrevem para o registo de configuração do sensor e para o *Pixel Data*.

Quanto à Função Scan, ela é especificamente destinada à construção da matriz que contém todos os valores dos pixels que representam a imagem que está a ser capturada. Isto é feito enviando para o Registo *Pixel Data* o respectivo pedido (rever Figura 16). A resposta contém os parâmetros da Função que constituem uma matriz de números reais (imagem) e um inteiro, o endereço da porta.

Resumindo pode afirmar-se que a Função Ler lê um valor de um endereço da memória do *chip*, a Escrever escreve um valor para um endereço da memória e a Scan lê uma imagem inteira do *chip*.

A Figura 29 mostra a interface gráfica (GUI) feita em MATLAB que com a DLL comandam todo o mecanismo de captura de imagens baseado na técnica de *substepping*.

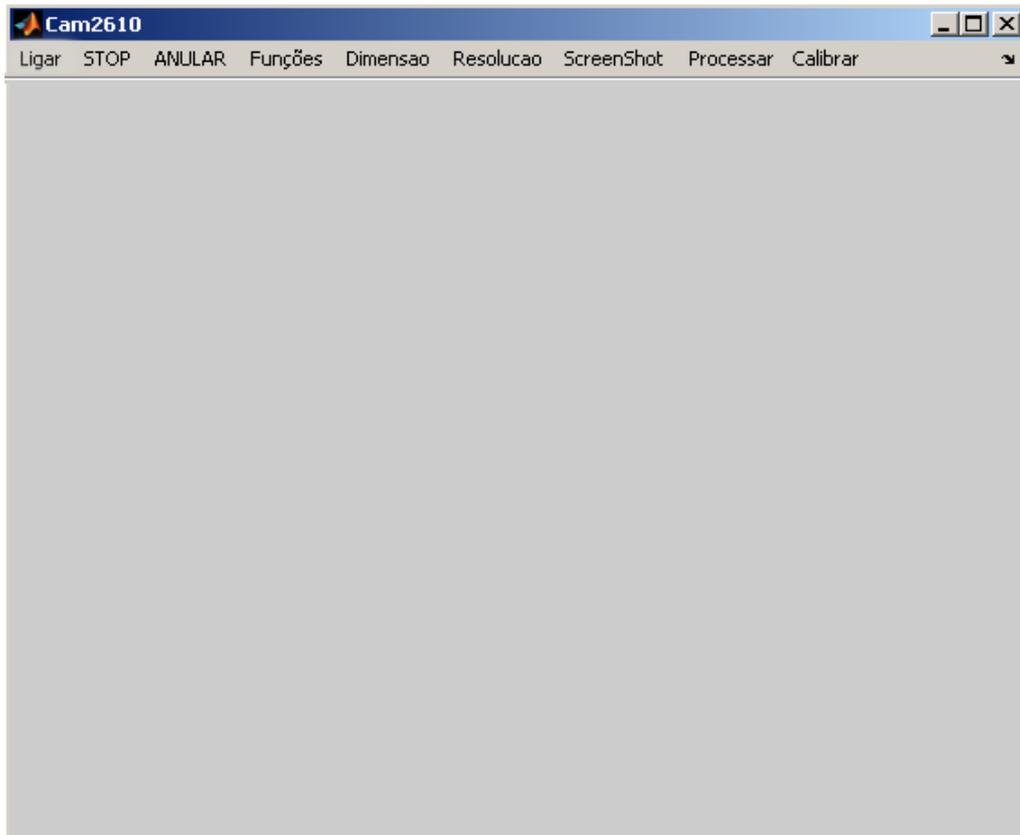


Figura 29 - Interface gráfica (GUI) em Matlab para controlo do equipamento.

Para que melhor se perceba como toda a aplicação opera com o mecanismo e que utilidade tem cada uma das opções disponíveis, faça-se uma apresentação mais detalhada.

O programa em MATLAB tem por nome 'Cam2610Move' que ao correr chama a GUI 'Cam2610' e faz a ligação dinâmica à Biblioteca da DLL chamada 'LibNar'. As opções vêm bloqueadas excepto a 'Ligar', que permite fazer a ligação à LPT1 e detectar se o sensor está ligado, para evitar que se possa prosseguir com a aplicação sem o sensor estar activo e ocorrerem erros. A Figura 30 demonstra como é feita esta operação que desbloqueia as opções se o sensor for detectado e permanece bloqueado caso contrário.

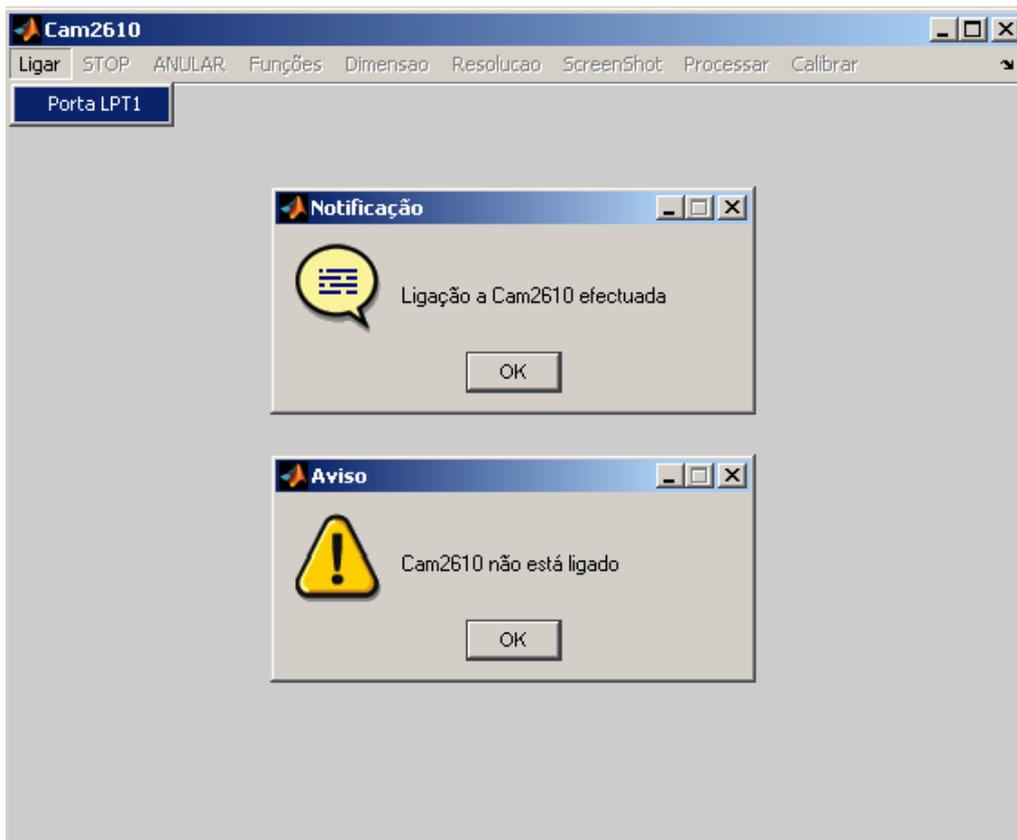


Figura 30 - GUI com a opção 'Ligar' e possíveis respostas.

