

## Avaliação 2 | Sistemas Multimídia | 2021.1

1 mensagem

**Formulários Google** <forms-receipts-noreply@google.com>

15 de maio de 2021 20:29

Para: joao.gabriel.carneiro02@aluno.ifce.edu.br

Agradecemos o preenchimento de [Avaliação 2 | Sistemas Multimídia | 2021.1](#)

Isto foi o que recebemos de você:

## Avaliação 2 | Sistemas Multimídia | 2021.1

Seu e-mail ([joao.gabriel.carneiro02@aluno.ifce.edu.br](mailto:joao.gabriel.carneiro02@aluno.ifce.edu.br)) foi registrado quando você enviou este formulário.

Como nossos olhos fazem uma integração espacial, é possível obter resultados razoáveis usando uma estratégia de mapeamento em que intensidades de cinza são transformadas em densidades de pixels pretos. Estamos nos referindo a : \*

- ☒ Limiarização
- ☐ Convolução
- ☐ Laplaciano
- ☐ Dithering
- ☐ Borramento de imagem

Temos uma imagem em cores usando apenas 6 bits, sendo dois para cada componente RGB. Queremos exibir essa imagem em um dispositivo onde cada

canal de cor usa apenas 1bit (imagem binária). Descreva como usar dithering para obter o melhor resultado possível. \*

\* Primeiro que para se obter o 'melhor resultado possível' usando 'dithering' teríamos de usar uma 'matriz dithering ordenada' de tamanho/ordem 'nxn' que mantivesse a mesma quantidade de níveis de cinza que a imagem do enunciado da questão dispõe durante o processo de transformação dessa imagem em uma imagem binária, para que essa não tenha uma perda muito perceptível de tons (Se comparado aos tons da imagem de entrada inicialmente) na imagem de saída gerada após o processo de dithering.

\*\* Agora, para sabermos que tamanho é esse que a matriz de dithering precisa ter para obter-se o melhor resultado possível basta lermos o que é dito no enunciado sobre a imagem de entrada e o número de níveis de cinza que ela possui: "Uma imagem em cores usando apenas 6 bits, sendo dois para cada componente RGB". Logo, uma imagem com  $2^6-1(63)$  níveis de cinza (supõe-se que os pixels dessa imagem variem de 0 a 63), como são 63 níveis, a matriz de dithering para o melhor resultado no processo de dithering deve ser uma com o tamanho 8 (Pois,  $n=8$ ,  $8 \times 8 + 1 = 65$  níveis de cinza que podem ser representados)!

Qual é o efeito de aumentar o tamanho da máscara no filtro Highboost ? Explique.

\*

Para entender a consequência desse efeito de aumentar a máscara vamos nos lembrar o que é o processo de filtragem de realce: É exatamente um processo para aumentar a nitidez das imagens e que consiste em subtrair uma versão não nítida (suavizada) de uma imagem da imagem original. Esse processo segue os seguintes passos:

1- Borrar a imagem original.

2- Subtrair a imagem borrada da original (A diferença resultante é chamada de 'máscara'):

$$\rightarrow g(\text{mask})(x,y) = f(x,y) - s(x,y)$$

3- Adicionar a máscara à imagem original:

$$\rightarrow g(x,y) = f(x,y) + k * g(\text{mask})(x,y)$$

Dos passos mostrados acima podemos ter as seguintes relações com essa constante 'K' que aparece no passo '3-':

\*  $k = 1$  ---> 'unsharp masking'

\*  $K > 1$  ---> 'highboost filtering' (filtragem alto-reforço) <<< Caso pedido no enunciado!

\*  $K < 1$  ---> 'atenua a contribuição da máscara de nitidez'

Com isso, vemos que esse parâmetro 'K' é o responsável por controlar a intensidade desse filtro de realce, e manipulá-lo altera o tamanho da máscara usada nesse processo de filtragem de realce, assim, para cada valor maior de K maior será o aumento no tamanho da máscara na filtragem de tipo 'highboost' e mais 'nítida' será a imagem de saída (maiores serão os detalhes mínimos mostrados e haverá menos 'borramento' gerado e maior realce na figura de saída).

Queremos implementar um filtro para detectar contornos de objetos em uma imagem em um dispositivo com capacidade de processamento muito baixa, então esse filtro deve usar um número reduzido de operações. Até mesmo um filtro com máscara de tamanho 3x3 pode ser inviável nesse dispositivo. O Professor Pardal propôs o uso do filtro abaixo. Argumente sobre as vantagens desse filtro (número

de operações) e também das suas deficiências. Como esse filtro se compara com outros detectores de contornos que vimos? DICA: aplicar o filtro a algumas imagens usando o Gimp ajuda a compreender seu funcionamento. \*

-1	1
----	---

\* Primeiro veja que esta máscara apresenta dois elementos apenas, de valores iguais e ainda de SINAIS OPOSTOS, ou seja, vai ter-se um pouco daquelas questões de 'diferenças' que são presentes em outros detectores de contorno vistos (Como o Sobel e o Laplaciano por exemplo).

