

# Universidade do Estado de Santa Catarina

## Aguçamento de imagens

### **Tarefa 3**

Draylon Vieira  
Victor Bernardes

19 de março de 2021

# Sumário

Parte A	1
Parte B	2
Parte C.1	3
Parte C.2	4

### **Resumo**

Neste trabalho será abordado técnicas para obtenção de gradientes e magnitudes bem como formas de varrer esses dados da imagem e aplicar os diversos filtros para testes.

# Parte A:

Para calcularmos as coordenadas da Figura 1 realizamos primeiramente uma conversão para tons de cinza usando uma função própria da biblioteca opencv.

```
cv.cvtColor(image, cv.COLOR_BGR2GRAY)
```

Depois implementamos a função janela onde recebe os parametros: a imagem, x, y e n. Onde x,y são a dimensão da imagem e n o valor da janela que nos queremos usar, no caso será  $n = 3$ .



Figura 1: Imagem utilizada.

```
Grayscale da imagem  
Janela 3 x 3 em 460 : 200  
[[120, 125, 131], [126, 132, 142], [134, 144, 152]]
```

Figura 2: Saída do algoritmo após realizar o procedimento.

## Parte B:

Usando a função de janela que foi implementado na parte A e sendo  $N = 3$  o tamanho da matriz de convolução (janela deslizante), para fazer o filtro de média basta calcular a média da janela com  $n = 3$  iterando pela imagem e aplicar na região. Como podemos observar na Figura 3, a imagem da direita está com uma nitidez maior em relação a imagem da esquerda ao qual foi aplicado o filtro de média. Portanto o filtro de média ofusca os ruídos da imagem.

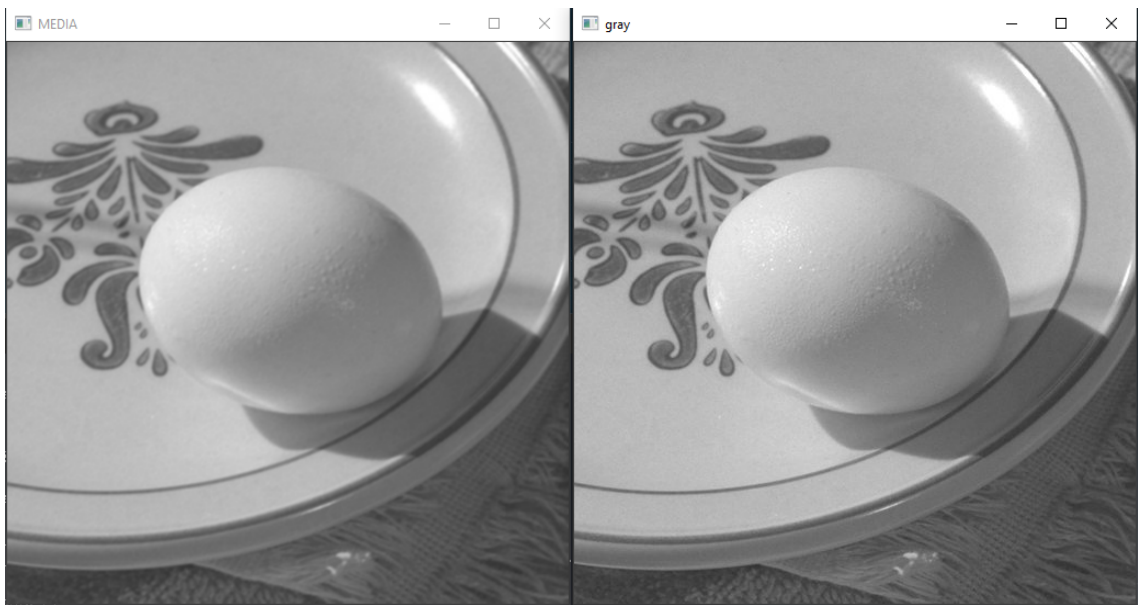


Figura 3: Saida do algoritmo após realizar o procedimento.

## Parte C.1:

Após aplicar o gradiente proposto pela questão podemos observar o resultado na Figura 4. A primeira imagem é a original, a próxima está com sobel em x aplicado e a imagem ao lado está com sobel em y aplicado.

Nas imagens abaixo foram aplicados o gradiente proposto, sendo  $k = 0$  a primeira imagem,  $k = 1$  a imagem ao lado e  $k = 2$  a ultima imagem.

Portanto, podemos observar que quanto maior o valor de  $k$ , menor será o threshold portanto maior será a detecção da intensidade pelo filtro gradiente em questão.