Se ocorrer a situação em que a resposta é o aviso de que o dispositivo não está ligado, terá de se fechar o programa, ligar o aparelho e tornar a correr. Caso contrário reúnem-se condições de se prosseguir, sendo activada a opção 'Funções', ver Figura 31:

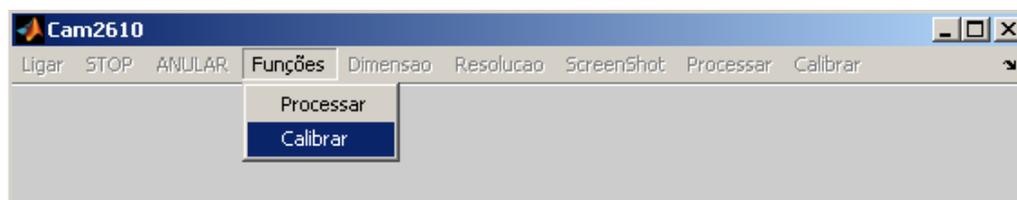


Figura 31 - GUI com a opção 'Funções' e selecção de 'Calibrar'.

Seleccionada a opção 'Calibrar', é então possível fazer a calibração da câmara com a imagem de dimensão pretendida (descrita na Figura 27) e efectuar um 'ScreenShot' permitindo guardar um ficheiro com o nome e formato 'ScreenShot.bmp', que representa a imagem que a câmara está a capturar. Um exemplo pode ser visto na Figura 32.

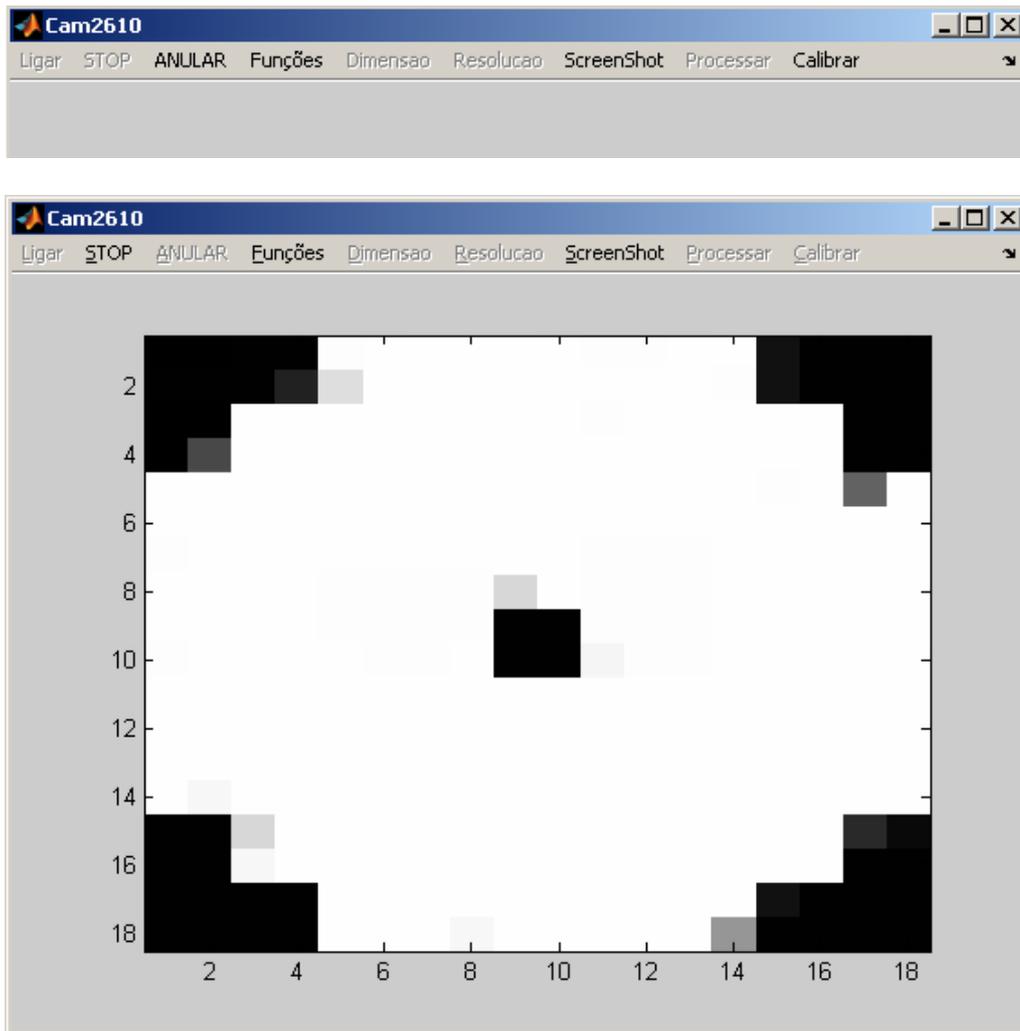


Figura 32 - Opção 'Calibrar' com visualização da imagem de calibração.

Durante a visualização da imagem de calibração, a qual é actualizada a cada 5s, pode ajustar-se o posicionamento horizontal movendo a imagem, ou o vertical, subindo ou descendo a câmara na calha do suporte, de maneira a que a imagem coincida nas suas dimensões com a imagem capturada, permitindo

no rigor possível saber que tamanho tem o pixel da imagem que a câmara está a detectar. A posição de calibração é por defeito 1000 passos de *plotter* em x e 1500 em y. A Função 'Calibrar' faz uma chamada remota à Função 'Scan' da Biblioteca dinâmica já apresentada, onde vai recolher a matriz que define a imagem. Esta depois de convertida é tratada e mostrada na GUI como mostra a Figura 32. A opção 'STOP' faz com que a actualização da imagem termine.

