

Processamento de Imagem e Biometria

Trabalho Prático 1

Mihail Ababii, 46435@alunos.isel.ipl.pt
Rodrigo Silva, A41429@alunos.isel.pt
Antonio Flavius Kronemberger, A50073@alunos.isel.pt

03 Dezembro de 2021

Análise estática

1 Exercício 1

1.a Explain, in detail, the DIP actions taken by the four *.m files on the MATLAB demos named Demo1.zip and Demo2.zip, available on the Moodle page.

DEMO 1

O Documento `read_image.m` lê uma imagem, a partir da qual obtêm diversas informações. Numa primeira fase, obtêm o tamanho espacial da imagem, bem como cria uma janela para apresentar-la, seu histograma e a entropia da imagem. Numa segunda fase, obtêm e apresenta na consola detalhes mais específicos, sendo estes a energia, a potência, o valor médio e o valores mínimo e máximo.

O Documento `read_image_color.m` lê uma imagem, a partir da qual obtêm as respectivas componentes RGB em imagens separadas, apresentando-as numa janela. Este módulo também guarda a imagem em formato JPEG. Numa terceira fase converte a imagem para a escala de cinzento, apresentando numa nova janela a imagem original, assim como a imagem na escala de cinzento.

O Documento `read_image_color_v2.m` lê uma imagem, a partir da qual obtêm as componentes RGB em separado e apresenta as mesmas numa figura. Este módulo também cria duas janelas, na primeira são apresentadas a imagem original e a imagem com as componentes R e G trocadas. Na segunda a imagem original e a imagem com as componentes R e B trocadas. Podemos observar que são apresentados os valores da energia das diversas componentes RGB, na consola.

O Documento `test_images.m` lê uma imagem a partir da qual cria 3 janelas. Na primeira janela apresenta a imagem original, na escala de cinzento e na escala binária. Na segunda janela apresenta as versões na escala de cinzento e na escala binária, assim como seus respectivos histogramas. Na última janela apresenta os histogramas das diversas componentes RGB.

DEMO 2

O modulo `arithmetic_operations.m` lê duas imagens, às quais aplica diversas funções aritméticas. Nas diversas janelas, são apresentadas as imagens originais, o resultado da aplicação das funções e os seus respectivos histogramas. Nas primeiras três janelas são apresentadas a soma, subtração e a multiplicação entre as duas imagens. Na quarta janela é apresentado o aumento do brilho, adicionando 50 a todos os pixels de uma imagem. Na quinta janela é apresentado a alteração o contraste, multiplicando 2 a todos os pixels de uma imagem.

O modulo `logical_operations.m` lê duas imagens, aplica as várias operações lógicas e apresenta numa janela o resultado de cada operação e as imagens originais. Estas operações consistem em AND, OR, XOR e NOT.

O modulo `optical_ilusion.m` apenas converte uma imagem para a escala de cinzento e apresenta o resultado numa janela, adicionando a possibilidade visualizar a intensidade de cada pixel, de modo a melhor visualizar a ilusão de cores.

O modulo `optical_ilusion2.m` gera 3 imagens 256x256. O conteúdo de cada imagem é gerado de forma computacional: a primeira imagem contém um quadrado cinzento no centro com valor 100 e à sua volta valor 10, a segunda imagem contém um quadrado cinzento no centro com valor 100 e à sua volta valor 128, a terceira imagem contém um quadrado cinzento no centro com valor 100 e à sua volta valor 250. Após realizar a geração das imagens, são criadas duas janelas: uma para apresentar as imagens geradas, outra para apresentar os histogramas das diferentes imagens, de modo a verificar que o quadrado central mantém o mesmo valor.

2 Exercício 2

Nos testes realizados às 3 imagens [`boat.tif`, `mandrill.tif`, `squares.tif`], usando o código desenvolvido em java para ajustar o brilho e contraste das imagens, chegou-se às seguintes conclusões: alterar o brilho implica um "shift" ao histograma da imagem, podendo haver saturação na parte inferior ou superior dependendo de quanto se ajusta o mesmo; ao alterar o contraste faz com que o histograma da imagem seja mais ou menos disperso, podendo haver saturação nos valores superiores.

3 Exercício 3

3.a For each image and its negative version, compute the brightness, contrast, and entropy values. Present the histogram of each image. Comment on these results.

