

Visão computacional - 2020/2

Relatório PS1

Tiago Araújo Mendonça

[Código fonte](#)

Questão 1.1

Nesta questão usei o tutorial da biblioteca OpenCV com primeiros passos para abrir uma imagem. Nesta questão é opcional passar um parâmetro na linha de comando para especificar a imagem utilizada, se não for usado, uma imagem padrão é utilizada.

Questão 1.2

Utilizei a função `split` do OpenCV para dividir a imagem em três planos (R, B, G), e utilizei a função `calcHist` do OpenCV para calcular o histograma, utilizei o tamanho 256 para o histograma pois existem 256 valores possíveis para cada R, G ou B. Para exibir os histogramas, utilizei a biblioteca `matplotlib.pyplot`. Nesta questão é opcional passar um parâmetro na linha de comando para especificar a imagem utilizada, se não for usado, uma imagem padrão é utilizada.

Questão 1.3

Nesta questão uma cópia da imagem é exibida até que o usuário pressione ESC. Foi definida uma função de callback para eventos de mouse, chamada `evento_mouse`, esta função faz o seguinte sempre que o ponteiro do mouse se mexe:

1. É obtido o pixel central que corresponde ao pixel onde está o cursor do mouse. A posição deste pixel é exibida no terminal
2. Obtem um conjunto de pixels de tamanho 11x11 com o pixel central como referência
3. Obtem os valores RGB para o pixel central e sua intensidade respectiva. Estes valores são exibidos no terminal.
4. São calculados a média e o desvio padrão do conjunto de pixels do item 2 utilizando a biblioteca `numpy`. Estes valores são exibidos no terminal
5. É exibido um retângulo verde de tamanho 13x13 que representa a borda EXTERNA do conjunto de pixels do item 2

Nesta questão é opcional passar um parâmetro na linha de comando para especificar a imagem utilizada, se não for usado, uma imagem padrão é utilizada.

Questão 1.4

Janelas homogêneas possuem cores parecidas em toda a janela, com intensidades parecidas, isso acontece em superfícies de um mesmo objeto por exemplo. Áreas não-homogêneas possuem cores bem diferentes, como por exemplo janelas que possuem a divisão entre dois objetos distintos. Janelas homogêneas possuem um histograma mais uniforme, e com baixa variância.

Questão 2

Nesta questão um vídeo é aberto e utilizamos a função `vid.read()` do OpenCV para iterar sobre cada quadro (frame) do vídeo. Para cada um dos quadros é calculado:

1. O contraste; utilizando uma função implementada manualmente conforme a fórmula disponível no livro da disciplina
2. A média do quadro; utilizando a função `average` do numpy
3. A variância do quadro; utilizando a função `var` do numpy

Para cada quadro, estes três valores são adicionados em um vetor separado. A execução destes cálculos é demorada, então o terminal sempre exibe quantos quadros já tiveram esses valores analisados. Ao terminar de obter esses valores para todos os quadros, estes três vetores são normalizados para terem a mesma média e desvio padrão do que o vetor de médias (escolhido arbitrariamente). A normalização foi implementada manualmente seguindo as fórmulas disponíveis no livro da disciplina.

São calculadas as três diferenças L1 (contraste-média; contraste-variância; média-variância) e elas são exibidas no terminal. Por fim, é exibido um gráfico com os três vetores normalizados, a legenda para o gráfico é exibida no terminal. Nesta questão é opcional passar um parâmetro na linha de comando para especificar a imagem utilizada, se não for usado, uma imagem padrão é utilizada.

Questão 3.a

Utilizei as funções `fft2` e `fftshift` da biblioteca numpy para realizar a transformada, e calculei a fase e amplitude também com esta biblioteca. Para fazer a transformada inversa, utilizei a fase da imagem 1 e amplitude da imagem 2, e vice-versa, por isso são exibidos dois resultados. É possível perceber que a fase contribui bem mais na reconstrução da imagem. Nesta questão é opcional passar dois parâmetros na linha de comando para especificar as imagens utilizadas, se não forem passados, ou se for passado apenas um parâmetro, imagens padrão serão utilizadas.

Questão 3.b

Realizei a transformada de Fourier e obtive amplitude e fase da mesma forma que a questão anterior. Na primeira imagem apliquei uma função linear arbitrária na amplitude, e na segunda apliquei uma função linear arbitrária na fase, ambas são exibidas. É possível observar que a alteração na fase faz uma diferença bem maior na reconstrução da imagem. Nesta questão é opcional passar dois parâmetros na linha de comando para especificar as imagens utilizadas, se não forem passados, ou se for passado apenas um parâmetro, imagens padrão serão utilizadas.

Questão 3.c

Nesta questão são exibidos 4 resultados:

1. Uma imagem reconstruída com a fase da imagem 1 e a amplitude da 2
2. Uma imagem reconstruída com a fase da imagem 2 e a amplitude da 1
3. A imagem 1 com a amplitude alterada
4. A imagem 2 com a fase alterada

As alterações na fase são mais significativas. Nesta questão é opcional passar dois parâmetros na linha de comando para especificar as imagens utilizadas, se não forem passados, ou se for passado apenas um parâmetro, imagens padrão serão utilizadas.

Questão 4

Não consegui fazer esta questão