

Introdução ao Processamento Digital de Imagem: Trabalho 2

Guilherme Pereira Corrêa 198397

Outubro 2021

1 Introdução

O objetivo deste trabalho é estudar a aplicação da Transformada de Fourier em imagens digitais, convertendo-as para o domínio da frequência. Neste domínio é possível aplicar diversos filtros e transformações de uma maneira mais intuitiva e, as vezes, mais eficiente, do que no domínio espacial.

A entrega deste relatório foi feita no Google Classroom e, junto do relatório, se encontram os arquivos: as imagens utilizadas, o script em python trabalho2.py que aplica os métodos implementados e um jupyter notebook Trabalho2.ipynb, que foi utilizado para facilitar o teste das funções.

2 Execução do programa

O programa foi implementado em Python 3.9.6 e foram utilizadas as bibliotecas PIL, NumPy 1.20.3, matplotlib 3.4.2 e OpenCV 4.5.3. O código pode ser executado através do script trabalho2.py.

O script deve ser executado da seguinte maneira, onde os parâmetros entre colchetes são opcionais: `python3 trabalho2.py INPUT [-o OUTPUT] OPERATION [EXTRA_ARGUMENTS]`

- INPUT: caminho para a imagem de entrada que será usada nas operações realizadas
- OUTPUT: caminho opcional para a imagem de saída gerada, caso não seja especificado ela será salva como "output.jpg". O formato desse parâmetro é "-o OUTPUT", exemplo: `python3 trabalho2.py input.png -o output.png negative`
- OPERATION: operação a ser realizada, as operações disponíveis são descritas a seguir:
 - `fft-transform`: calcula a Transformada Rápida de Fourier da imagem de entrada e salva a magnitude da mesma como saída, esta operação não precisa de argumentos extras;
 - `lowpass-filter`: cria um filtro passa-baixa no espectro de frequência de acordo com o raio desejado, que deve ser especificado como parâmetro extra. Exemplo com raio 20: `python3 trabalho2.py lowpass-filter baboon.png 20`
 - `highpass-filter`: cria um filtro passa-alta no espectro de frequência de acordo com o raio desejado, que deve ser especificado como parâmetro extra. Exemplo com raio 50: `python3 trabalho2.py highpass-filter baboon.png 50`

- `bandpass-filter`: cria um filtro passa-faixa no espectro de frequência de acordo com o raios desejados, sendo eles o interno e o externo da máscara criada. Exemplo com raio externo 30 e raio interno 10: `python3 trabalho2.py bandpass-filter baboon.png 30 10`
- `bandreject-filter`: cria um filtro rejeita-faixa no espectro de frequência de acordo com o raios desejados, sendo eles o interno e o externo da máscara criada. Exemplo com raio externo 30 e raio interno 10: `python3 trabalho2.py -o baboon-out.png bandreject-filter baboon.png 30 10`
- `compress`: aplica uma compressão no espectro de Fourier e retorna a imagem recuperada. Esta compressão depende do parâmetro `keep`, que é um valor entre 0 e 1 que define qual a porcentagem dos maiores valores de magnitude do espectro de Fourier serão mantidos. Exemplo de uso: `python3 trabalho2.py baboon.png compress 0.01`
- **EXTRA_ARGUMENTS**: parâmetros adicionais das operações, exemplo: `python3 trabalho2.py input.png compress 0.5`

Saída A saída do programa é uma imagem PNG alterada conforme a operação selecionada. A imagem resultante será salva conforme especificado e será mostrada em uma janela pop-up.

Limitações O programa aceita imagens PNG e JPG em escalas de cinza.

3 Solução

3.1 Transformada rápida de Fourier (FFT)

Para converter a imagem digital de entrada para o domínio da frequência, foi utilizada a biblioteca NumPy, que implementa a FFT. Foram utilizados, em conjunto, os métodos `np.fft.fft2` e `np.fft.fftshift`, o primeiro faz a transformação da imagem no domínio espacial para o domínio de frequência, já o segundo método faz um shift das frequências iguais à zero para o centro da imagem.