\* (O que a máscara faz) Nessa máscara estamos fazendo a diferença do pixel central com o da direita, pois ambos tem sinais opostos, a diferença é feita para cada pixel na imagem (a diferença de cada pixel com o seu vizinho da direita).

- Quando essa diferença é zero temos que o pixel vizinho tem o mesmo valor que o central (a variação dos níveis de cinza é zero).

- Quando a diferença for diferente de zero é porque tem-se uma variação dos níveis de cinza entre o pixel central e seu vizinho da direita o que resultará em um valor de saída diferente de zero de valor positivo ou negativo.

\*\* (Vantagens) Como primeira vantagem podemos citar que este, diferente de alguns dos filtros de convolução estudados, possui bem menos operações a se fazer, pois só trabalha com dois pixels, o central e o da direita e só é feito ao todo UMA OPERAÇÃO entre esses, que no caso é uma diferença (já citada no tópico acima). Com tem menos operações esse filtro possuirá 'custo computacional' baixo em sua aplicação.

\*\*\* (Desvantagens) A principal desvantagem pode ser vista na imagem de saída desse filtro que não possuirá bordas/contornos muito 'vantajosos' ou 'realçados', nem todos os contornos da imagem de entrada serão bem destacados, muito dos contornos no eixo VERTICAL mal serão exibidos, somente os do eixo HORIZONTAL, e a aparência 'visual' da imagem parecerá um pouco 'artificial' e poderá deixar ficar um pouco de ruído nela caso haja anteriormente na imagem de entrada.

\*\*\*\* (Comparação com outros filtros) A imagem de saída gerada pela aplicação desse filtro, depois de gerada, é pouco comparável com aquelas obtidas após a aplicação de detectores de contorno laplacianos sendo mais semelhante aquelas que usam dos filtros de Sobel. Essa semelhança com Sobel se dá pois na imagem de saída do filtro  $[-1, 1]$  só podemos ver dentre os contornos destacados os 'verticais' com mais clareza, isso se assemelha com um dos filtros mostrados pelo professor na cadeira, chamado de 'Sobel x'. Outra semelhança do filtro  $[-1, 1]$  com esse 'Sobel x' é o fato de que em ambos os contornos do lado esquerdo da imagem de saída aparecem 'claros' e os do lado direito 'escuras'. Mesmo que se assemelhe com os filtros Sobel, a imagem de saída do filtro  $[-1, 1]$  possui seus contornos 'menos delimitados' do que as que usam Sobel, que são um pouco mais detalhadas em seus contornos, seja no eixo X ou no eixo Y, além do fato de que os níveis de cinza da imagem de saída também não são exatamente os mesmos nas imagens que usam Sobel e nas que usam do filtro  $[-1, 1]$ , são níveis de tonalidades bem 'próximas' mas ainda diferentes visualmente (um mais escuro que o outro e vice-versa). (Obs. As imagens de saída laplacianas não só tem níveis de cinza de tonalidades bem diferentes das com o filtro  $[-1, 1]$ , como, também, as bordas das imagens destacam-se de forma diferente das com o filtro  $[-1, 1]$ , que só destaca mais as bordas no eixo horizontal, enquanto nos filtros laplacianos temos destaque nos contornos em ambos os eixos mas de uma forma menos intensa/nítida que no filtro  $[-1, 1]$ ).

Proponha um filtro cuja máscara não ultrapassa o tamanho 2x2 e que tem funcionamento melhor que o proposto pelo Professor Pardal na questão anterior. Justifique sua escolha. \*

\* Um filtro que vem a mente, depois de muita pesquisa kkkkk, é o filtro de Roberts, que segue as duas possíveis variâncias de uso em suas máscaras de convolução:

i)  $G_x = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$

ii)  $G_y = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$

\*\* Comparando os resultados desse filtro com o [-1,1] vemos que de fato, a imagem de saída do filtro de Roberts é 'muuuuito' superior em detalhes do que a apresentada pelo filtro da questão anterior (Se pudesse eu mostraria a diferença visualmente que é BEM destacável). A começar pelo realce e a qualidade das bordas que é a principal desvantagem do filtro [-1,1] e, também, é o que mais se percebe na imagem de saída do mesmo, no de Roberts as bordas da imagem são bem visíveis em todos os eixos (Horizontal e Vertical) da imagem, os detalhes, também, são muito mais nítidos do que no filtro [-1,1] e não há muito essa diferença de 'claro' e 'escuro' que havia nas bordas da direita e esquerda na imagem do filtro [-1,1], ou, pelo menos, não foi o que encontrei nos exemplos de imagens com filtro de Roberts.

