

Universidade Estadual de Campinas

MC920 - Introdução ao processamento de
imagem digital

Trabalho 2

Felipe Hideki Matoba
RA: 196767

Sumário

1	Resumo	2
2	Ambiente de execução	2
3	Transformada rápida de Fourier	2
4	Filtros passa-baixa	3
4.1	Filtro passa-baixa ideal	3
4.2	Filtro passa-baixa de Butterworth	4
5	Filtros passa-alta	7
5.1	Filtro passa-alta de Butterworth	7
6	Filtros passa-faixa	9
6.1	Filtro passa-faixa de Butterworth	9
7	Rotação da imagem	10
8	Compressão de imagem	12
9	Desempenho	13
10	Conclusão	14

1 Resumo

A transformada de Fourier pode ser aplicada em imagens, convertendo suas informações para o domínio da frequência. Nesse domínio, a operação de convolução, já estudada no trabalho anterior no domínio espacial, corresponde a um simples produto entre duas funções, que no caso discreto gera um número bem reduzido de operações aritméticas.

Assim, desde que o custo de transformação da imagem entre os dois domínios não seja muito elevado teremos um algoritmo mais eficiente para a filtragem de imagens. Utilizaremos neste trabalho a transformada rápida de Fourier, que possui um custo reduzido de $O(N \log N)$ em comparação com a transformada usual, de $O(N^2)$, onde N é o tamanho dos dados.

Serão aplicadas nas imagens filtros passa-baixa, passa-alta e passa-faixa, será observado o efeito que uma rotação tem em seu espectro de Fourier e, por fim, será feita uma compressão da imagem.

2 Ambiente de execução

O código foi feito utilizando o Jupyter Notebook, com a inclusão das seguintes bibliotecas:

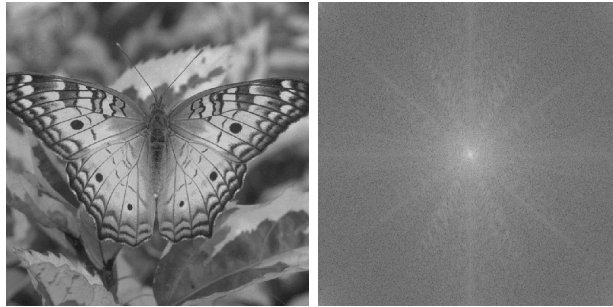
- skimage, versão 0.17.2
- cv2 (opencv), versão 4.0.1
- numpy, versão 1.19.2
- matplotlib, versão 3.3.2

Para executar o código, basta executar cada célula do notebook *project2.ipynb* na pasta src, com o cuidado de rodar a importação das bibliotecas pertinentes e as declarações das funções primeiro. A imagem de entrada pode ser escolhida pela variável “img_name” no início de cada célula, desde que esteja dentro da pasta “in”; as imagens de saída são salvas na pasta “out”.

3 Transformada rápida de Fourier

Para a aplicação dos filtros, primeiro as imagens serão convertidas para o domínio da frequência por meio da transformada rápida de Fourier (um

método mais eficiente do que a transformada de Fourier discreta usual). O método utilizado é fornecido pela biblioteca opencv e seu resultado pode ser visto na figura 1.



(a) Imagem original (b) Espectro de Fourier

Figura 1: Imagem original e seu espectro de Fourier

Para visualizar o resultado da transformada como observado na Figura 1b, calculamos a magnitude de cada elemento. Como pertencem ao conjunto dos números complexos possuem o formato $z = x + yi$ e sua magnitude é dada por $|z| = \sqrt{x^2 + y^2}$. Além disso, é preciso usar uma escala logarítmica, visto que há valores em intervalos com escalas bem distintas.

4 Filtros passa-baixa

Um filtro passa-baixa, como o nome sugere, é aquele que permite a “passagem” de baixas frequências, isto é, não altera as baixas frequências do sinal enquanto zera as maiores frequências a partir de um certo limiar. Aplicaremos dois filtros com essa característica e seus resultados serão discutidos nas próximas subseções.