Para fins de visualização do espectro de Fourier, foi calculada a magnitude de cada valor deste espectro, este calculo foi feito através da expressão:

$$magnitude = \log(abs(x))$$

Para recuperar a imagem espacial a partir da mesma no espectro de Fourier, foi feita a inversa da transformada. Tal operação também foi realizada com a biblioteca NumPy, através dos métodos `np.fft.ifftshift` e `np.fft.ifft2`, onde o primeiro desfaz o deslocamento ao centro das frequências zero e o segundo se trata do inverso da transformada de Fourier.

A Figura 1 mostra o resultado destas operações.

Transformada rápida de Fourier

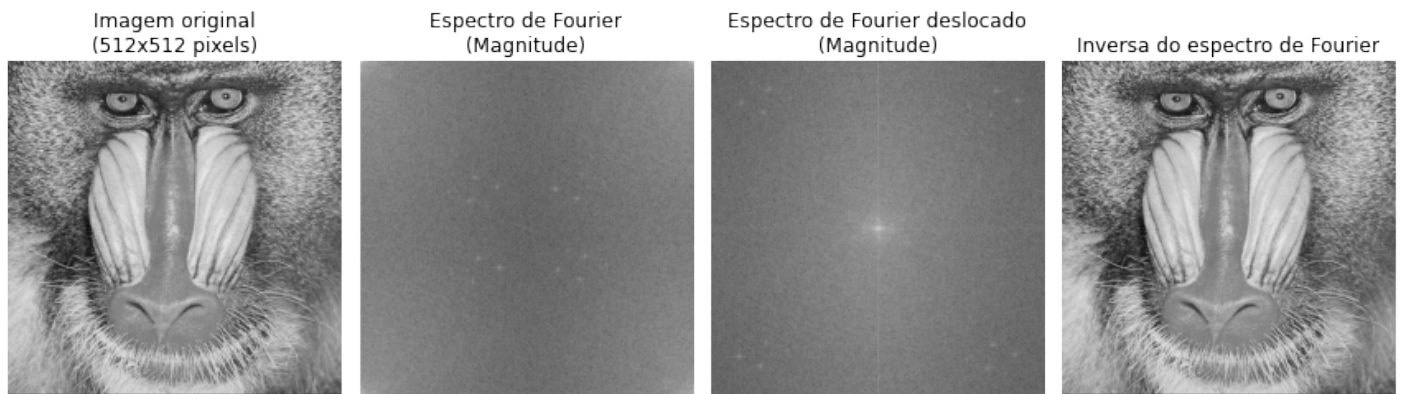


Figura 1: Transformação da imagem do domínio espacial para o domínio de frequência.

Nota-se que foi possível recuperar a imagem original após a aplicação da transformada, indicando que o método utilizado funcionou corretamente. Além disso, percebe-se a mudança no espectro de Fourier após deslocamento, onde as partes mais claras (frequências mais baixas) foram deslocadas para o centro da imagem, assim como o esperado.

3.2 Filtros passa-baixa

As frequências baixas em imagens estão relacionadas com regiões onde não há grandes variações de tom, sendo assim, filtros passa-baixa tendem borrar a imagem, já que regiões de borda serão atenuadas. A aplicação de filtros passa-baixa no espectro de Fourier é mais intuitiva do que no domínio espacial, já que, dado que as frequências baixas foram deslocadas para o centro, basta selecionar os pontos ao redor do centro e remover os que estão distantes do mesmo.

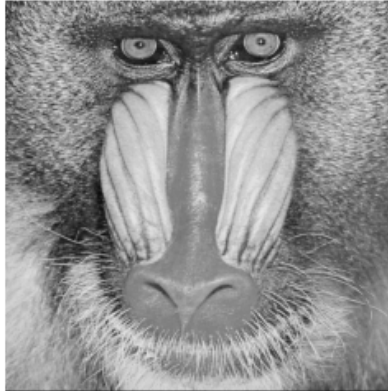
Isto foi feito utilizando-se uma máscara com as mesmas dimensões do espectro de Fourier de entrada, mas com um círculo de raio R no centro, onde o interior do mesmo é preenchido com 255 e todo seu exterior com 0. Desta forma, ao multiplicar termo a termo esta máscara pelo espectro de Fourier em questão e dividir por 255, as frequências menores serão mantidas enquanto as maiores serão descartadas. O círculo foi feito utilizando o método `cv.circle`.