\*\*\* Por esse detalhamento mais rico na detecção de bordas em imagens dado pelo filtro de Roberts, é que creio que esse seja uma proposta muito boa para o que o enunciado pede, além do fato de que esse filtro trabalha com as dimensões 2x2, que é perfeito para a escolha que é pedida na questão (Claro, há filtros bem melhores que o de Roberts, como o de Sobels que por muitos é considerado o 'ideal' na maioria dos casos, mas seu tamanho ultrapassa o tamanho 'máximo' aceitado pela questão, logo, o descartei, se não, com toda a certeza eu o escolheria porque cobre algumas falhas que existem no filtro de Roberts mas que não vem ao caso serem discutidas nessa solução).

Pergunta sem título \*



Opção 1

Crie seu próprio formulário do Google.

Denunciar abuso

**\* Abaixo as duas questões que não estavam totalmente corretas com o "feedback" do professor, vejam se conseguem acertá-las agora kkkkk:**

Temos uma imagem em cores usando apenas 6 bits, sendo dois para cada componente RGB. Queremos exibir essa imagem em um dispositivo onde cada canal de cor usa apenas 1bit (imagem binária). Descreva como usar dithering para obter o melhor resultado possível. \*

\* Primeiro que para se obter o 'melhor resultado possível' usando 'dithering' teríamos de usar uma 'matriz dithering ordenada' de tamanho/ordem 'n x n' que mantivesse a mesma quantidade de níveis de cinza que a imagem do enunciado da questão dispõe durante o processo de transformação dessa imagem em uma imagem binária, para que essa não tenha uma perda muito perceptível de tons (Se comparado aos tons da imagem de entrada inicialmente) na imagem de saída gerada após o processo de dithering.

\*\* Agora, para sabermos que tamanho é esse que a matriz de dithering precisa ter para obter-se o melhor resultado possível basta lermos o que é dito no enunciado sobre a imagem de entrada e o número de níveis de cinza que ela possui: "Uma imagem em cores usando apenas 6 bits, sendo dois para cada componente RGB". Logo, uma imagem com  $2^6 - 1 = 63$  níveis de cinza (supõe-se que os pixels dessa imagem variem de 0 a 63), como são 63 níveis, a matriz de dithering para o melhor resultado no processo de dithering deve ser uma com o tamanho 8 (Pois,  $n = 8, 8 \times 8 + 1 = 65$  níveis de cinza que podem ser representados)!

**Feedback individual**

*Você supôs corretamente que são 2 bits por pixel em cada canal mas usou o valor 6 bits para fazer as contas.*

Qual é o efeito de aumentar o tamanho da máscara no filtro Highboost ?  
Explique. \*

Para entender a consequência desse efeito de aumentar a máscara vamos nos lembrar o que é o processo de filtragem de realce: É exatamente um processo para aumentar a nitidez das imagens e que consiste em subtrair uma versão não nítida (suavizada) de uma imagem da imagem original. Esse processo segue os seguintes passos:

- 1- Remover a imagem original.
- 2- Subtrair a imagem borrada da original (A diferença resultante é chamada de 'máscara'):  
 $\rightarrow g(\text{mask})(x,y) = f(x,y) - s(x,y)$
- 3- Adicionar a máscara à imagem original:  
 $\rightarrow g(x,y) = f(x,y) + k \cdot g(\text{mask})(x,y)$

Dos passos mostrados acima podemos ter as seguintes relações com essa constante 'K' que aparece no passo '3':

- \*  $k = 1 \rightarrow$  'unsharp masking'
- \*  $K > 1 \rightarrow$  'highboost filtering' (filtragem alto-reforço) <<< Caso pedido no enunciado!
- \*  $K < 1 \rightarrow$  'atenua a contribuição da máscara de nitidez'

Com isso, vemos que esse parâmetro 'K' é o responsável por controlar a intensidade desse filtro de realce, e manipulá-lo altera o tamanho da máscara usada nesse processo de filtragem de realce, assim, para cada valor maior de K maior será o aumento no tamanho da máscara na filtragem de tipo 'highboost' e mais 'nítida' será a imagem de saída (maiores serão os detalhes mínimos mostrados e haverá menos 'borramento' gerado e maior realce na figura de saída).

**Feedback individual**

*O valor de k e o tamanho da máscara de borramento são parâmetros independentes entre si mas que, ambos, influenciam o efeito do filtro.*