Quando o decurso de calibração estiver concluído pode então proceder-se ao processamento de imagens com base em movimentos de sub pixel escolhendo agora a opção 'Processar', Figura 33.

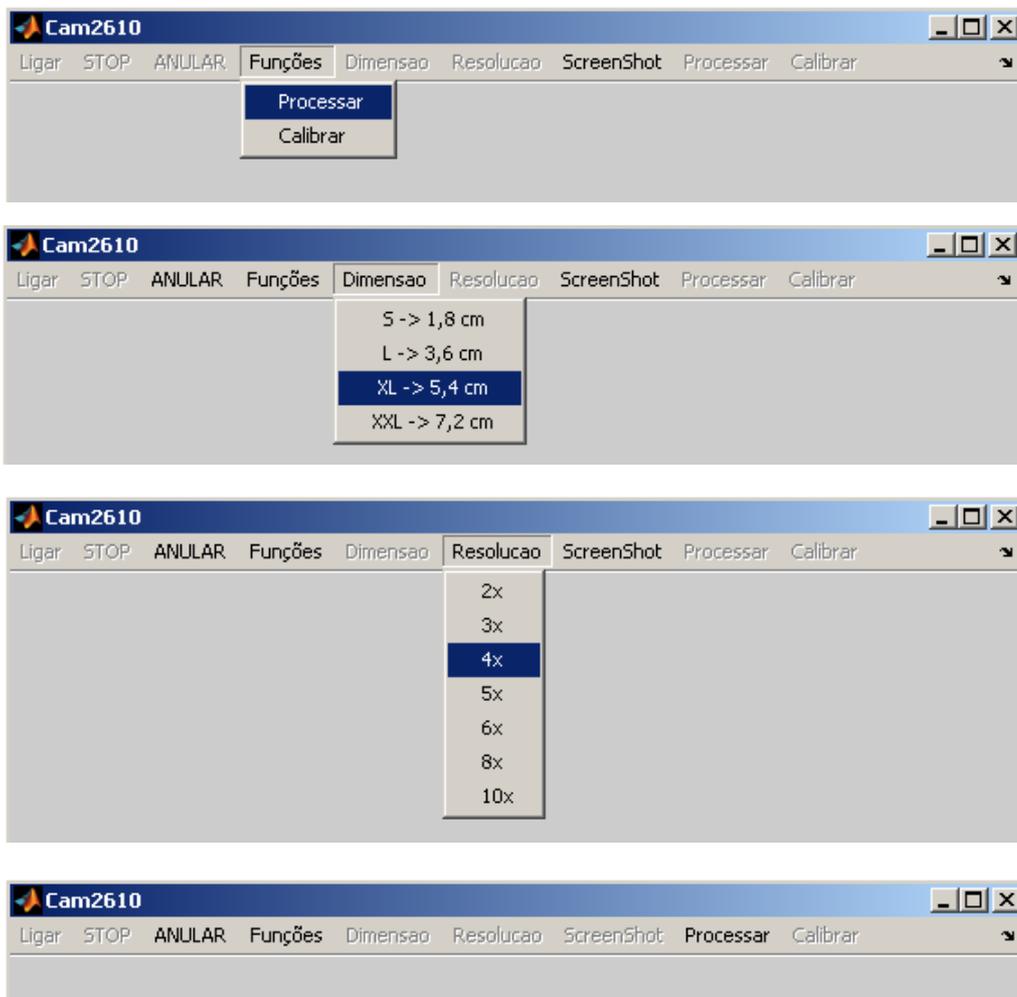


Figura 33 - Opção 'Processar' com as escolhas de dimensão de imagem 5,4cm e aumento de resolução espacial por um factor de 4.

Substituindo a imagem de calibração pelo fantoma que caracteriza a imagem de teste (Figura 34), para a qual se vão efectuar capturas de *subtepping*, selecciona-se a dimensão que teve por base a calibração e a resolução que se pretende obter.

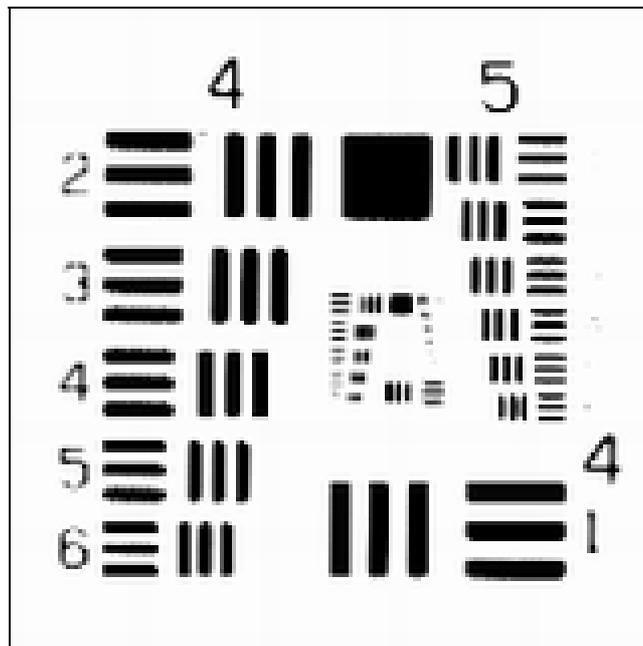


Figura 34 - Imagem de teste utilizada nos testes do sistema.

Ao seleccionar a dimensão, estamos a dar a informação do tamanho do pixel que temos na imagem. No caso do exemplo, ao introduzir-se 5,4 cm, esta distância corresponde a um tamanho de 3 mm para o pixel, pois basta dividir 5,4 cm por 18 que é a quantidade de pixels que se tem nessa distância (resolução da câmara).

A escolha do factor para o qual corresponde a melhoria de resolução espacial, introduz o valor do tamanho que a *plotter* deve efectuar entre sucessivas capturas. No seguimento do exemplo para a opção do factor '4x' está a introduzir-se um valor para o movimento de 3/4 mm: basta dividir o tamanho do pixel (3 mm) pelo número de vezes que se quer aumentar a resolução espacial (factor = 4).

A última opção, 'Processar', executa as Funções 'Scan', 'MovePlotterRelative' e 'MovePlotterAbsolute' em chamadas remotas à Biblioteca 'LibNar' da DLL de uma forma sequencial e otimizada.

