Processamento Digital de Imagens – Trabalho 2
Filtros passa-baixa e passa-alta Butterworth

Guilherme L. Salomão, João Victor M. Freire,
Martin Heckmann, Renan D. Pasquantonio

8 de Outubro de 2020

1 Introdução

A Transformada de Fourier é uma poderosa ferramenta no Processamento Digital de Imagens e Sinais, uma vez que ela permite representar uma imagem (domínio espacial) no domínio das frequências. Essa representação é, em muitos casos, mais simples e intuitiva que a do domínio espacial.

Além de realizar filtragens no espectro da imagem para depois converte-la para o domínio espacial novamente, o Teorema da Convolução nos permite *convoluir* uma imagem com um filtro utilizando o produto da Transformada de Fourier de ambos e depois calculando sua Transformada Inversa, da seguinte forma:

$$f(x) * g(x) = \mathscr{F}^{-1}\{F(\mu)G(\mu)\},$$

na qual F e G são as transformadas de f e g, respectivamente, e $\mathscr{F}^{-1}$ é a transformada inversa de Fourier.

Esse teorema nos permite aplicar qualquer filtro linear à uma imagem no domínio da frequência. Dependendo do tamanho dos filtros que utilizarmos, esse método de filtragem é mais eficiente que a convolução e correlação-cruzada comum.

2 Motivação

Dentre os diversos filtros que podemos aplicar no domínio da frequência usando a transformada, dois que temos um interesse particular são os filtros passa-baixa e passa-alta.

Como o nome sugere, o filtro passa-baixa permite apenas a passagem de frequências baixas, causando a eliminação das mais altas – comumente associadas a detalhes de uma imagem. Assim, temos um efeito análogo ao de um filtro de suavização do domínio espacial.

De forma semelhante, o passa-alta elimina frequências menores, e obtém um efeito similar ao cálculo da derivada da imagem.

São diversos os filtros de passa-baixa e passa-alta. Em aula, fomos apresentados ao filtro passa-baixa ideal e gaussiano, e ao passa-alta gaussiano e laplaciano. Além desses, existem também os filtros Butterworth passa-baixa e passa-alta. Esses são muito utilizados por buscarem uma resposta de frequência o mais plana possível, mas sem ser abrupto como em um filtro passa-baixa ideal.

3 Explicação do Método Implementado

3.1 Passa-Baixa

A função que vamos utilizar para definir o filtro é:

$$H(\mu,\nu) = \frac{1}{1 + \left(\frac{D(\mu,\nu)}{D_0}\right)^{2n}},$$

na qual $D(\mu,\nu) = \sqrt{\mu^2+\nu^2}$. Os valores D_0 e n são parâmetros variáveis, e faremos uma análise dos seus efeitos na Seção 4 do documento. A seguir está nossa implementação do filtro:

```python
def filtro_passa_baixa_butterworth(img, d0, n):
    num_rows, num_cols = img.shape
    freq_r, freq_c = generate_frequencies(num_rows, num_cols)

    low_pass_butterworth = np.zeros([num_rows, num_cols])
    for row in range(num_rows):
        for col in range(num_cols):
            dist = np.sqrt(freq_r[row]**2 + freq_c[col]**2)
            H = 1/(1 + (dist/d0)**(2*n))

            low_pass_butterworth[row, col] = H

    return low_pass_butterworth
```

A função `generate_frequencies` é utilizada para gerar as frequências na Transformada de Fourier com as dimensões da imagem, após isso será criado a matriz `low_pass_butterworth` que irá armazenar os valores do filtro depois de serem calculados. Dentro do loop é armazenado em cada elemento de `low_pass_butterworth` o resultado da função que define o filtro na mesma posição, com os valores de `d0` e `n` definidos na chamada da função.

3.2 Passa-Alta

No caso do filtro passa-alta, a função que define o filtro é:

$$H(\mu, \nu) = \frac{1}{1 + (\frac{D_0}{D(\mu,\nu)})^{2n}},$$

na qual $D(\mu, \nu) = \sqrt{\mu^2 + \nu^2}$.

