

Project 1 Questions

Instructions

- 4 questions.
- Write code where appropriate.
- Feel free to include images or equations.
- **Please use only the space provided and keep the page breaks.** Please do not make new pages, nor remove pages. The document is a template to help grading.
- If you really need extra space, please use new pages at the end of the document and refer us to it in your answers.

Questions

Q1: Explicitly describe image convolution: the input, the transformation, and the output. Why is it useful for computer vision?

A1: Input: Matriz da imagem que será transformada.

Transformation: (Matriz da imagem * filtro selecionado), o filtro pode ser uma matriz 3x3 ou 5x5, onde cada pixel da imagem principal é multiplicado pelo valor resultante do filtro multiplicado com os valores dos pixels da imagem principal, o valor final dos pixels ao redor é somado gerando um novo pixel para a imagem principal.

Output: Matriz da imagem final.

Pode ser utilizado para realçar bordas, aguçar a imagem, desfocar, destacar relevo e detectar bordas.

Q2: What is the difference between convolution and correlation? Construct a scenario which produces a different output between both operations.

A2: Convolução é uma correlação aplicada com o filtro girado a 180 graus. Isso não faz diferença, se o filtro é simétrico, como um gaussiano ou um laplaciano, mas isso faz uma grande diferença, quando o filtro utilizado não é simétrico, por exemplo um filtro que utilizada derivada.

Q3: What is the difference between a high pass filter and a low pass filter in how they are constructed, and what they do to the image? Please provide example kernels and output images.

A3: Os filtros denominados passa baixas eliminam altas frequências, são utilizados para eliminar ruídos em imagens. O ruído é uma fonte de alta frequência e o efeito produzido é uma desfocalização caracterizada por uma imagem borrada. Esta desfocalização depende das dimensões do filtro que está sendo utilizado, pois quanto maior forem as dimensões do filtro, maior será a desfocalização da imagem final.

Já os filtros passa altas, são utilizados para eliminar feições de baixa frequência e para realçar feições de alta frequência a imagem. O tamanho do filtro utilizada influencia diretamente no resultado final da imagem. Quanto menor forem as dimensões do filtro utilizado, menos detalhes serão realçados.

Q4: Explain the code in file gen-hybrid-image-fft. What each line is supposed to do? What does the function H do?

A4: Código comentado abaixo, pode ser visualizado acessando o arquivo gen-hybrid-image-fft.

```
1 function [hybrid_image, low_frequencies, high_frequencies]
   = gen_hybrid_image_fft( imagel, image2,
   cutoff_frequency )
2 pkg load image;
3
4 low_frequencies = []; %imagem com as componentes em baixa
   frequencia
5
6 figure, imshow(imagel); % print da imagem1
7
8 b = padarray(imagel, size(imagel), "zeros", "post"); %
   faz o padarray da imagem1 com 0
9
10 c = im2double(b(:, :, 1:3)); % transforma as componentes em
   double
11
12 d = fft2(c); % realiza a transformada de fourier 2D
13
14 d = fftshift(d); % Desloca a componente DC para o centro
   da imagem
15
16 figure, imshow(uint8(abs(d))); %print da imagem resultado
   apos a transformada d
17
18 [n m o] = size(c); %faz um vetor auxiliar do mesmo
   tamanho da imagem
19
20 h = zeros([n,m]); %cria uma matriz de zeros
21
22 for i = 1:n
23     for j = 1:m
24         h(i,j) = H(i,j,size(c),cutoff_frequency); %calcula o
           filtro e salva em h
25     end
26 end
27
28 figure, imshow(im2uint8(h)); %print da imagem de
   resultado h
29
```

```
30 g = d.*h; % multiplicacao de cada componente da imagem
    transformada com o filtro
31
32 g = ifftshift(g); % faz o shift de g
33
34 at = ifft2(g); % realiza a transformada inversa de
    fourier 2D
35
36 [x y o] = size(image1); % cria um vetor auxiliar do mesmo
    tamanho da imagem1
37
38 atc = at(1:x,1:y, :); % salva a imagem final em atc
39
40 figure, imshow(low_frequencies = atc); % print da imagem
    resultado em baixa frequencia low_frequencies
41
42 high_frequencies = []; %imagem com as componentes em alta
    frequencia
43
44 figure, imshow(image2); % print da imagem2
45
46 b = padarray(image2, size(image2), "zeros", "post"); %
    faz o padarray da imagem2 com 0
47
48 c = im2double(b(:, :, 1:3)); % transforma as componentes em
    double
49
50 d = fft2(c); % realiza a transformada de fourier 2D
51
52 d = fftshift(d); % Desloca a componente DC para o centro
    da imagem
53
54 figure, imshow(uint8(abs(d))); %print da imagem resultado
    apos a transformada d
55
56 [n m o] = size(c); %faz um vetor auxiliar do mesmo
    tamanho da imagem
57
58 h = zeros([n,m]); %cria uma matriz de zeros
59
60 for i = 1:n
61     for j = 1:m
62         h(i,j) = H(i,j,size(c),cutoff_frequency); %calcula o
            filtro e salva em h
63     end
64 end
```

```
65
66 h = imcomplement(h); %calcula a inversa do filtro obtido
    em h
67
68 g = d.*h; % multiplicacao de cada componente da imagem
    transformada com o filtro
69
70 g = ifftshift(g); % faz o shift de g
71
72 at = ifft2(g); % realiza a transformada inversa de
    fourier 2D
73
74 [x y o] = size(image1); % cria um vetor auxiliar do mesmo
    tamanho da image1
75
76 atc = at(1:x,1:y, :); % salva a imagem final em atc
77
78 figure, imshow(high_frequencies = atc); % print da
    imagem resultado em baixa frequencia high_frequencies
79
80 % Combine the high frequencies and low frequencies
81
82 hybrid_image = abs(low_frequencies + high_frequencies); %
    criacao da imagem hibrida
```