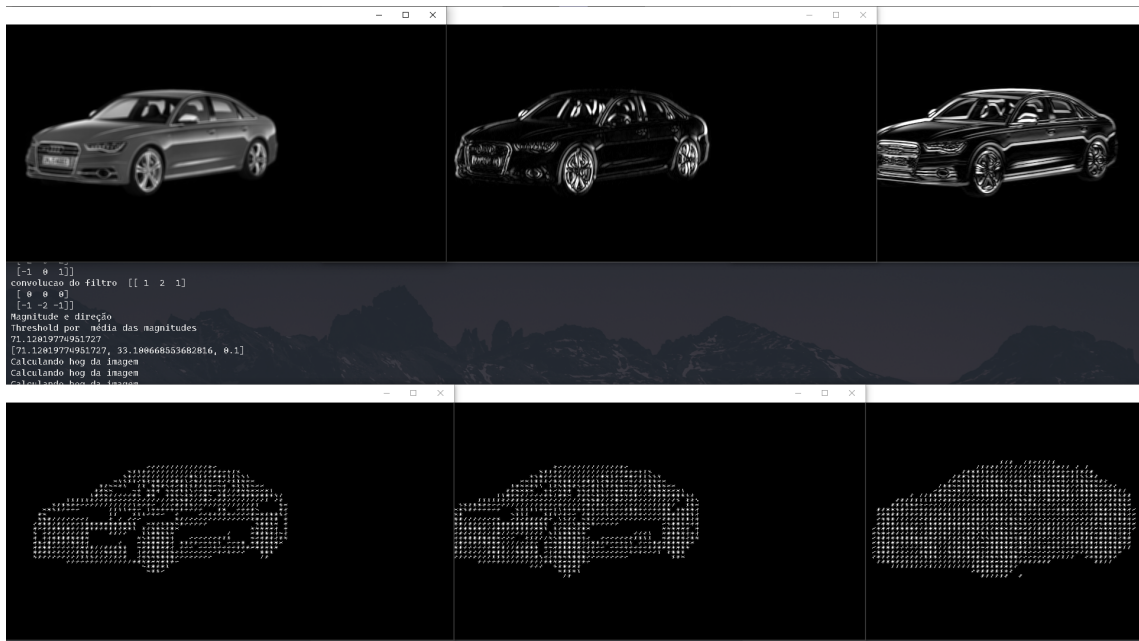


Figura 4: Saida do algoritmo após realizar o procedimento.

## Parte C.2:

Nessa questão utilizaremos a Figura 1 para aplicar o histograma de gradientes. A região em questão onde o HOG foi aplicado é [456 : 192], sendo que os pixels foram divididos em células de dimensão igual a 8x8. podemos observar que os angulos mais frequentes estão entre 0° e 80°



Figura 5: Região delimitada para estudo do HOG.

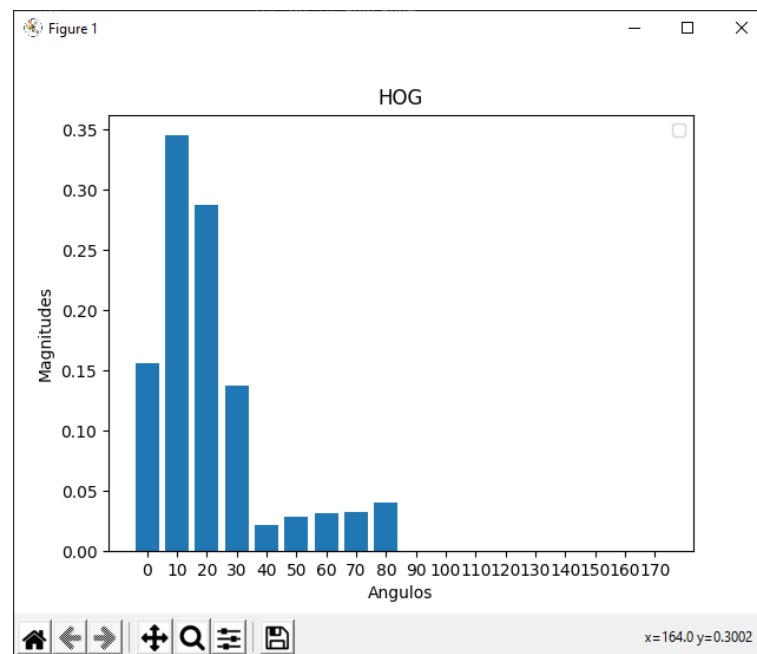


Figura 6: Saída do algoritmo após realizar o procedimento.

# Referências Bibliográficas

MALLICK, S. *Histogram of Oriented Gradients explained using OpenCV*. Disponível em: <<https://learnopencv.com/histogram-of-oriented-gradients/>>.

SOLOMON, C.; BRECKON, T. *Fundamentos de Processamento Digital de Imagens: Uma abordagem pratica com exemplos em matlab*. Ltc. São Paulo: [s.n.], 2013.

WENG, L. *Object Detection for Dummies Part 1: Gradient Vector, HOG, and SS*. Disponível em: <<https://lilianweng.github.io/lil-log/2017/10/29/object-recognition-for-dummies-part-1.html>>.