4.1 Filtro passa-baixa ideal

O filtro passa-baixa ideal corresponde simplesmente a um círculo cujo raio é igual à frequência de corte desejada. Valores dentro do raio são iguais a 1, enquanto fora são iguais a 0. Assim, valores da função transformada próximos do centro, que correspondem a baixas frequências, não são modificados e os de maior frequência são zerados.

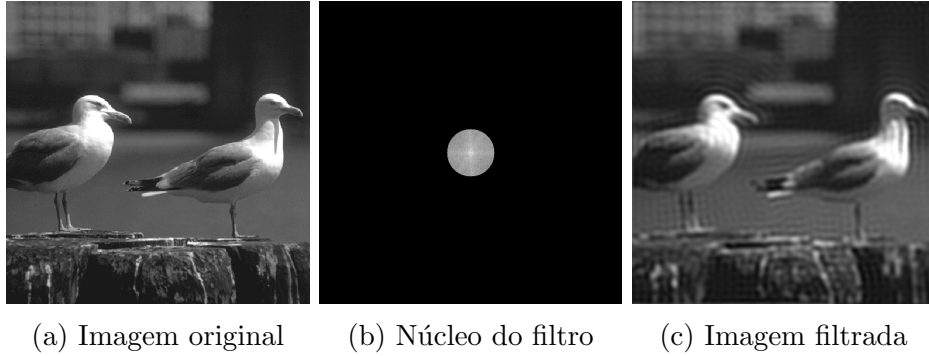


Figura 2: Imagem original, seu espectro sobreposto pelo filtro e a imagem resultante.

Para esse exemplo, foi escolhido um círculo de raio 40. Na figura 2b é possível visualizar o espectro de Fourier da imagem multiplicado pelo círculo do filtro, onde somente os elementos próximos do centro não são zerados. No resultado, é possível observar o efeito de um filtro passa-baixa após a aplicação da transformada inversa, que é de borrar a imagem.

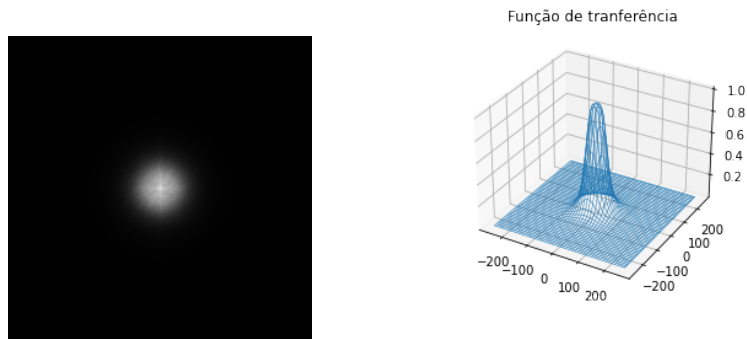
No entanto, tem-se um artefato indesejado, conhecido como “ringing”, pois parece-se com um efeito de vibração. Isso é causado pelo fenômeno de Gibbs, onde regiões próximas a uma descontinuidade apresentam um forte comportamento oscilatório, que afeta o resultado de filtros menos robustos. Para reduzir esse efeito, passaremos a utilizar o filtro Butterworth, que suaviza a região de corte da frequência.

4.2 Filtro passa-baixa de Butterworth

O filtro passa-baixa de Butterworth segue a seguinte função:

$$H(u, v) = \frac{1}{1 + [D(u, v)/D_0]^{2n}}$$

Onde $D(u, v)$ é a distância do ponto (u, v) até a origem, D_0 é a frequência de corte e n é a ordem do filtro. Para ordem 2 e $D_0 = 40$ temos a seguinte função:



(a) Núcleo do filtro (b) Plot em três dimensões da função

Figura 3: Visualização do filtro passa-baixa de Butterworth em 2D e 3D para ordem 2 e $D_0 = 40$.

Percebe-se que a região próxima à frequência de corte passa por uma transição mais suave, sem a descontinuidade presente no filtro ideal.



(a) Imagem original (b) Imagem filtrada

Figura 4: Imagem original e a filtrada pelo filtro passa-baixa de Butterworth de ordem 2 e $D_0 = 40$.

Note que o efeito de ringing não é mais perceptível e obtivemos uma imagem borrada como esperado. Iremos agora modificar os parâmetros do filtro e observar seus efeitos.

