

Computer Vision

PS-02

Nome: Caio Silva Gonçalves

Professor: Flávio Luis Cardeal Pádua

Centro Federal de Educação Tecnológica de Minas Gerais

1)

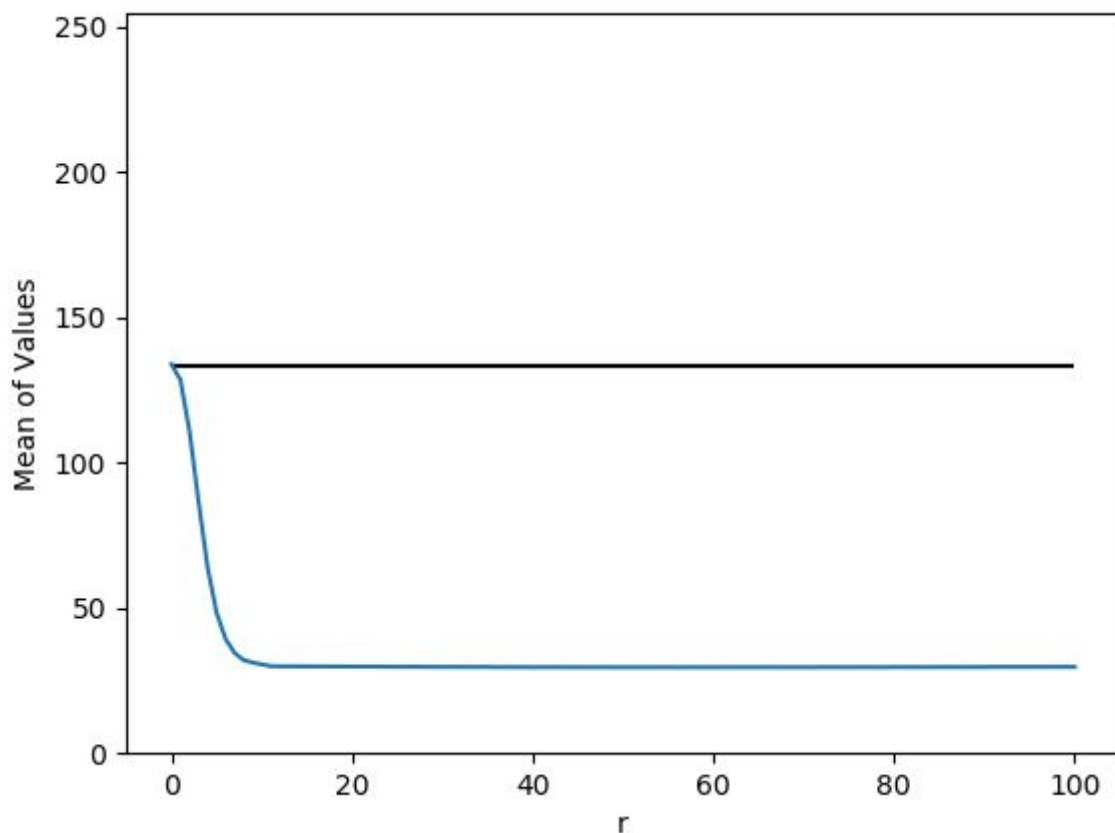
Seguindo a Fórmula:

$$g_{equal}^{(r)}(u) = \frac{G_{\max}}{Q} \sum_{w=0}^u h_I(w)^r \quad \text{with } Q = \sum_{w=0}^{G_{\max}} h_I(w)^r$$

Foram implementados 3 métodos principais:

- **calculate_relative_hist**: método para cálculo do histograma relativo: histograma / número de pixels, elevado à variável “r”. A constante Q da fórmula é o somatório desses valores;
- **calculate_g_equal**: para o cálculo do “g”, que servirá para o mapeamento dos valores da imagem;
- **equalize**: utilizando os valores de “g” do método anterior, percorre-se os valores da imagem alterando-os para os valores mapeados em “g”.