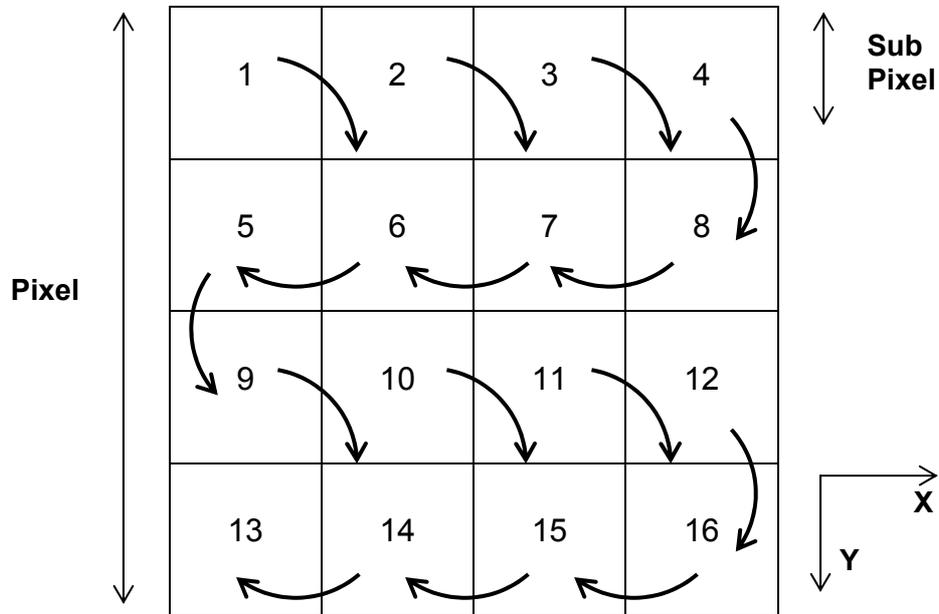


Figura 35 - Sentido do movimento e capturas das imagens de sub pixel para um exemplo de aumento de resolução espacial por um factor de 4. 'Pixel' é a área coberta por um pixel do sensor, 'Sub Pixel' representa os movimentos efectuados pela *plotter* com distâncias segundo X e Y menores que a dimensão do pixel do sensor. Os números representam os índices dos pixels da imagem de HR.

A figura 35 mostra como são adquiridas as imagens de sub pixel tendo em conta a dimensão do pixel da imagem com que a câmara foi calibrada. Cada número representa a posição da captura e as setas indicam como a *plotter* vai posicionando a câmara sobre o fantoma de acordo com o tamanho do movimento previamente escolhido e calculado.

Em cada posição são adquiridas 5 imagens, a partir das quais se calcula a média e se determina uma correcção de eficiência associada ao funcionamento do sensor, que se aborda mais adiante (secção 4.1).

Como na realidade não existe nenhum formato de imagem digital com profundidade de cor 6 bit/pixel, no tratamento e conversão de formato de imagens com 64 níveis de cinzento capturadas pelo sensor, é feita uma

normalização para 256 níveis aos valores das matrizes correspondentes a cada captura antes de se converter para o formato pretendido.

As imagens de cada posição são então gravadas com o formato 'BMP' com o nome do respectivo número da posição.

A constituição da Ferramenta de teste capaz de capturar imagens com base na técnica de *substepping* e respectivo equipamento de apoio pode ser visualizada como um todo na Figura 36.



Figura 36 - Ferramenta de teste e equipamento de apoio.

4 - Resultados

4.1 - Câmara

Após a visualização de imagens completamente escuras ou com padrões que não correspondiam ao objecto para o qual se orientava a câmara, optou-se por imprimir texto em papel com vários tamanhos de caracteres com o intuito de se poder observar melhor os padrões durante a aproximação ou afastamento do sensor ao alvo.

Nesta sequência nasce assim a primeira imagem que se pode ver na Figura 37, ela representa a letra 'N' maiúscula escrita numa folha de papel branca de tipo *Times New Roman* e de tamanho 84. As dimensões da letra são sensivelmente 2x2 cm e a distância à câmara é de 10 cm.

É bastante evidente o ruído inerente à captura, sobretudo nas zonas onde se deveria obter fundo branco. Esta característica está associada ao próprio funcionamento do sensor da câmara, que devido a um mecanismo (*Shutter*) de activação de excitação do CCD, o qual varia no tempo de acordo com os ciclos de relógio do processador, promove uma excitação distinta de pixel para pixel quando se faz a leitura do mesmo valor para os dois, dando um aspecto de actualização de valores segundo linhas periódicas diagonais (Figura 38). No fundo é como ter uma eficiência diferente de cada elemento do mesmo CCD o que se torna um problema. Outra característica é a particularidade da presença interna de um circuito de ganho automático (AGC - Automatic Gain Control) que ajusta o valor do *Shutter* para um nível de brilho máximo na ordem dos 50 [25], que comparado com o valor máximo que o sensor pode detectar, igual a 63, constitui uma diferença significativa.