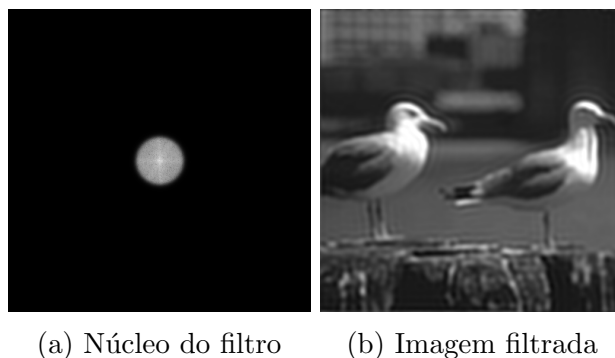


Figura 5: Núcleo do filtro e imagem filtrada pelo filtro passa-baixa de Butterworth de ordem 10 e $D_0 = 40$.

Pela figura 5, temos que o aumento da ordem do filtro reduz a suavidade da transição na região de corte, o que por sua vez intensifica o fenômeno de Gibbs já discutido. Para o propósito deste trabalho, utilizaremos um valor de 2 para a ordem a partir de agora.

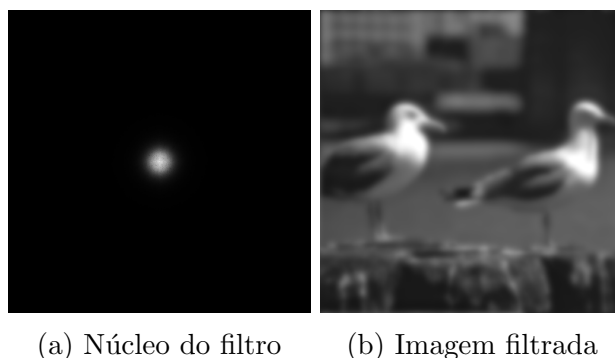


Figura 6: Núcleo do filtro e imagem filtrada pelo filtro passa-baixa de Butterworth de ordem 2 e $D_0 = 20$.

Intuitivamente, a redução na frequência de corte implicaria em uma imagem mais borrada, visto que mais informação está sendo retirada da imagem. De fato isso se observa pela figura 6, com raio $D_0 = 20$. Similarmente, um aumento no raio resulta em uma imagem menos borrada, mas pela concisão do código e deste relatório os resultados não serão apresentados.

5 Filtros passa-alta

Filtros passa-alta zeram ou reduzem valores de frequência abaixo de um certo limiar, enquanto deixam valores mais elevados intactos. Como discutido anteriormente, para reduzir os efeitos do fenômeno de Gibbs neste trabalho será aplicado somente o filtro de Butterworth.

5.1 Filtro passa-alta de Butterworth

A função do filtro passa-alta de Butterworth é dada por:

$$H(u, v) = \frac{1}{1 + [D_0/D(u, v)]^{2n}}$$

Os parâmetros são os mesmos do filtro passa-baixa.

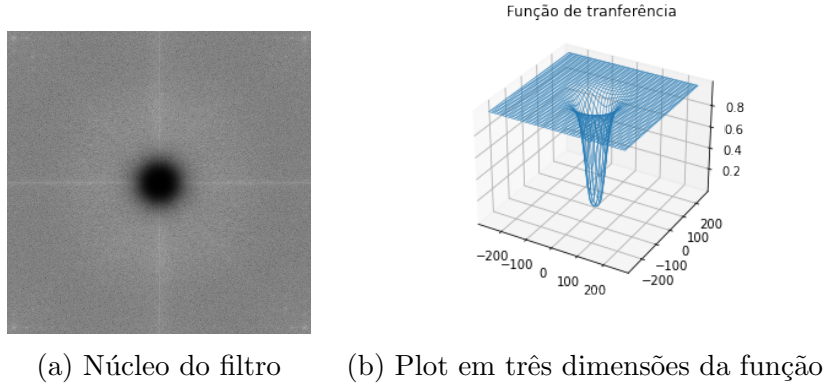
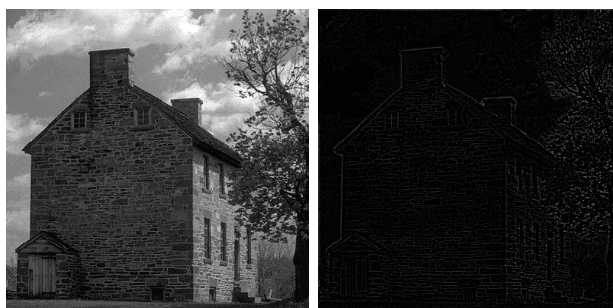


Figura 7: Visualização do filtro passa-alta de Butterworth em 2D e 3D para ordem 2 e $D_0 = 40$.