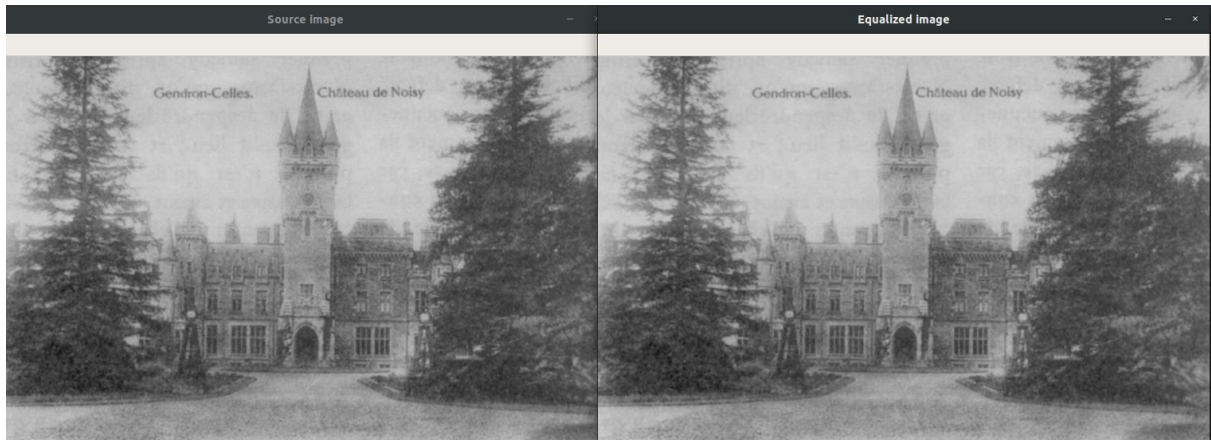
Foi aplicado na imagem de input um filtro Gaussian, com o método “GaussianBlur” do opencv. Variando os valores de “r” e salvando a média dos valores da imagem foi possível a construção do gráfico “r x Média”. A linha preta constante é o valor de média inicial da imagem (sem equalização):



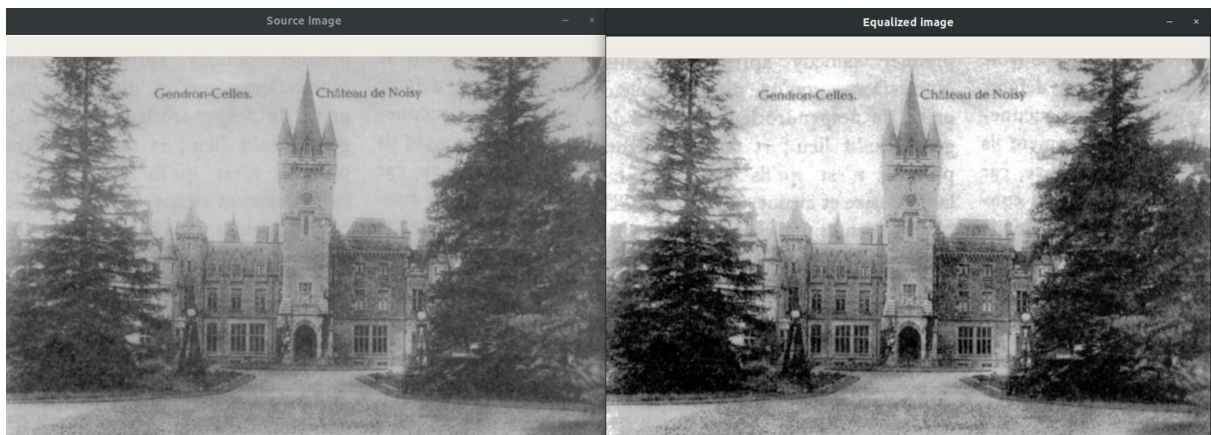
Pela observação do gráfico “r x Média”, foi verificado que a imagem fica mais escura (com maior quantidade de valores baixos) quando o “r” vai aumentando, mas chega em um ponto onde se atinge um limite.