Ao realizar a computação das várias imagens para o seu inverso, verificou-se que o histograma das imagens inverte horizontalmente (valores baixos passam para altos, e valores altos passam para baixos). Verificou-se também que o contraste e a entropia não foram alterados, no entanto, o brilho foi alterado especialmente para as imagens: `Sat.png` e `iris.png`. Para realizar esta alteração, utilizou-se a função *"imcomplement"* que aplica uma lookup table inversa (função inversa) à original.

3.b For each original image, apply an adequate operation to improve its contrast, using an intensity transform with a lookup table. Describe the operation chosen for each image and show the resulting image, its brightness, contrast, entropy values and its histogram. Report the experimental results. Comment on these results.

Ao realizar a computação das várias imagens para ajustar o seu contraste, verificou-se que imagens que tinham inicialmente baixo contraste. O contraste teve elevada melhoria, imagens que inicialmente tinham um contraste alto, notou-se pouco a melhoria. A função *"imadjust"* satura 1% dos valores da imagem nos extremos (mais altos e mais baixos) da lookup table, ajusta o contraste para abranger o mais possível de valores do espectro. Também podemos observar, ao aplicar este ajuste a entropia diminui ligeiramente.

3.c For each image, apply the histogram equalization and the histogram specification techniques to improve its contrast. Show the resulting image and its histogram. Report the experimental results. Comment on these results.

Ao aplicar a técnica de equalização do histograma, podemos observar que geralmente é boa alternativa, visto que melhora o contraste e/ou o brilho sem comprometer muito a qualidade de imagem e sua entropia. Em contrapartida, ao utilizar a técnica de especificação de histograma, o resultado passa a depender da imagem utilizada como referência. Devemos utilizar uma imagem referência com qualidades específicas pretendidas, de modo a que o resultado da imagem original com o histograma de referência, equalize o mais possível na imagem resultante. É importante salientar que ao utilizar uma imagem referência igual a própria imagem de entrada, nada será alterado.

4 Exercício 4

4.a For each image noisy/distorted image: identify the types of noise/distortion; propose a technique to restore it to its original contents. Explain the proposed technique.

Cada série de imagens identificadas no directório sofre com algum tipo de ruído ou distorção. Na lista que se segue, para cada série de imagens, são expostos os problemas identificados, assim como um possível método para restaurar as mesmas:

- Imagem face.png
 - Ruído/distorção: Smoothing e problemas com contraste e brilho.
 - Técnica utilizada para restauração: Sharpening e reajuste de contraste e brilho.
- Imagem face_thermogram.png
 - Ruído/distorção: Salt and pepper e problemas de contraste e brilho.
 - Técnica utilizada para restauração: Median filter e reajuste de contraste e brilho.
- Imagem fingerprint.png
 - Ruído/distorção: Ruídos periódicos padronizados.
 - Técnica utilizada para restauração: Aplicação de filtros frequency-based.
- Imagem iris.png
 - Ruído/distorção: White noise e gaussian noise.
 - Técnica utilizada para restauração: Mean filter.
- Imagem squares.gif
 - Ruído/distorção: Salt and pepper, rotação e inversão de cor.
 - Técnica utilizada para restauração: Median filter, reajuste de rotação e inversão de cor.
- Imagem xray.png
 - Ruído/distorção: Smoothing e ajuste de contraste e brilho.
 - Técnica utilizada para restauração: Laplacian filter e histogram specification.

4.b Apply the proposed techniques on all the noisy/distorted images of the NoisyImages.zip file. Compare it against the original version using the measures of image comparison, found at the end of this document. Comment on these results.

Após a obtenção de resultados, concluímos que imagens com ruídos mais acentuados e distorções, perdem certas informações da imagem, tornando o restauro desta mais difícil, por vezes impossível. Desta forma, o seu nível de restauro não é exactamente o mesmo da imagem original. No entanto, imagens com pouco ruído, são processadas mais facilmente, sendo possível a recuperação de informação anteriormente perdida, obtendo um resultado razoável.

Em adição, ao utilizar os métodos de comparação propostos, Mean Squared Error (MSE) e Mean Absolute Error (MAE), é possível observar que por mais que pareça estar sem ruídos a imagem recuperada, na realidade, é apenas uma aproximação do original. Pode-se observar também, que os resultados obtidos pelo MSE, são consideravelmente mais elevados que com o MAE. Isto acontece pois como o MAE apenas considera os valores absolutos de erro, atribuindo o mesmo peso a todos os erros, enquanto que o MSE ao considerar o quadrado do erro, irá atribuir mais peso aos maiores erros.