O nível de filtragem depende do raio R do círculo da máscara, quanto maior este raio, maior será a faixa de frequência que será selecionada e a imagem não ficará tão borrada, de forma análoga, quanto menor o R , menor a faixa de frequência selecionada e mais borrada ficará a imagem.

Foram feitos experimentos para três valores diferentes de R : 10, 30 e 70. O resultado é mostrado na Figura 2.

Filtro passa baixa

Imagem original



Núcleo do filtro
passa baixa com raio = 10

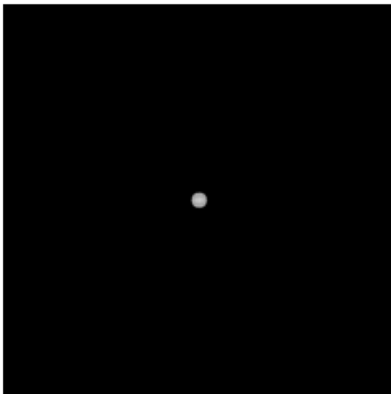


Imagem após filtragem
passa baixa com raio = 10



Núcleo do filtro
passa baixa com raio = 30

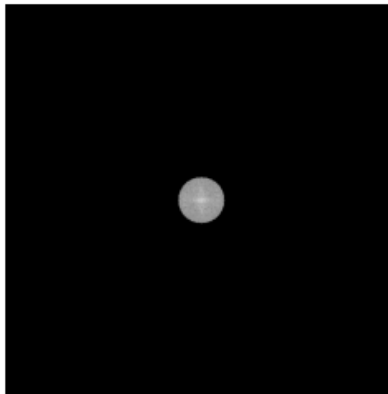


Imagem após filtragem
passa baixa com raio = 30



Núcleo do filtro
passa baixa com raio = 70

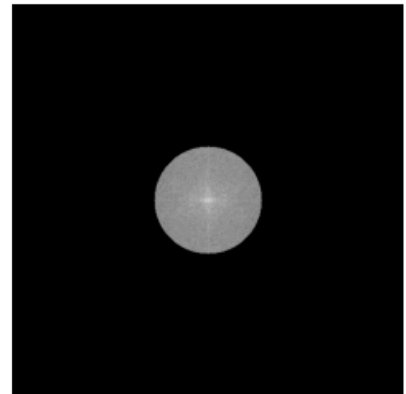


Imagem após filtragem
passa baixa com raio = 70

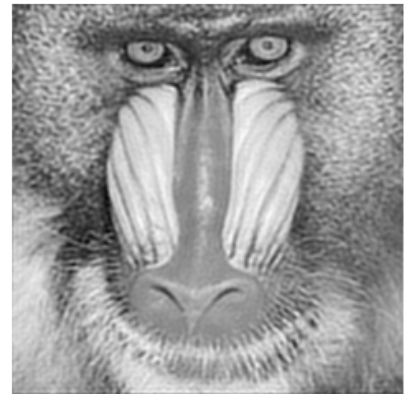


Figura 2: Aplicação de um filtro passa baixa no espectro de Fourier, para diferentes valores de raio da máscara.

Percebe-se que o resultado foi de acordo com o esperado, já que a imagem ficou muito borrada para um raio de máscara pequeno, e ficou praticamente igual para um raio de máscara maior.

3.3 Filtros passa-alta

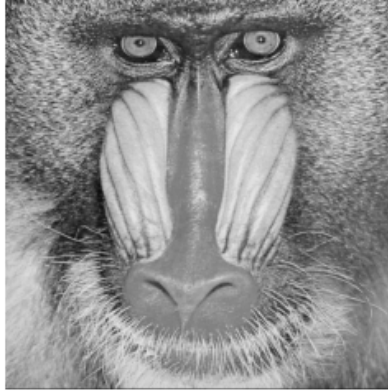
Ao contrário das baixas frequências, as altas frequências em imagens estão associadas a regiões onde há uma grande variação de tom, ou seja, regiões de borda. Assim como na aplicação de filtros passa baixa no espectro de Fourier, a aplicação de filtros passa alta nesse espectro também é mais intuitiva, sendo necessário somente selecionar os pontos mais distantes do centro, já que os mesmos correspondem a regiões de maior frequência.