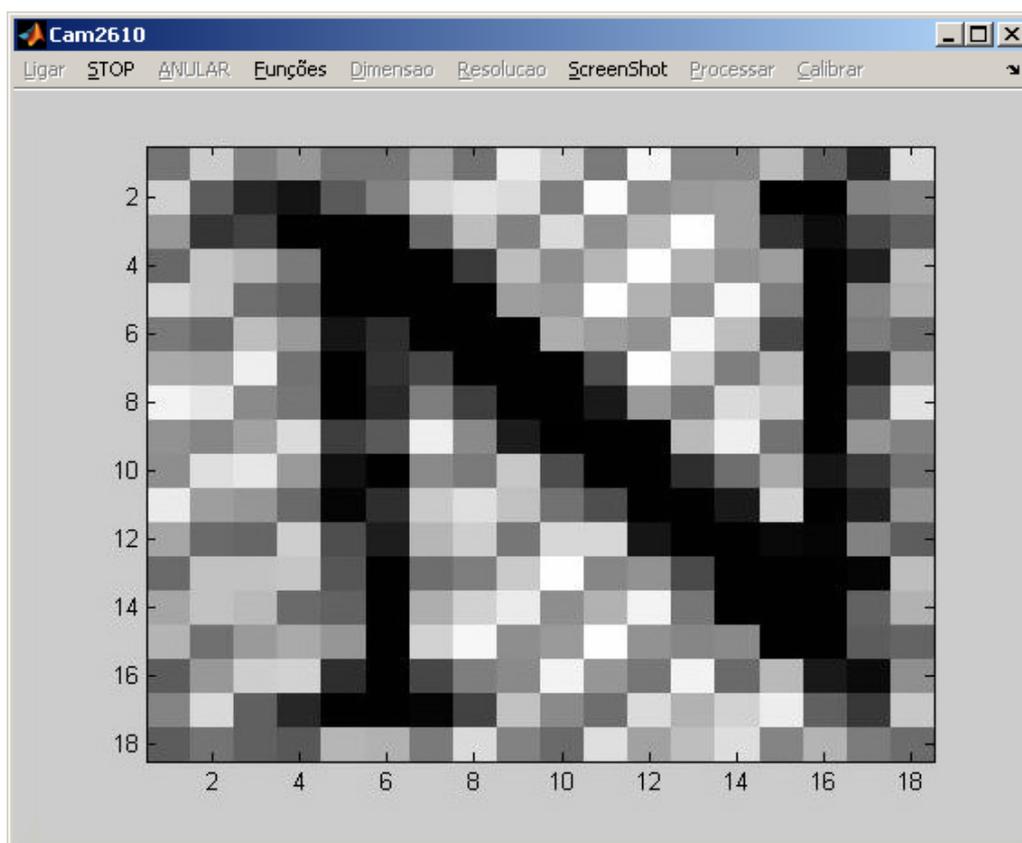


Figura 37 - Primeira imagem caracterizada capturada pela aplicação.

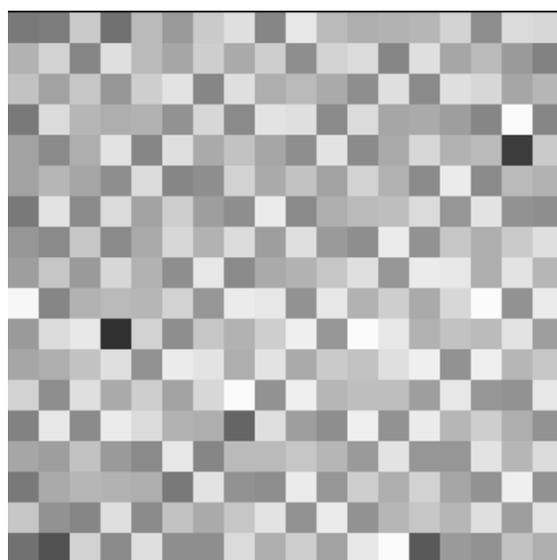


Figura 38 - Imagem de uma folha de papel branca. Actualização dos valores dos pixels segundo linhas periódicas diagonais, provocadas pelo mecanismo *Shutter*. São também evidentes diferentes eficiências entre pixels.

Para dar resposta ao problema que o *Shutter* está a introduzir é feita a média de 5 capturas da mesma imagem. Assume-se que o comportamento do *Shutter* é aleatório e que com um número infinito de imagens a média dar-nos-á a imagem original com o padrão correcto (na prática um pequeno número de imagens será satisfatório). É notório, também, que existem pixels com diferentes eficiências independentemente do *Shutter* e para colmatar esta deficiência uma matriz de eficiência é calculada.

A determinação desta matriz é efectuada usando uma imagem de fonte uniforme (folha branca) e calculando a média da intensidade de todos os pixels da imagem capturada pela câmara. Os valores de cada pixel dividido pela média constituem assim a matriz de eficiência 'efi'. Quando se adquire uma outra imagem qualquer multiplica-se o valor de cada pixel pelo inverso de 'efi', resultando na imagem corrigida com um padrão muito mais regular. Este método não é mais do que compensar a eficiência dos pixels de menor rendimento e preservar os restantes. Um exemplo pode ser visto na Figura 39 onde é efectuada a média de 5 capturas e aplicada a matriz eficiência.

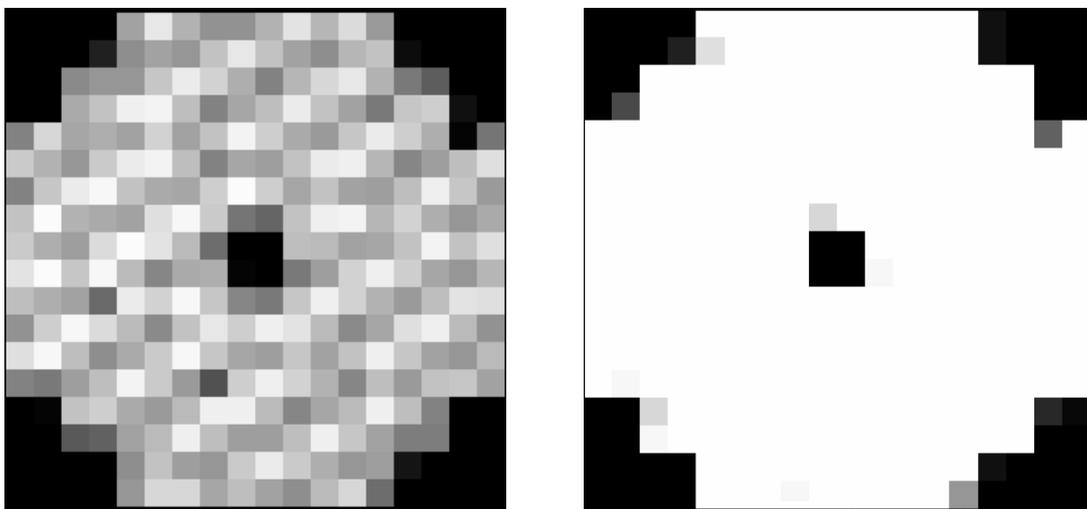


Figura 39 - Captura de uma imagem (esquerda), após tratamento (direita).

Antes de se conseguir um resultado tão satisfatório ocorreu um contratempo na conversão para um formato de imagem. Como na realidade não existe nenhum formato com profundidade de cor 6bit/pixel, o MATLAB ao

converter a matriz resultante para qualquer formato após o tratamento deparava-se com valores superiores ao limite da profundidade, aos quais atribuía valores nulos culminando num resultado pouco coerente, ver Figura 40. As profundidades de cor previstas para a maioria dos formatos de imagens são 1bit, 2bit, 4bit, 8bit e 16bit na escala de cinzentos e por isso fez-se uma normalização para 256 níveis de cinzento (8bit/pixel) antes de se proceder à gravação no formato 'BMP'.