5 Exercício 5

5.a Develop a CAPTCHA application, that produces grayscale and color images on both JPEG and PNG format, with different resolutions and different levels of challenge/difficulty to the user. Describe your proposed solution.

A solução proposta começa por criar uma imagem com uma determinada resolução espacial, calculada dinamicamente de acordo com determinados parâmetros internos. A geração do seu conteúdo, envolve 2 ciclos. Um para as linhas e outro para as colunas. Se estiver na 1ª linha, adiciona as "labels" das colunas, se estiver na 1ª coluna adiciona as "labels" das linhas. Nas restantes, gera um valor aleatório do intervalo $[0..999]$, escolhe uma cor para apenas um caractere e adiciona a string à imagem.

5.b Report five different images generated with the application. Comment on the results.

Nas várias imagens geradas, foi verificado que devido à cor utilizada como "background", é difícil de ver certos caracteres coloridos, no entanto esses caracteres podem ser facilmente reconhecidos na escala de cinzento. Também se verifica que há caracteres que são difíceis de ver nas duas situações. Uma solução para estes dois casos, é usar outro método de conversão da imagem colorida para cinzento, por exemplo dar valores de peso diferentes para cada componente na converção para a escala de cinzento.

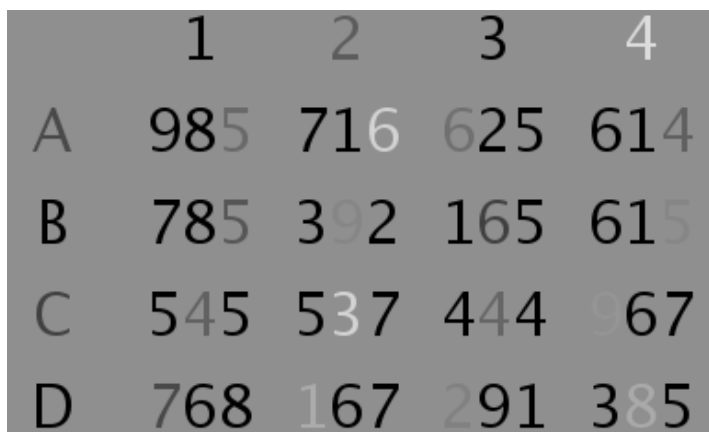


Figura 1: RGB 4x4, PNG

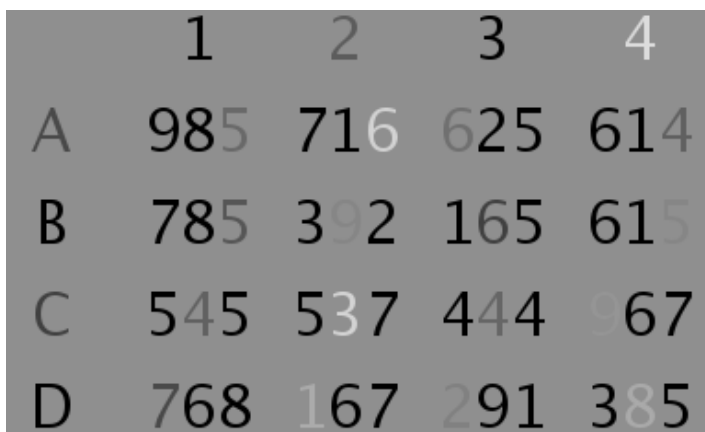


Figura 2: Gray 4x4, PNG

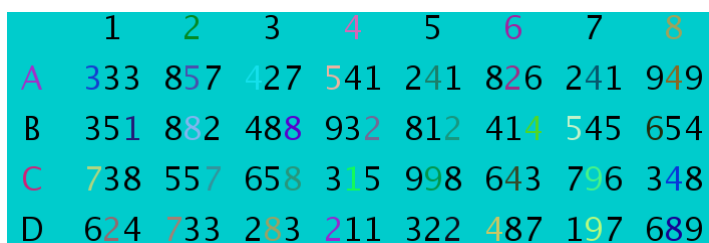


Figura 3: RGB 4x8, PNG

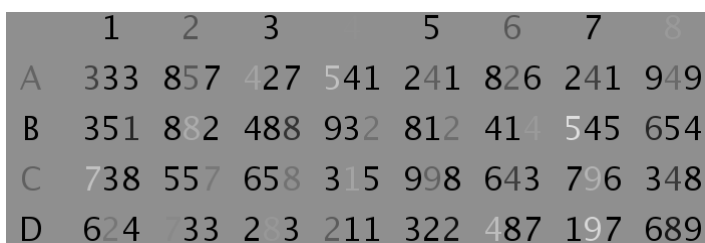


Figura 4: Gray 4x8, PNG

	1	2	3	4
A	161	859	298	191
B	939	937	955	455
C	636	994	289	377
D	852	591	919	623
E	943	868	536	983
F	487	176	329	363
G	815	366	638	465
H	648	687	215	823