A ideia de implementação deste filtro foi a mesma que a do filtro passa-baixa, sendo que neste caso o interior do círculo é preenchido com 0 e o exterior com 255. Sendo assim, o nível de filtragem também depende do raio R do círculo e, quanto maior este raio, mais as baixas frequências estarão sendo "excluídas", acarretando numa imagem resultante com linhas mais finas e mais ruído, o contrário é válido para raios menores.

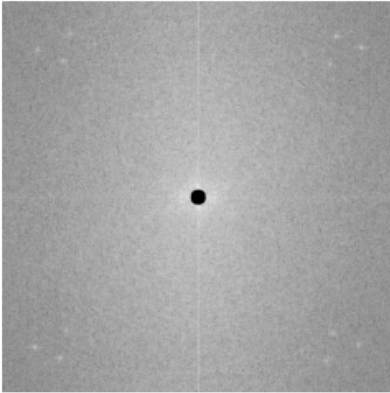
Foram feitos experimentos para os mesmos valores de R do experimento anterior. O resultado obtido pode ser observado na Figura 3

Filtro passa alta

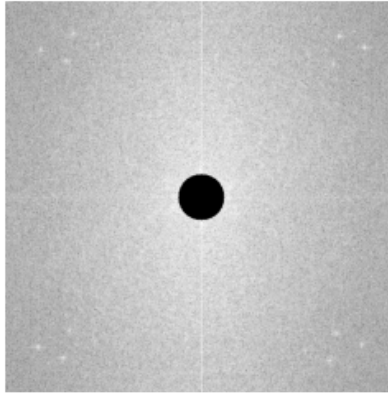
Imagem original



Núcleo do filtro
passa alta com raio = 10



Núcleo do filtro
passa alta com raio = 30



Núcleo do filtro
passa alta com raio = 70

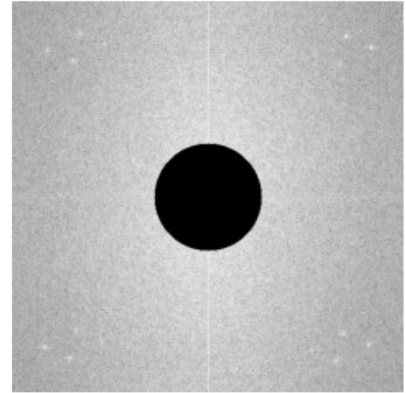


Imagem após filtragem
passa alta com raio = 10



Imagem após filtragem
passa alta com raio = 30

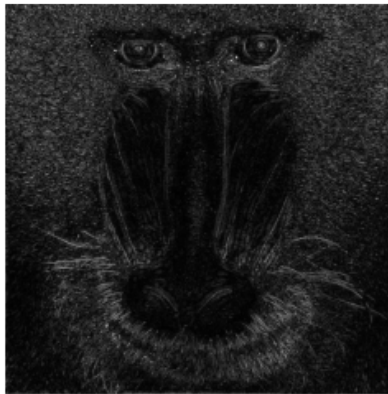


Imagem após filtragem
passa alta com raio = 70



Figura 3: Aplicação de um filtro passa alta no espectro de Fourier, para diferentes valores de raio da máscara.

O resultado, neste caso, também atendeu as expectativas, já que a imagem ficou com linhas mais finas e menos ruído para raios maiores, e linhas mais grossas com mais ruído para raios menores.

3.4 Filtros passa-faixa

Os filtros passa faixa caracterizam-se, assim como o nome diz, por atenuarem todo o espectro de frequência, exceto por uma faixa específica, portanto, o resultado depende muito de qual faixa foi selecionada.