As reconstruções da imagem confirmaram os resultados:

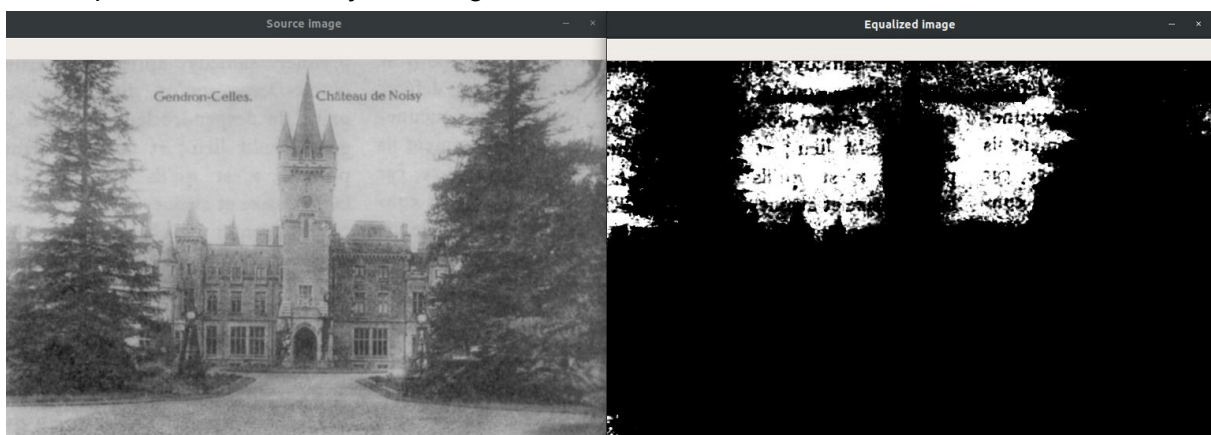
r = 0, imagem não se alterou:



r = 1, histograma uniforme:



r = 20, ponto onde o limite já foi atingido:

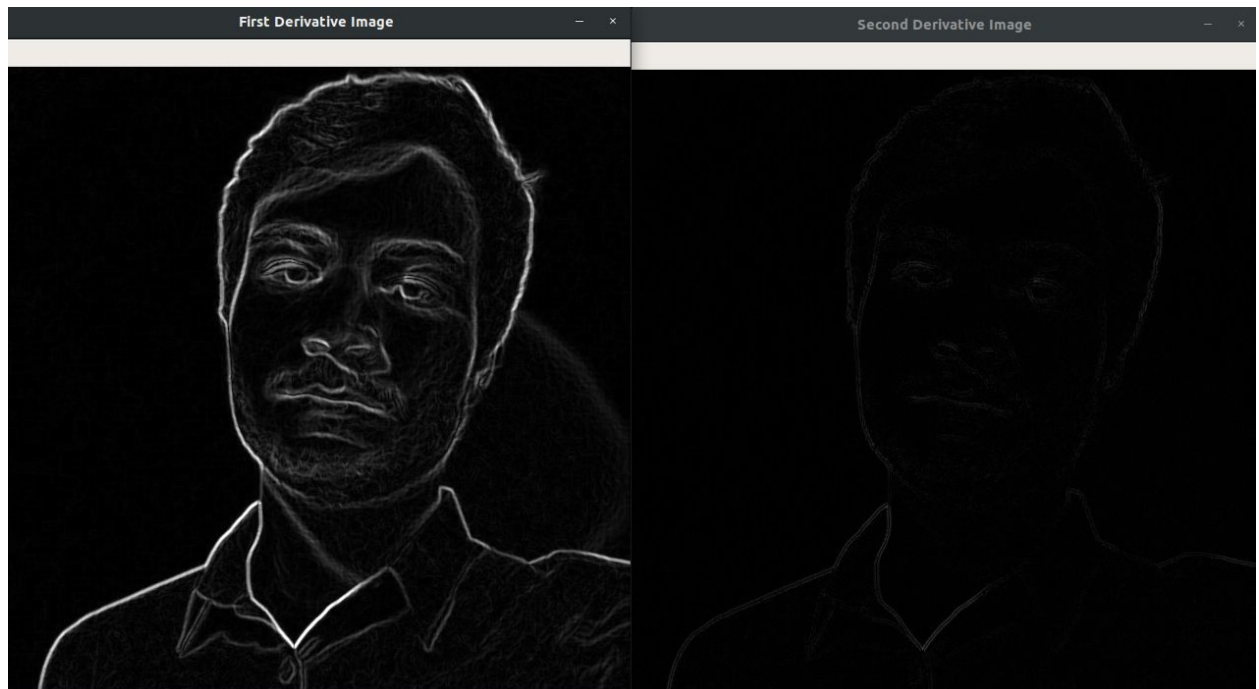


2) Foi implementado um detector de borda utilizando as estratégias de derivadas de primeira ordem e segunda ordem. Foram utilizados os seguintes métodos do opencv: “Sobel”, para a derivada de primeira ordem, e “Laplacian”, para a de segunda ordem.