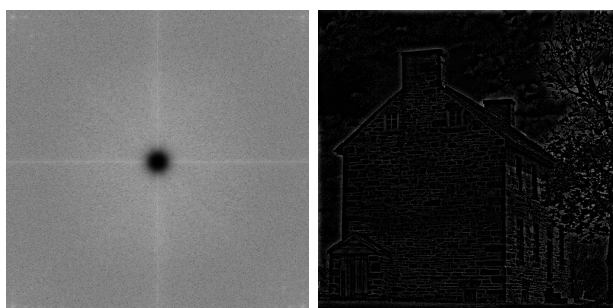
A função é praticamente o oposto do filtro passa-baixa, com as regiões próximas ao centro pretas (iguais a zero) e com uma transição suave perto da frequência de corte.



(a) Imagem original (b) Imagem filtrada

Figura 8: Imagem original e a filtrada pelo filtro passa-alta de Butterworth de ordem 2 e $D_0 = 40$.

Filtros passa-alta têm o efeito de detecção de bordas, como pode ser observado na figura 8. Note que não há limitação na orientação das bordas detectadas, como há em algumas das máscaras de convolução mais simples já vistas. Analisaremos agora o efeito da mudança do raio.



(a) Imagem original (b) Imagem filtrada

Figura 9: Imagem original e a filtrada pelo filtro passa-alta de Butterworth de ordem 2 e $D_0 = 20$.

Com a redução do raio, observamos que a o destaque das bordas fica mais intenso, mais sensível. Similarmente, esperamos que o aumento do raio deixe o filtro menos sensível.

6 Filtros passa-faixa

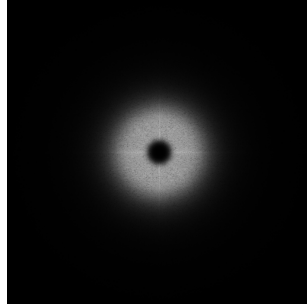
A última categoria de filtros que será explorada neste trabalho será de passa-faixas. Neste caso, temos uma frequência mínima e máxima de corte, onde valores abaixo do mínimo e acima do máximo são cortados.

6.1 Filtro passa-faixa de Butterworth

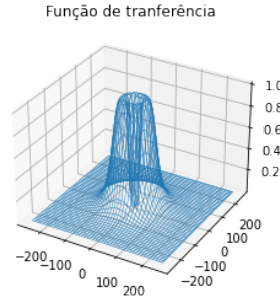
A fórmula é a seguinte:

$$H(u, v) = 1 - \frac{1}{1 + [(W \cdot D(u, v)) / (D^2(u, v) - D_0^2)]^{2n}}$$

O parâmetro novo, W , é igual à largura de banda, ou seja, o tamanho da faixa que passa pelo filtro. Os demais são iguais aos dos filtros de Butterworth anteriores.

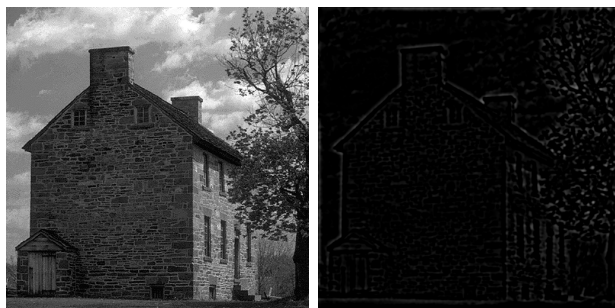


(a) Núcleo do filtro



(b) Plot em três dimensões da função

Figura 10: Visualização do filtro passa-faixa de Butterworth em 2D e 3D para ordem 2, $D_0 = 40$ e $W = 60$.