A implementação deste filtro também foi parecida com a implementação dos filtros anteriores, onde a máscara foi obtida ao subtrair um círculo maior de um menor, obtendo-se um disco que corresponde a faixa de frequência que será selecionada. Como a faixa de frequência selecionada varia com o valor dos raios, é difícil prever o resultado, todavia, como o raio interior será sempre maior que zero (caso contrário seria um filtro passa baixa), então as frequências próximas de zero serão sempre atenuadas e a imagem resultante provavelmente se assemelhará à resultante do filtro passa alta.

Foram feitos experimentos com três tipos diferentes de faixa e o resultado pode ser observado na Figura 4

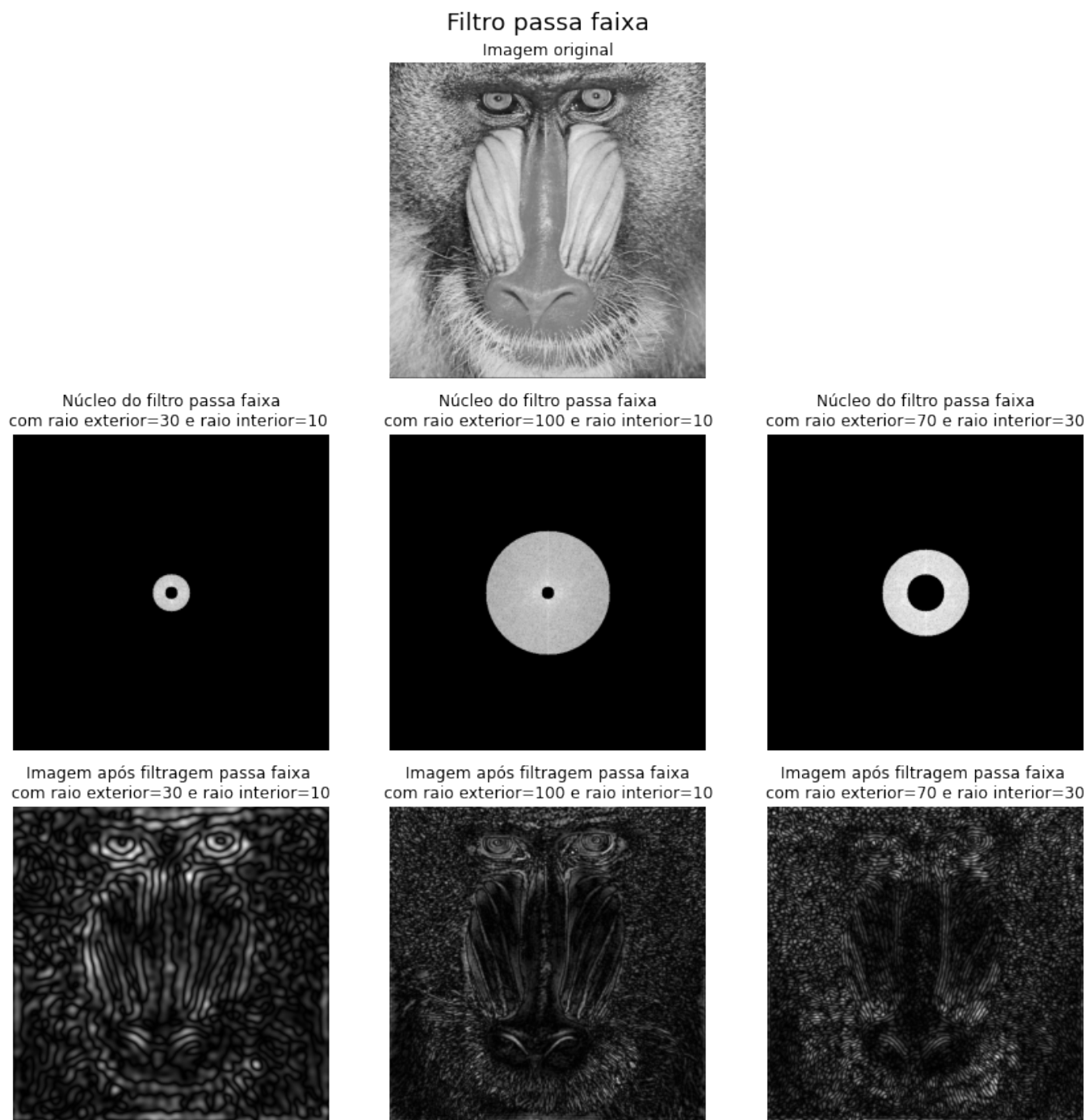


Figura 4: Aplicação de um filtro passa faixa no espectro de Fourier, para diferentes faixas de frequência.