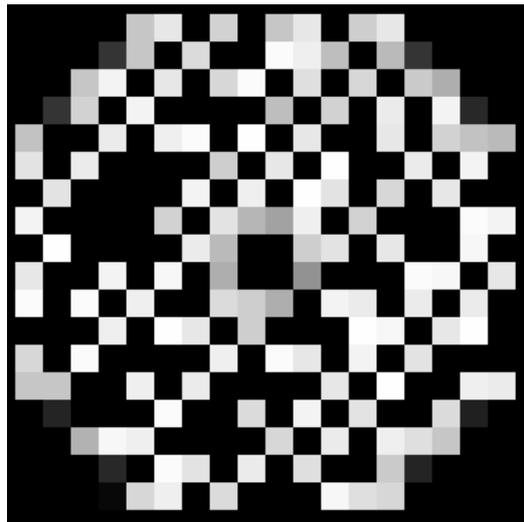


Figura 40 - Imagem resultante após tratamento sem normalização para a profundidade de cor 8 bit/pixel na escala de cinzentos (256 níveis).

Um efeito muito parecido ao apresentado na Figura 40 acontece quando a iluminação é excessiva, pelo que se procedeu a um controlo da luz incidente no objecto/imagem, de modo a que a distribuição do nível de brilho fosse o mais uniforme possível. A projecção da luz directa para o alvo não é a melhor opção, tendo-se obtido melhores resultados com uma forte iluminação difusa.

O ruído que se faz notar na imagem da direita da Figura 39 após correcção pode dever-se à falta de precisão na calibração ou aberração da lente utilizada. Pequenas diferenças na aproximação da câmara ou mesmo rotações provocadas pelo suporte durante a aquisição, são suficientes para que o resultado da captura tenha influências em determinados pixels dos seus vizinhos. De facto a calibração revelou ser a tarefa mais difícil de se fazer

depois de concluída a construção de todo o equipamento, mas extremamente importante para um resultado final desejável.

Em relação ao propósito principal da Ferramenta de teste, capturar imagens de baixa resolução com base em pequenos movimentos, o seu comportamento revelou-se bastante satisfatório. Assumindo que toda a precisão inerente a cada equipamento é cumprida, as capturas são obtidas com o número correcto relativamente ao aumento de resolução escolhida.

Para uma noção mais visível do resultado de uma captura com a câmara construída, veja-se a Figura 41 onde se mostra o fantoma utilizado para teste de dimensões 54x54 mm e a respectiva imagem adquirida.

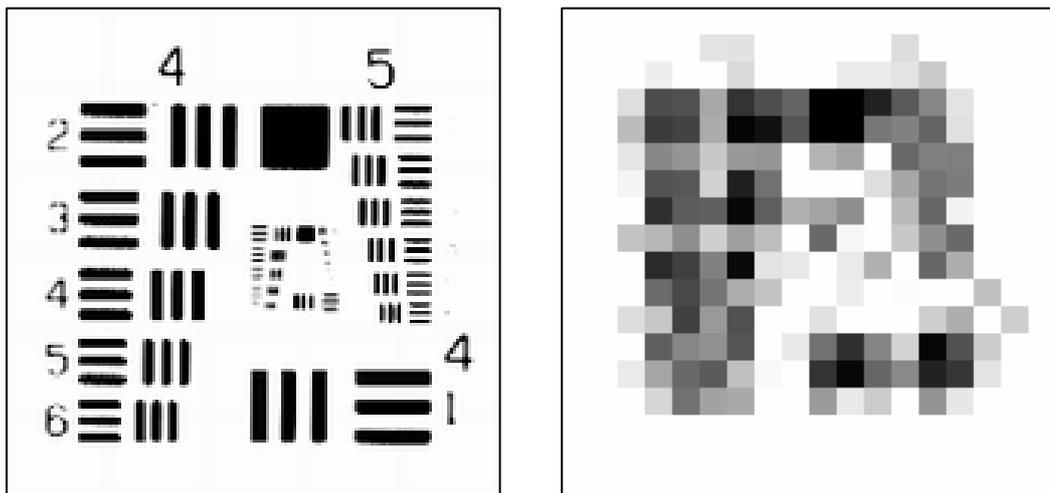


Figura 41 - Fantoma (esquerda) utilizado nos testes de aplicação do algoritmo de super-resolução e exemplo de uma imagem de baixa resolução adquirida pela câmara, numa única posição (direita). O quadrado sólido a negro do Fantoma tem de lado 7 mm. Cada pixel da imagem de baixa resolução tem a dimensão de 3 mm.

O algoritmo de super resolução foi testado com os resultados obtidos de várias experiências, para se poder apurar conclusões mais definitivas relativas ao bom funcionamento de todas as peças e relativas ao próprio algoritmo. Devido aos métodos escolhidos para a captura das imagens LR o processo torna-se um pouco lento: se a escolha do aumento de resolução for de 10x, tal implica que se façam 100 capturas (relembrando que o número de imagens necessárias é n^2 , para uma melhoria de resolução por um factor de N, como descrito na secção 2.3), que demora sensivelmente 2 horas e 15 minutos.

4.2 - Algoritmo

Perante as diferentes experiências feitas com aquisição de imagens, variando o tamanho da imagem e conseqüentemente do pixel, foram escolhidos os testes que envolviam o tamanho do pixel igual a 3 mm para aumentos de resolução por um factor de 4x e 5x.

O algoritmo apresentou uma limitação no cálculo da matriz do sistema em ambas as versões (IDL e MATLAB), quando o seu tamanho ultrapassa o valor da memória física do PC parando o processo e a reconstrução para SR.

O aumento para 5x constitui o limite para o qual é possível calcular a matriz do sistema e posteriormente a reconstrução da imagem de SR.