Figura 5: RGB 8x4, PNG

	1	2	3	4
A	161	859	298	191
B	939	937	955	455
C	636	994	289	377
D	852	59	919	623
E	943	868	536	983
F	487	176	329	363
G	815	366	638	465
H	648	687	215	823

Figura 6: Gray 8x4, PNG

6 Exercício 6

A Completely Automated Public Turing test to tell Computers and Humans Apart (CAPTCHA) can be defined as a type of question-response test to determine whether or not the user is human, <http://www.captcha.net/>.

6.a

A solução proposta começa por criar uma imagem com uma determinada resolução espacial, calculada dinamicamente de acordo com determinados parâmetros internos. O conteúdo da imagem gerada, terá 1 linha com "max-Letters" caracteres. Dependendo da dificuldade [0..2], é adicionado ruído e/ou distorção à imagem. É possível gerar a mesma imagem passando uma "seed" por parâmetro. No final, as imagens geradas serão guardadas de acordo com o nome: "captcha_<dificuldade>_<tipo de cor>" em PNG e JPEG.

6.b

Nas várias imagens geradas, foi verificado que a adição de ruído em jpeg torna o texto um pouco mais difícil de ler, devido à compressão usada. Com distorção horizontal e vertical, certas letras poderão ser confundidas umas com as outras, por exemplo: 'G' com o 'C', 'Q' e 'J' por um 'O' e 'J', como se pode verificar através das imagens recolhidas.

ILGCWQJB

Figura 7: RGB com dificuldade 0, JPEG

ILGCWQJB

Figura 8: RGB com dificuldade 0, PNG

ILGCWQJB

Figura 9: RGB com dificuldade 1, JPEG

ILGCWQJB

Figura 10: RGB com dificuldade 1, PNG



Figura 11: RGB com dificuldade 2, JPEG

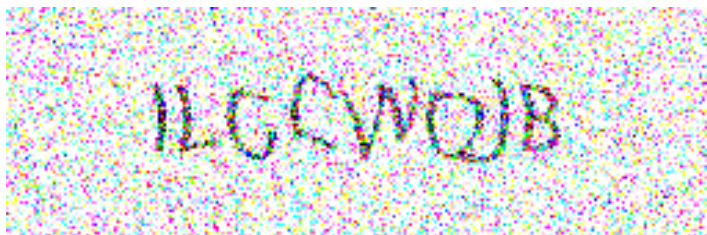


Figura 12: RGB com dificuldade 2, PNG

ILGCWQJB

Figura 13: Gray com dificuldade 0, JPEG

ILGCWQJB

Figura 14: Gray com dificuldade 0, PNG

ILGCWQJB

Figura 15: Gray com dificuldade 1, JPEG

ILGCWQJB

Figura 16: Gray com dificuldade 1, PNG

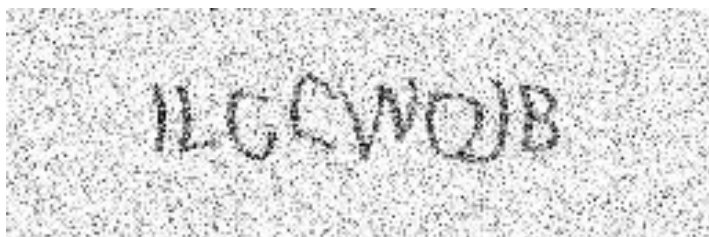


Figura 17: Gray com dificuldade 2, JPEG

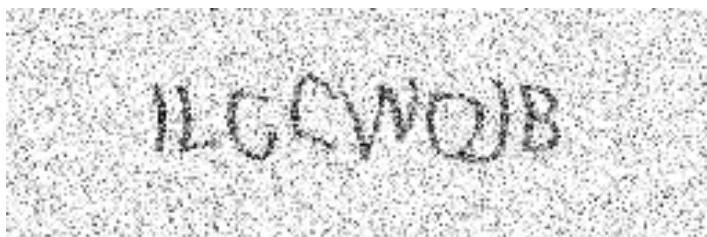


Figura 18: Gray com dificuldade 2, PNG