(a) Imagem original (b) Imagem filtrada

Figura 11: Imagem original e a filtrada pelo filtro passa-faixa de Butterworth de ordem 2, $D_0 = 40$ e $W = 60$.

Para o filtro passa-faixa, podemos exergar o efeito de D_0 similarmente com o de um filtro passa-alta, pois somente valores acima dele são considerados. No entanto, nesse caso há um limite superior, que é igual a $D_0 + W$. Assim, alterar valores de W controla quanta informação da imagem é preservada, sendo que valores baixos de W geram uma imagem menos definida.

7 Rotação da imagem

Aplicaremos agora uma rotação de 45° na imagem e observaremos o efeito dessa operação em seu espectro de Fourier. A função utilizada para realizar a rotação é a `rotate` da biblioteca `skimage`, que retorna a imagem rotacionada e com o tamanho ajustado para que não haja cortes em partes da figura.

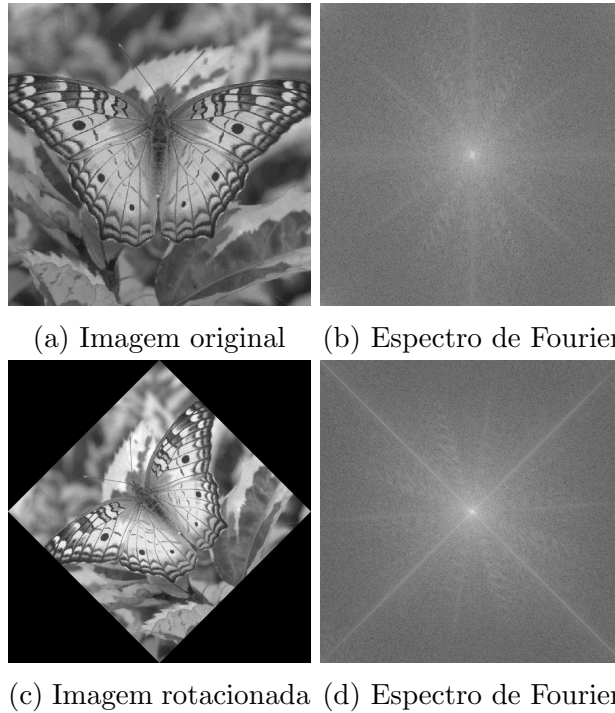


Figura 12: Comparação entre as imagens original e rotacionada com seus respectivos espectros de Fourier.

O espectro de Fourier resultante da imagem rotacionada é, salvo duas linhas que formam um “X”, igual ao espectro original rotacionado de 45° , assim como a imagem. Essa operação, portanto, não altera as frequências presentes na imagem, o que é de se esperar, uma vez que o conteúdo permanece o mesmo.

As linhas brancas que surgiram podem ser explicadas pelo fato da imagem gerada pelo processo de rotação possuir grandes áreas vazias, pois seu tamanho precisou ser aumentado para que nenhuma parte seja cortada. Assim, as regiões de transição entre a imagem e as áreas vazias apresentam uma grande diferença de valores, que se traduz em regiões de alta frequência após a transformada.

8 Compressão de imagem

A estratégia de compressão a ser utilizada neste trabalho consiste em definir um certo limiar e zerar elementos cuja magnitude esteja abaixo dele. O limiar inicial foi escolhido de modo que os maiores 10% dos valores fossem mantidos.

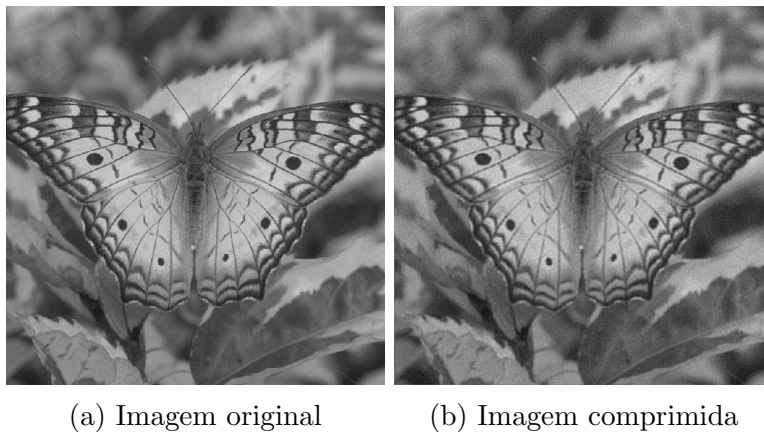


Figura 13: Comparação entre a imagem original e a comprimida.