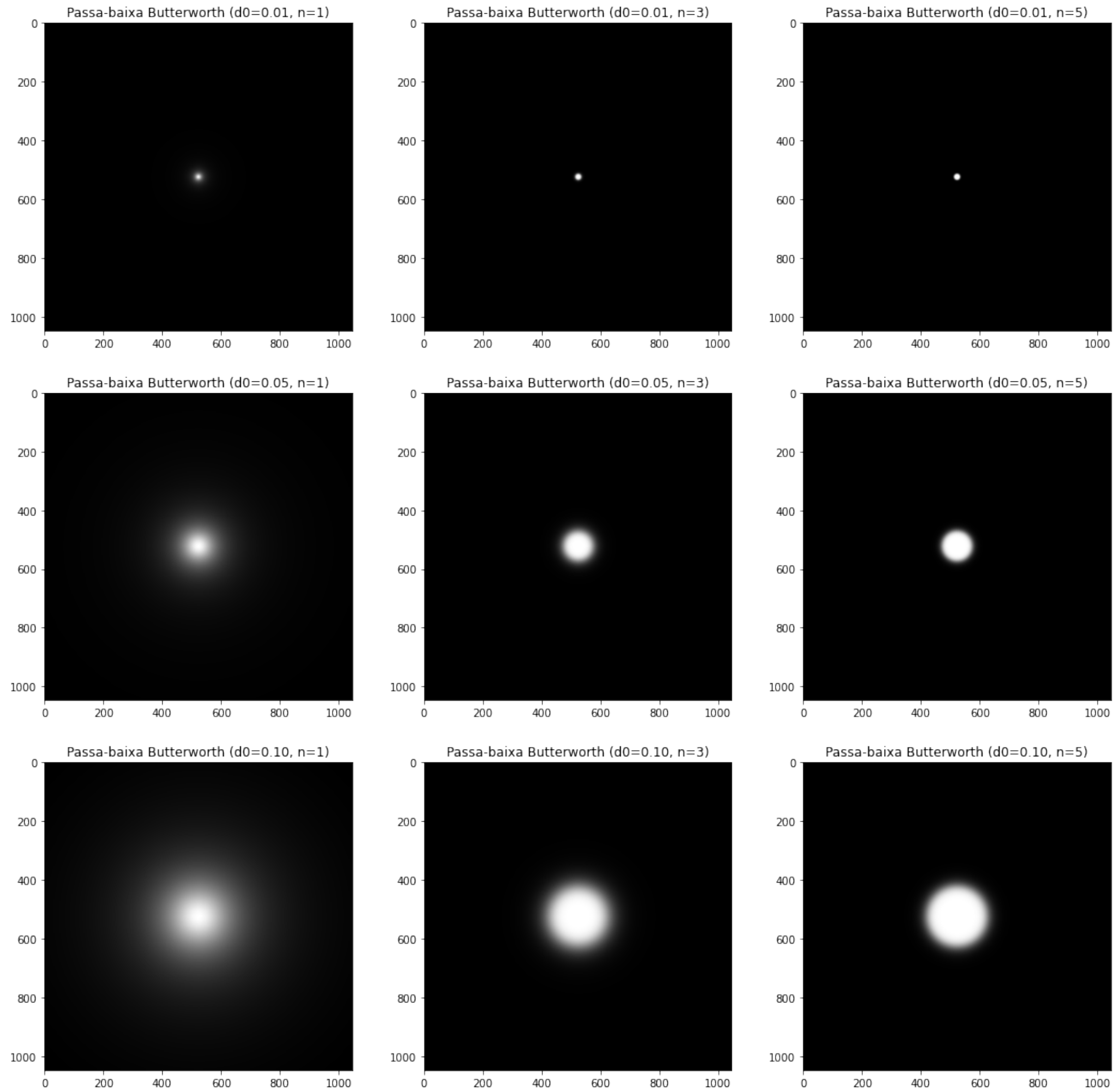
```python
def filtro_passa_alta_butterworth(img, d0, n):
    num_rows, num_cols = img.shape
    freq_r, freq_c = generate_frequencies(num_rows, num_cols)

    high_pass_butterworth = np.zeros([num_rows, num_cols])

    for row in range(num_rows):
        for col in range(num_cols):
            dist = np.sqrt(freq_r[row]**2 + freq_c[col]**2)
            if dist == 0:
                dist = 1
            H = 1/(1+(d0/dist)**(2*n))

            high_pass_butterworth[row,col] = H

    return high_pass_butterworth
```