Observando-se as imagens obtidas, percebe-se que, como esperado, elas assemelham-se com as resultantes de um filtro passa alta. Porém, o efeito é diferente, onde formas relativamente peculiares foram destacadas, principalmente no caso do filtro de raios 30 e 10, onde uma faixa bem pequena de frequência foi selecionada.

O filtro de raios 100 e 10, por sua vez, teve um resultado muito semelhante ao do filtro passa alta, onde as regiões de borda foram destacadas, o que faz sentido, dado que a faixa de frequência selecionada foi larga, atenuando-se somente as frequências bem baixas.

Já o filtro de raios 70 e 30 teve um resultado difícil de se prever, sendo até relativamente difícil de reconhecer sobre o que se trata a imagem, principalmente na região dos olhos do babuíno.

3.5 Filtros rejeita-faixa

Assim como o nome diz, filtros rejeita-faixa atenuam somente uma faixa de frequência do espectro, ou seja, o contrário dos filtros passa-faixa, sendo assim, o resultado também depende bastante da faixa de frequência selecionada.

A implementação também foi baseada em máscaras e, como este filtro é o contrário do filtro passa-faixa, foi criada uma máscara inteira com 255 e então um disco passa-faixa foi subtraído da mesma. Assim como o filtro anterior, o resultado deste também é difícil de prever, mas, como o raio interior é sempre maior que zero (caso contrário seria um filtro passa alta), então as frequências ao redor de zero sempre passarão, o que deve acarretar em um resultado mais semelhante ao do filtro passa-baixa.

Também foram feitos três experimentos, com os valores dos raios iguais aos do experimento anterior. O resultado obtido pode ser observado na Figura 5