Apesar de apenas 10% dos valores terem sido mantidos, a perda de qualidade da imagem foi bem baixa.

Tamanho:	155 KB (159.677 bytes)	Tamanho:	153 KB (157.080 bytes)
Tamanho em disco:	156 KB (159.744 bytes)	Tamanho em disco:	156 KB (159.744 bytes)
(a) Tamanho original		(b) Tamanho comprimido	

Figura 14: Comparação entre o tamanho da imagem original e a comprimida com limiar de 10%.

Apesar disso, para esse limiar o tamanho final da imagem não ficou muito diferente do original, possivelmente por causa da maneira com que os dados são salvos. Uma redução ainda maior, tomando apenas 1% dos valores, conseguiu uma boa redução do tamanho, mas comprometeu significativamente a qualidade da imagem.

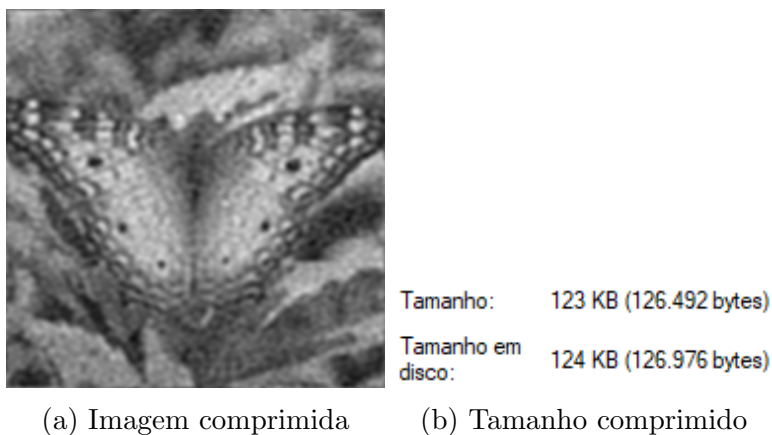


Figura 15: Imagem comprimida e seu tamanho com limiar de 1%.

9 Desempenho

Como dito anteriormente, a operação de convolução no domínio da frequência é mais barata do que a equivalente no domínio espacial, então espera-se que, desde que a aplicação da transformada e de sua inversa não seja muito custosa, o desempenho dos algoritmos utilizados neste trabalho seja superior.

Para comparar os métodos, foram aplicados filtros passa-baixa na imagem da borboleta. O algoritmo no domínio da frequência utilizado foi o de Butterworth, enquanto no domínio espacial foi utilizado o mesmo algoritmo do Trabalho 1, totalmente vetorizado para melhor desempenho. Os resultados obtidos foram: 0.073s para o filtro de Butterworth, 0.026s para a convolução normal com uma máscara 3x3, 0.085s para uma máscara 7x7 e 0.208s para uma 11x11.

Para máscaras pequenas, o custo de transformar a imagem foi mais elevado do que o ganho possível pela aplicação do filtro, mas a partir de um kernel de tamanho 7x7 esse custo extra já é coberto e o ganho de desempenho aumenta conforme o tamanho do kernel também aumenta.

10 Conclusão

Com este trabalho exploramos uma nova maneira de processar imagens, agora no domínio da frequência. Foi possível perceber as possíveis vantagens e desvantagens de escolher fazer as operações nos dois universos, além de introduzir o conceito de compressão de imagens.

Futuramente, seria interessante explorar mais abordagens para realizar a compressão, visto que com o método utilizado só foi possível obter uma redução relevante no tamanho com sacrifícios significativos na qualidade da imagem. Também não foi testado o desempenho dos algoritmos em questão para imagens com tamanho maior, especialmente para o caso da compressão.

Em suma, os resultados obtidos mostraram-se consistentes com a teoria apresentada e permitiram um melhor entendimento de conceitos que não são intuitivos, mas que valem a pena explorar pelas possibilidades que oferecem em processamento de imagens.