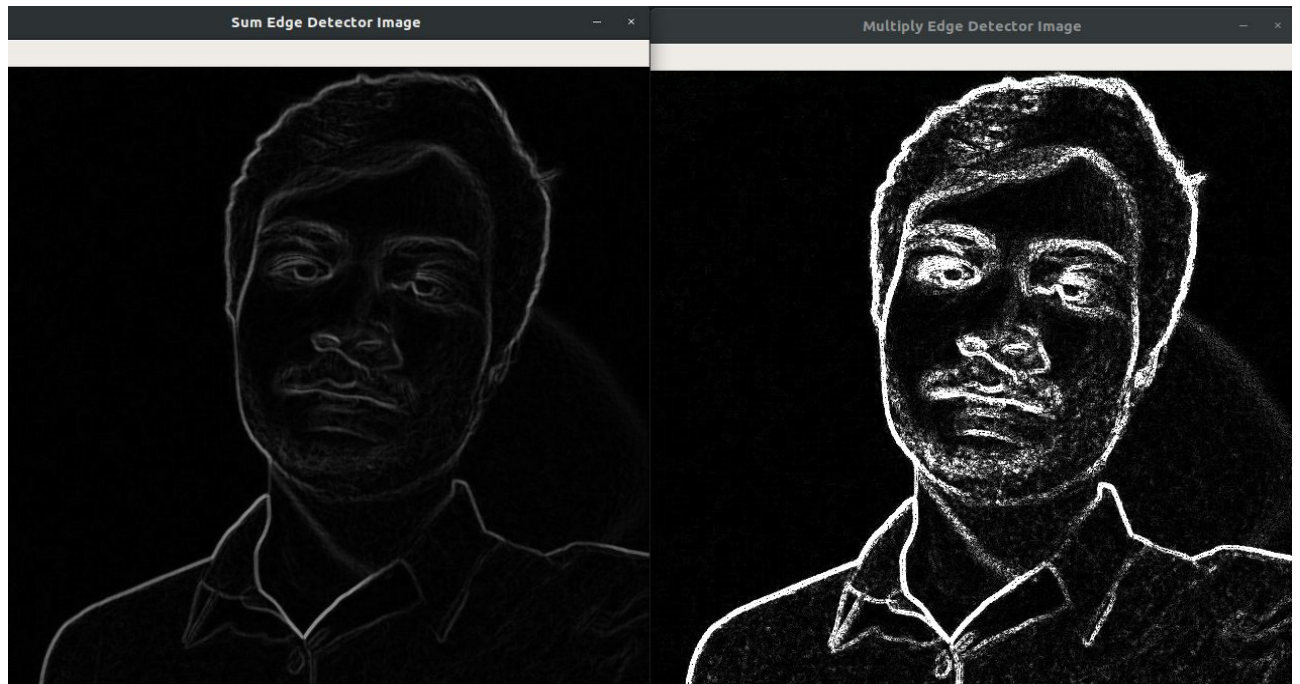
Imagem de input:



Outputs das derivadas de primeira e segunda ordem, respectivamente:

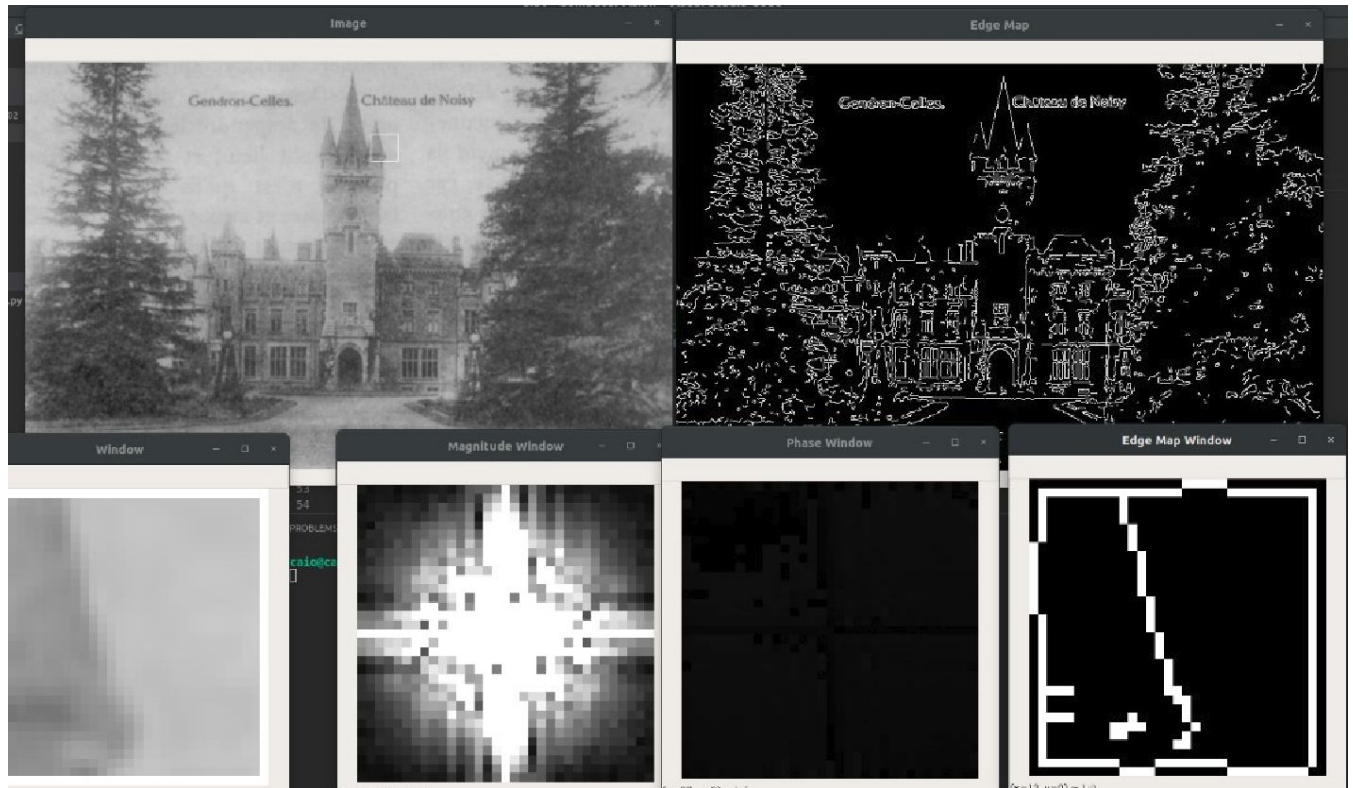


A combinação da derivada de 1ª ordem com a de 2ª ordem foi realizada de duas formas: soma ponderada, com pesos iguais a 0.5, e multiplicação. Os resultados se encontram a seguir:



3) Parte do código implementado nas questões sobre janela e transformações fourier do “PS_01” foi aproveitado nessa questão. O acréscimo foi a criação de 3 novas janelas: imagem mapeada pela magnitude, imagem mapeada pela phase e o mapa de bordas.

Para o mapa de bordas foi utilizado o método “Canny” do opencv.



Em relação à magnitude, foi observado uma uniformidade na janela quando se passa por partes uniformes da imagem, porém quando se aproxima de bordas, a janela de magnitude começa a variar muito.

Em relação à phase, não se obteve nenhum resultado concreto.

Referências:

https://docs.opencv.org/3.4.3/d2/d2c/tutorial_sobel_derivatives.html

https://docs.opencv.org/2.4/doc/tutorials/imgproc/imgtrans/laplace_operator/laplace_operator.html

https://docs.opencv.org/3.1.0/da/d22/tutorial_py_canny.html