De forma similar ao filtro passa baixa, também é utilizada a função `generate_frequencies` para gerar as frequências na Transformada de Fourier com as dimensões da imagem, após isso será criado a matriz `high_pass_butterworth` que irá armazenar os valores do filtro depois de serem calculados. Dentro do loop é armazenado em cada elemento de `high_pass_butterworth` o resultado da função que define o filtro na mesma posição, com os valores de `d0` e `n` definidos na chamada da função.

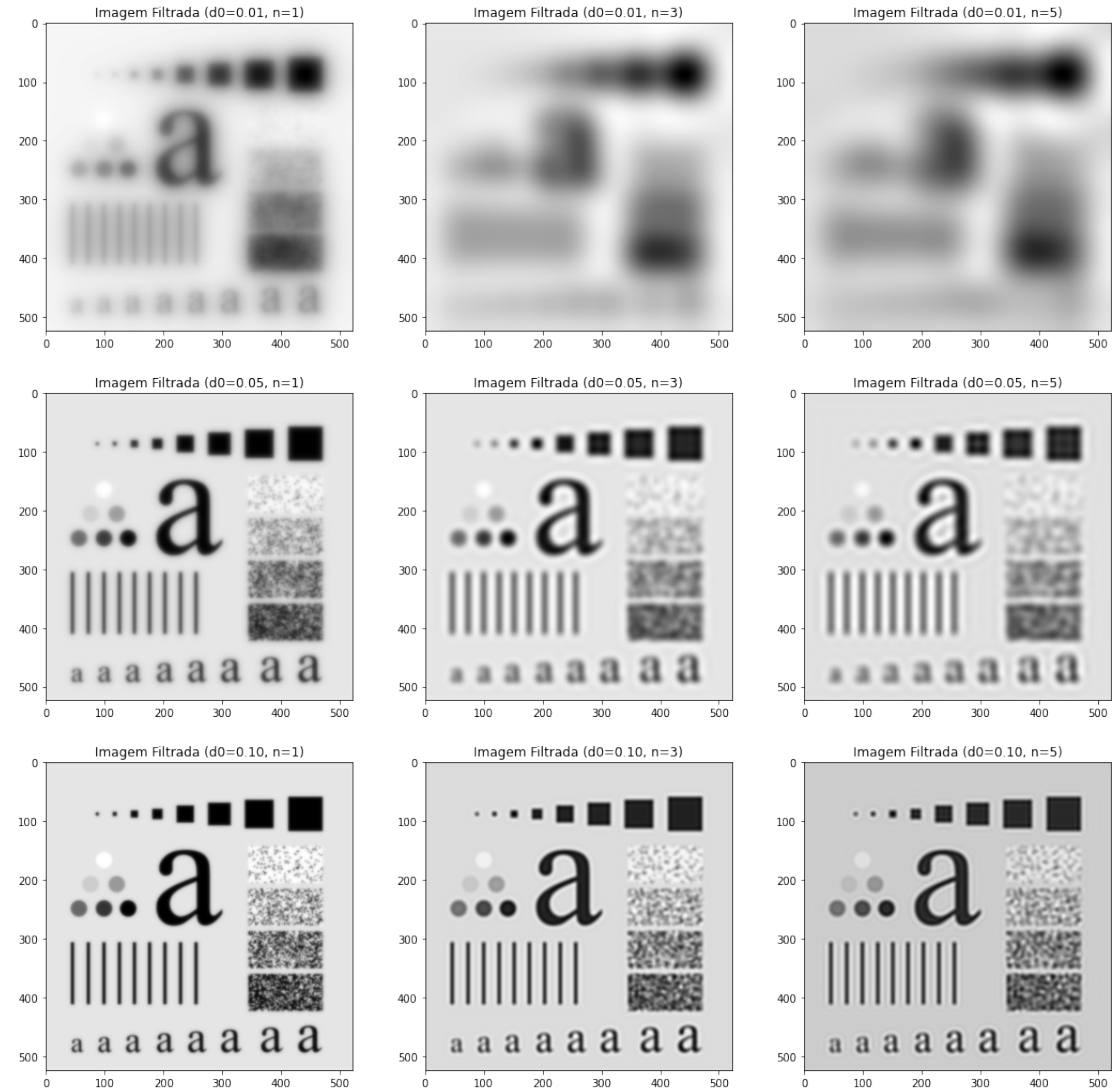
4 Análise dos Parâmetros do Filtro

Os filtros Butterworth recebem dois parâmetros: D_0 e n. Testamos diferentes valores de cada um para analisarmos como eles afetam o filtro resultante. Fizemos os experimentos com os valores $D_0 = \{0,01; 0,05; 0,10\}$ e de $n = \{1, 3, 5\}$.

Para esses valores, obtivemos os seguintes filtros passa-baixa:

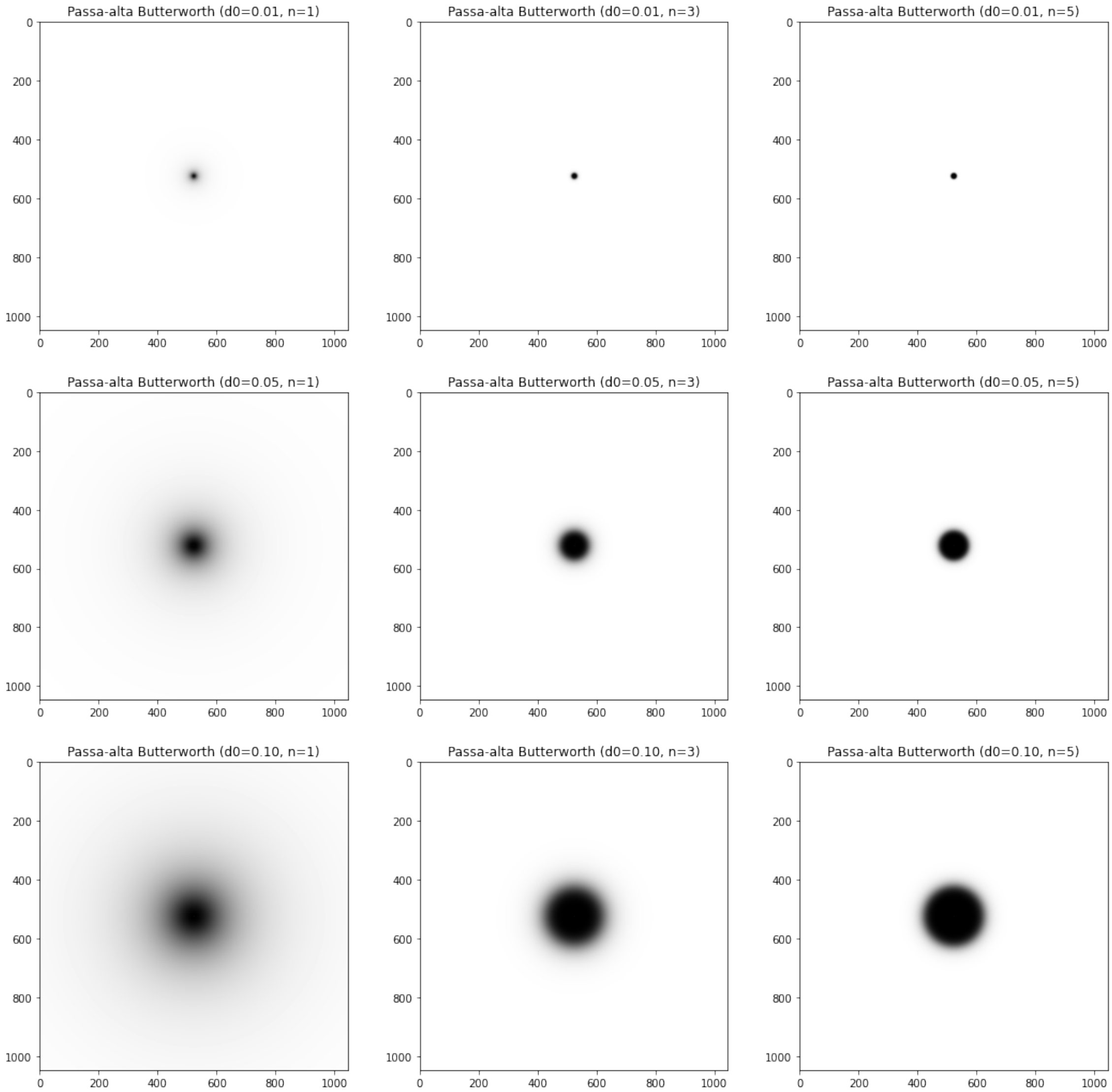


Observando os filtros, nota-se que conforme D_0 cresce, o raio das frequências permitidas também cresce. Assim, um valor pequeno de D_0 vai fazer uma suavização maior da imagem. Já ao observarmos o parâmetro n, percebe-se que seu valor esta associado com a suavidade da transição entre as frequências permitidas e eliminadas, de forma que um valor grande de n se aproxima de um passa-baixa ideal. A seguir, aplicamos os filtros a uma imagem.