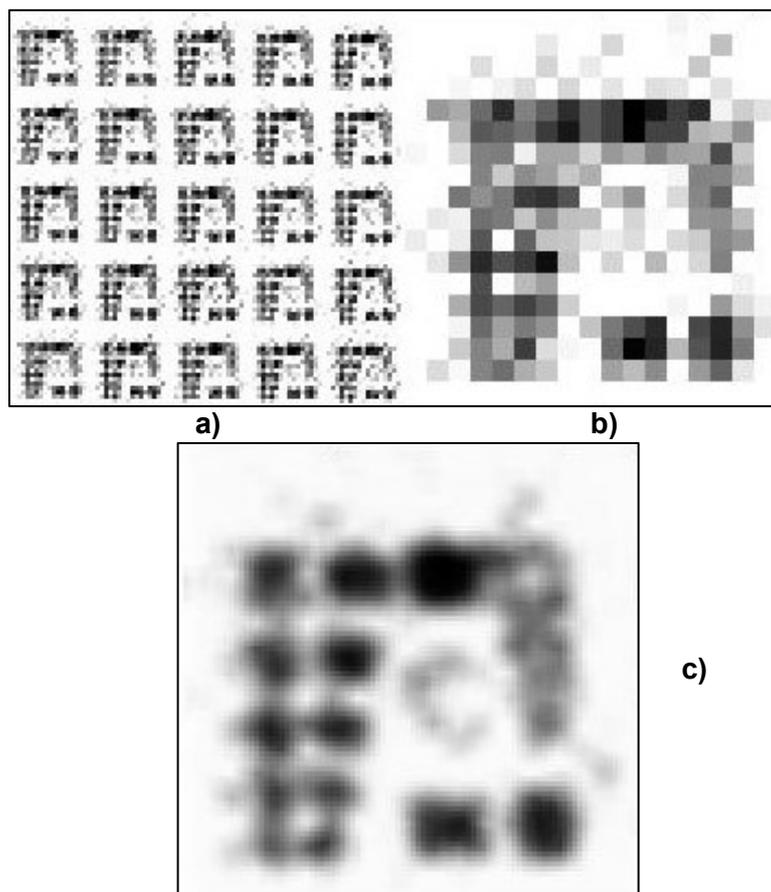


Figura 42 - Resultado da reconstrução de imagem num aumento de resolução 5x (versão IDL). a) 25 capturas necessárias para o processo, b) captura da imagem pela câmara (18x18 pixels), c) resultado da reconstrução de imagem pelo algoritmo num aumento de resolução espacial por um factor de 5x.

Os resultados apresentados na figura 40c) dizem respeito à reconstrução da imagem de teste pelo algoritmo de super resolução, versão IDL, num aumento de resolução espacial por um factor de 5x. Tendo em conta a imagem de teste original mostrada na figura 39, à esquerda, na secção 4.1, na realidade a imagem de alta resolução resultante da reconstrução converge para a original. Reconhece-se algum padrão e respectivas dimensões constituintes da imagem de teste, mas carece de detalhe na visualização das linhas mais finas. A numeração está longe de ser identificada, mesmo a de dimensão superior. Contudo, a separação entre os padrões de maior contraste e correspondentes dimensões são evidentes.

5 - Discussão e Conclusões

Comparando os resultados das imagens de baixa resolução obtidas pela câmara e as resultantes do algoritmo de super resolução apresentadas na secção 4.2, figura 40, é aparente a convergência para a imagem original. Os detalhes visualizados na imagem de super resolução reconstruída pelo algoritmo (figura 40c)) dependem do aumento de resolução que foi possível testar no algoritmo. Tendo em conta o modelo utilizado, onde se assume que o erro associado aos deslocamentos é constante, que não existe aberração da lente e não é tomado em conta o *blurring* relacionado com a captura de cada pixel da câmara, o resultado é bastante satisfatório.

O mecanismo revela por isso a capacidade e potencialidade para responder às exigências propostas inicialmente.

As imagens capturadas pela câmara e o tratamento posteriormente efectuado (secção 4.1, figura 37) mostraram ser necessários e eficazes, embora ainda seja visível ruído associado. Este pode estar a ser introduzido pela aberração das lentes da óptica utilizada (secção 3.2, figura 21a). Em relação ao próprio sensor não se pode exigir mais dada a sua simplicidade.

Os movimentos onde os deslocamentos eram de dimensão tal que a conversão para passos da *plotter* não era inteiro, constituíram uma limitação no alcance da precisão requerida afectando desde logo as contribuições para os cálculos da matriz do sistema responsável pela construção da imagem de alta resolução. Isto pode ser evitado, escolhendo factores para melhoria de resolução espacial onde os deslocamentos tenham dimensões para passos inteiros da *plotter*.

O suporte imprimiu fidelidade nas funções para as quais foi construído, embora a sua oscilação e vibração durante os movimentos possa promover

falta de rigor no posicionamento horizontal da câmara. Este aspecto é importante para uma captura perfeita. Esta perfeição assenta nos pressupostos teóricos abordados na secção 2.3 onde é apresentado o algoritmo de SR e nos pressupostos da Óptica geométrica, secção 2.1.

O posicionamento vertical da câmara para focagem e calibração da dimensão de imagens é muito preciso e de fácil utilização, devido às funções da calha e da mola do suporte. É possível com estes fazer deslocar a haste onde está fixa a câmara, em movimentos pequenos e contínuos de modo a que a calibração seja a mais minuciosa possível. A actualização da imagem por ser feita apenas a cada 5 s, constituiu a única contrariedade encontrada neste processo. Por vezes, o número de ajustes efectuados para calibrar a imagem durante o compasso de espera era exagerado, tornando o processo mais difícil e lento.

O *blurring* associado ao resultado da reconstrução de imagem no aumento de resolução (secção 4.2, figura 40c) está relacionado com erros de posição das imagens de baixa resolução. Será necessário apurar qual o erro mínimo permitido pelo algoritmo para esses movimentos e relacioná-lo com o erro que a *plotter* possa estar a introduzir. Outra possível razão é o algoritmo não possuir um modelo de *blurring* associado à câmara construída e que é utilizada na captura das imagens de baixa resolução. Assim sendo, seria útil fazer-se essa análise para melhoria de resultados, no que diz respeito à reconstrução da imagem de alta resolução.

Considerando as dimensões do quadrado sólido negro do fantoma (7mm) da figura 39 da secção 4.1, é clara a resposta do sensor a essa dimensão do quadrado (na mesma figura 39 à direita). Sabendo que o pixel dessa imagem de baixa resolução tem a dimensão de 3 mm, facilmente se observa a definição do quadrado por quatro pixels, na mesma posição na imagem, resultando numa aresta de 6 mm (bem próximo dos 7 mm originais). Por outro lado, é evidente também a influência desse quadrado na imagem de baixa resolução nos pixels vizinhos (*blurring*).

Como considerações e tarefas futuras pode ainda acrescentar-se que seria desejável testar uma outra lente de melhor qualidade e menor curvatura e caracterizar novamente as imagens.

O suporte pode ser alterado de maneira a que se garanta uma posição horizontal da câmara constante, ou menos oscilante. Uma solução seria fixar o suporte à estrutura da *plotter* e movimentar com a cabeça o objecto/imagem, visto ser mais leve e promover uma oscilação menor.

A calibração das imagens pode ser também melhorada com a introdução de uma operação na aplicação gráfica que permite movimentar a cabeça da *plotter* passo a passo nos seus eixos, ajudando o suporte na sua função.

Em relação à integração do sistema, seria interessante tentar efectuar a comunicação com o sensor por uma porta série: é mais fácil encontrar um PC que tenha disponíveis uma porta série e uma paralela em vez de duas portas paralelas, promovendo a versatilidade do mecanismo.

Referências bibliográficas

[1] - Boggione, G. A.; Fonseca, L.M.G. "Restoration of Landsat-7 Images". In: *International Symposium on Remote Sensing of Environmental (ISRSE)*, Hawaii, Nov. 2003.

[2] - Battiato, S.; Gallo, G.; Stanco, F. "A locally adaptive zooming algorithm for digital images". *Image and Vision Computing*, v. 1, n. 20, p. 805-812, 2002.

[3] - Candocia, F. M.; Principe, J. C. "Super-resolution of images based on local correlations". *IEEE Transaction on Neural Networks*, v. 10, n. 2, p. 372-380, Mar 1999.

[4] - Candocia, F. M.; Principe, J. C. "Super-resolution of images based on local correlations". *IEEE Transaction on Neural Networks*, v. 10, n. 2, p. 372-380, Mar 1999.

[5] - Rajan, D.; Chaudhuri, S. "Data fusion techniques for super-resolution imaging". *Information Fusion*, v. 3, p. 25-38, 2002.

[6] - Park, S. C.; Park, M. K.; Kang, M. G. "Super-Resolution Image Reconstruction: A Technical Overview". *IEEE Signal Processing Magazine*. n. 03. p. 21-36. 2003.

[7] - Freeman, W. T.; Pasztor, E. C.; Carmichael, O. T. *Learning Low-level Vision*. Cambridge, MA: MERL, Mitsubishi Electric Research Labs, 2000. 45 p. (TR-2000-05a).

[8] - http://www.ibili.uc.pt/~nuno/API_2007-2008/Aula_08.pdf

[9] - <http://www.all-in-one.ee/~dersch/interpolator/interpolator.html>

[10] - Xueting Liu, Daojin Song, Chuandai Dong, and Hongkui Li; “MAP-Based Image Super-Resolution Reconstruction”. International Journal of Computer Science and Engineering Volume 2 Number 3.

[11] - F. J. Caramelo, G. Almeida, L. Mendes, N. C. Ferreira; “Study of an iterative super-resolution algorithm and its feasibility in high-resolution animal imaging with low-resolution SPECT cameras”

[12] - G. Almeida; “Melhoria da resolução em SPECT através de pequenos movimentos”

[13] - Peleg, S., Keren, D., Schweitzer, L. Improving Image Resolution Using Subpixel Offsets (Motion). Pattern Recognition Letters, 1987, 223-226.

[14] - Peleg, S., Irani, M., Improving Resolution by Image Registration. Graphical Models and Image Processing. vol. 53, nº 3, 1991, p. 231-239.

[15] - Gross, D., Ur, A. Improved Resolution from Subpixel Shifted Pictures. CVGIP- Graphical Models and Image Processing. Vol. 54, Nº 2, Março, 1992, pp. 181-186.

[16] - Hathaway, D., Meyer P. NASA: Video Enhancer For Law Enforcement & More. Advanced Imaging: Solutions for the Electronic Imaging Professional; Maio, 1999, pp. 7.

[17] - Kang, M.J, et al. The Digital Image Acquisition of High-Resolution by Enhancement the Multiple Images. ASPRS-Proceedings - Launching the Geospatial Information Age, Washington, 2000.

[18] - Wiman, H. Improvement of Digital Resolution by Oversampling. ISPRS Commision II, Washington, 1992.

[19] - Joyeux, L., et al. Reconstruction of degraded image sequences. Application to film restoration. Image and Vision Computing. N° 19, France, 2001, pp. 503-516.

[20] - Baker S., Kanade T. Super-Resolution Optical Flow. [s.l.]: CMU-RI-TR-99-36, 1999.

[21] - R. C. Gonzales, R. E. Woods. Digital Image Processing, second edition. New Jersey, 2002, pp. 34-75.

[22] - N. Ferreira. Apontamentos da disciplina de 'Análise e Processamento de Imagem'. IBILI, 2007/2008 aula 8.

[23] - Edward C. *Print Unchained: Fifty Years of Digital Printing: A Saga of Invention and Enterprise*. West Dover, VT: DRA of Vermont, Inc. 2000.

[24] - Durda IV, Frank. "Centronics and IBM Compatible Parallel Printer Interface Pin Assignment Reference", 2004.

[25] - Agilent Technologies. Agilent ADNS-2610 Optical Mouse Sensor data sheet, 2004. (Online - <http://cp.literature.agilent.com/litweb/pdf/5988-9774EN.pdf>).

[26] - Hayit Greenspan. "Super-Resolution in Medical Imaging", Faculty of Engineering, Biomedical Engineering Department, Tel-Aviv University, Tel-Aviv, Israel, 2008.

[27] - M.Margarida Costa, Maria J. Almeida. "Fundamentos de Física", Livraria Almedina. Coimbra 1993, pp 629-710.

[28] - William H. Press, Saul A. Teukolsky, William T. Vetterling, Brian P. Flannery. "Numerical Recipes in C - The Art of Scientific Computing", second edition, Cambridge University Press, 1997.

Sun Microsystems. C Programmer's Guide, Revision A, Maio 1988.

Brian V. Kernighan, Dennis M. Ritchie. The C Programming Language, New Jersey, 1978.

Alan R. Feuer. The C Puzzle Book, New Jersey, 1982.

Graphtec Corporation. X-Y Plotter MP3000 Series user's manual, Tokyo - Japan, 1988.

ANEXOS

Ficheiro 'Cam2610Move.m' - MATLAB

Ficheiro 'LibNar.dll' - Linguagem C