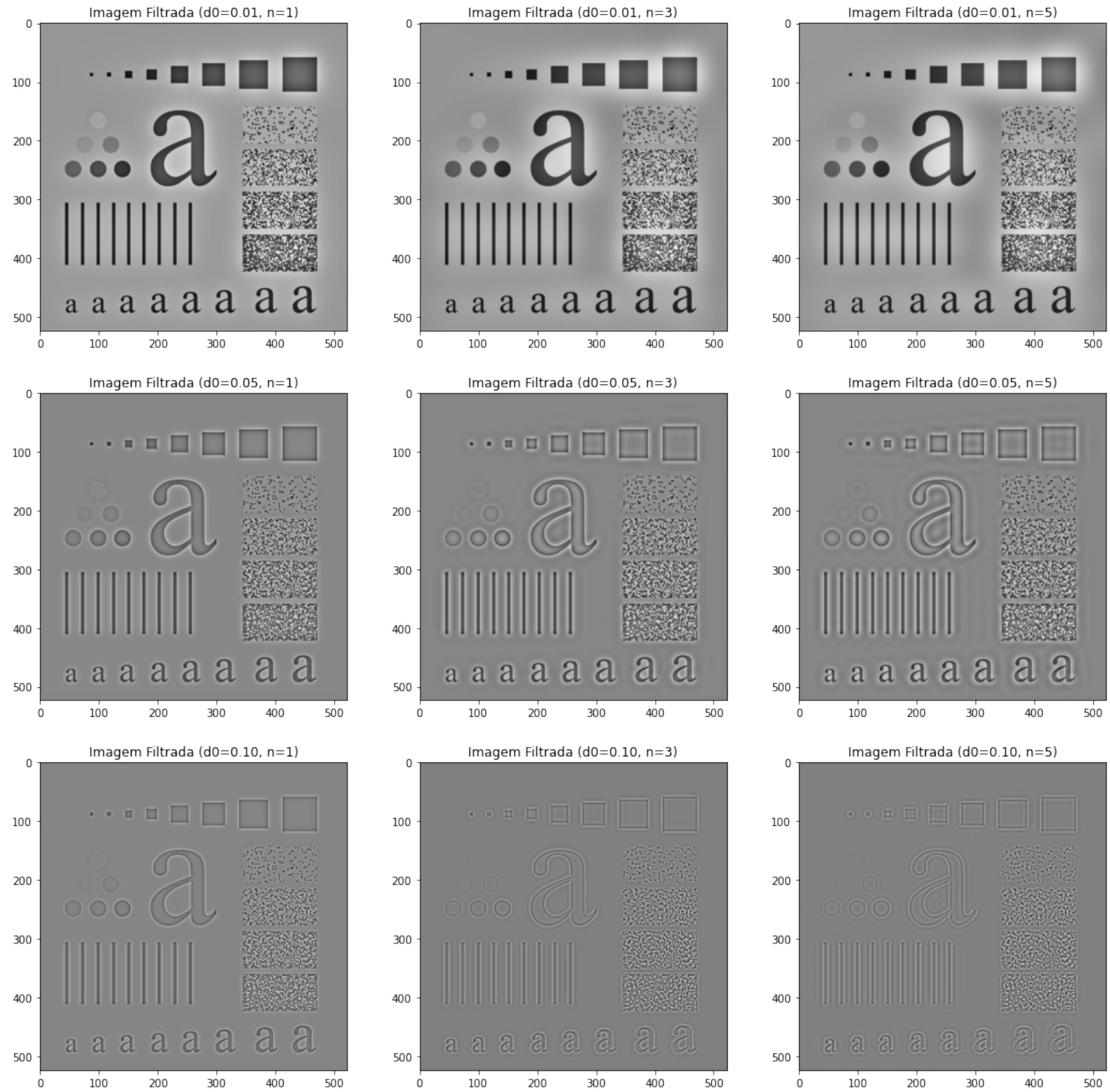
Conforme o valor de n cresce e nosso filtro se torna abrupto, os artefatos de ondulação percebidos no filtro passa-baixa ideal também acontecem aqui. É possível observar o valor de D_0 controlando o nível da suavização, sendo bastante intensa quando seu valor é pequeno.

Efeitos análogos acontecem no filtro passa-alta. O parâmetro n tem o mesmo efeito de controlar a suavidade da transição entre as frequências permitidas e eliminadas. Entretanto, a razão que continha D_0, $\frac{D(\mu,\nu)}{D_0}$, se tornou $\frac{D_0}{D(\mu,\nu)}$. Assim, D_0 passou a controlar o raio das frequências que serão eliminadas.



Já ao analisarmos as imagens com o filtro aplicado, percebemos que o artefato de ondulação continua aparecendo para grandes valores de n. Para D_0, se forem pequenos o filtro não remove frequências o suficiente, não obtendo o efeito de derivada esperado. Se for muito grande, ele vai eliminar muitos detalhes, deixando as bordas muito pouco definidas.

Dentre os casos testados, acreditamos que a combinação $D_0 = 0.05$ e $n = 1$ obteve o melhor resultado dentro das nossas expectativas, com poucos artefatos e bordas relativamente definidas.



Assim sendo, baseado em nossas análises empíricas obtivemos resultados com menos artefatos com valores pequenos de n, como 1 ou 2. Além disso, as imagens resultantes satisfatórias se deram com valores intermediários de D_0 no passa-alta e com valores pequenos no passa-baixa.

Referências

[1] Comin, C. H. Processamento Digital de Imagens – Aula 15: Filtragem Espacial no Domínio da Frequência 2, 2020.

[2] Wikipedia. Butterworth Filter. `https://en.wikipedia.org/wiki/Butterworth_filter`.