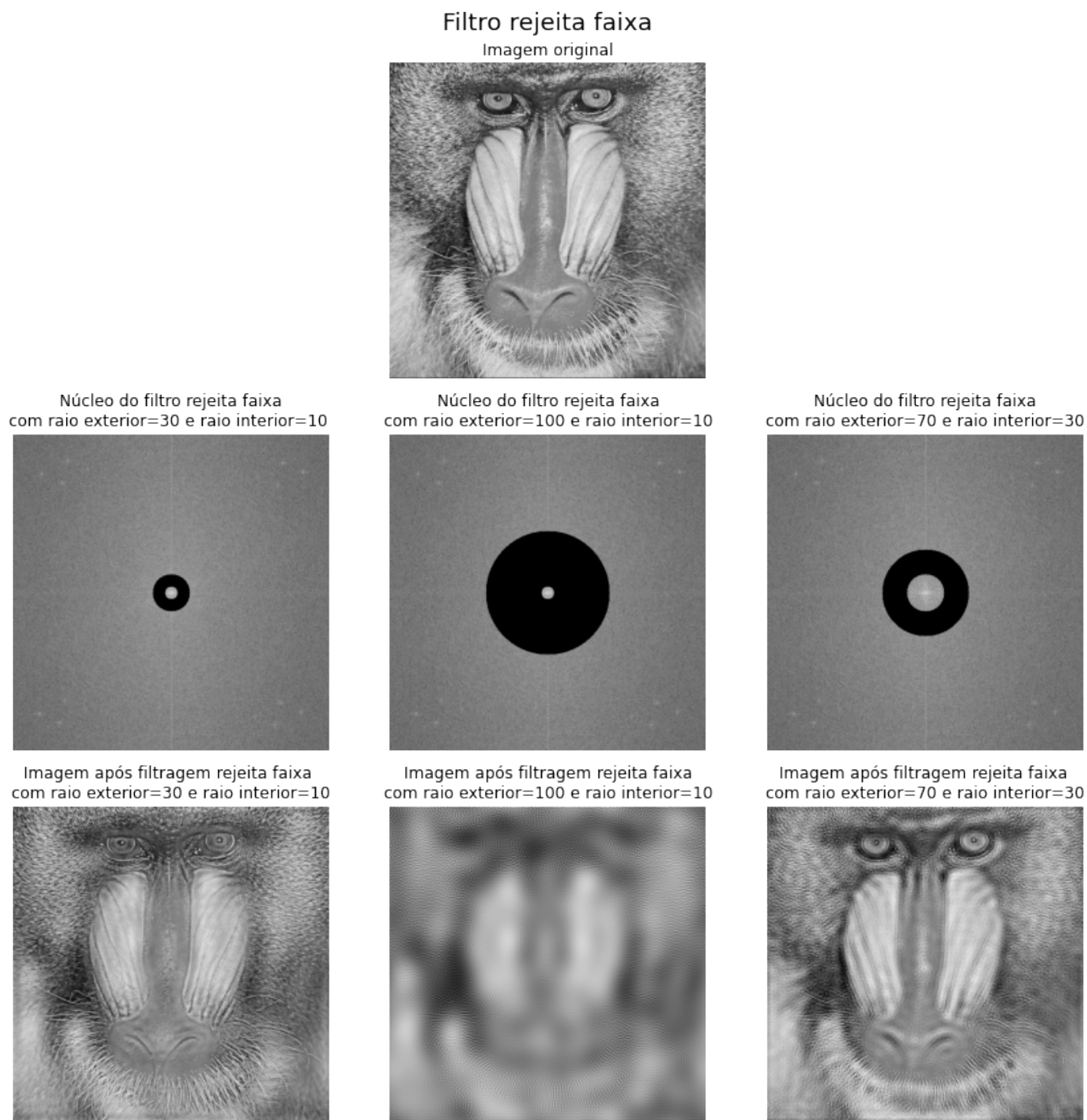


Figura 5: Aplicação de um filtro rejeita faixa no espectro de Fourier, para diferentes faixas de frequência.

Percebe-se que, assim como o esperado, o resultado assemelha-se com o obtido dos filtros passa baixa, porém com algumas notáveis diferenças, que são difíceis de prever. O primeiro filtro, em especial, mostrou um resultado muito interessante, onde a imagem aparenta estar dentro d'água, mostrando que resultados muito interessantes podem ser obtidos tanto com filtros passa-faixa quanto com filtros rejeita-faixa para pequenas faixas.

O segundo filtro, com raios de 100 e 10, resultou em uma imagem totalmente borrada, mostrando que as

frequências responsáveis pela "nitidez" estão concentradas nesta faixa, o que é curioso, dado que elas não são as maiores frequências da imagem.

Já o segundo filtro, de raios 70 e 30, apresentou um resultado parecido com uma mistura entre um filtro passa-baixa e um passa-alta.

3.6 Compressão

A compressão de imagens é uma das aplicações mais úteis da FFT, permitindo diminuir o tamanho de imagens sem perder muitas informações da imagem original. A compressão é feita ao selecionar somente os maiores valores de magnitude no espectro de Fourier e, como grande parte dos valores possui uma magnitude muito pequena, é possível economizar muita memória após esta remoção.

A estratégia adotada nesta solução foi ordenar os valores de magnitude no domínio da frequência e selecionar aqueles que pertencem aos X% maiores, isso foi feito ao ordenar a magnitude do espectro de Fourier e, após esta ordenação, definir um limiar definido a partir da porcentagem de seleção desejada. Após a definição deste limiar, uma máscara é criada e todos os pontos onde a magnitude é menor que o limiar são zerados.

Como a maior parte dos pontos possuem uma magnitude muito baixa, é esperado que mesmo selecionando uma pequena parte das magnitudes, como por exemplo 1%, ainda seja possível recuperar a imagem original. Todavia, é esperado que detalhes da imagem sejam perdidos conforme esta seleção vai ficando mais restrita.

Para observar este efeito, foram feitos experimentos com três imagens diferentes e três valores diferentes de porcentagem de seleção, começando ao selecionar as 1% maiores magnitudes e diminuindo até 0,01%. Além disso, foi plotado o histograma das imagens em todas as situações. O resultado pode ser observado na Figura 6.

Compressão

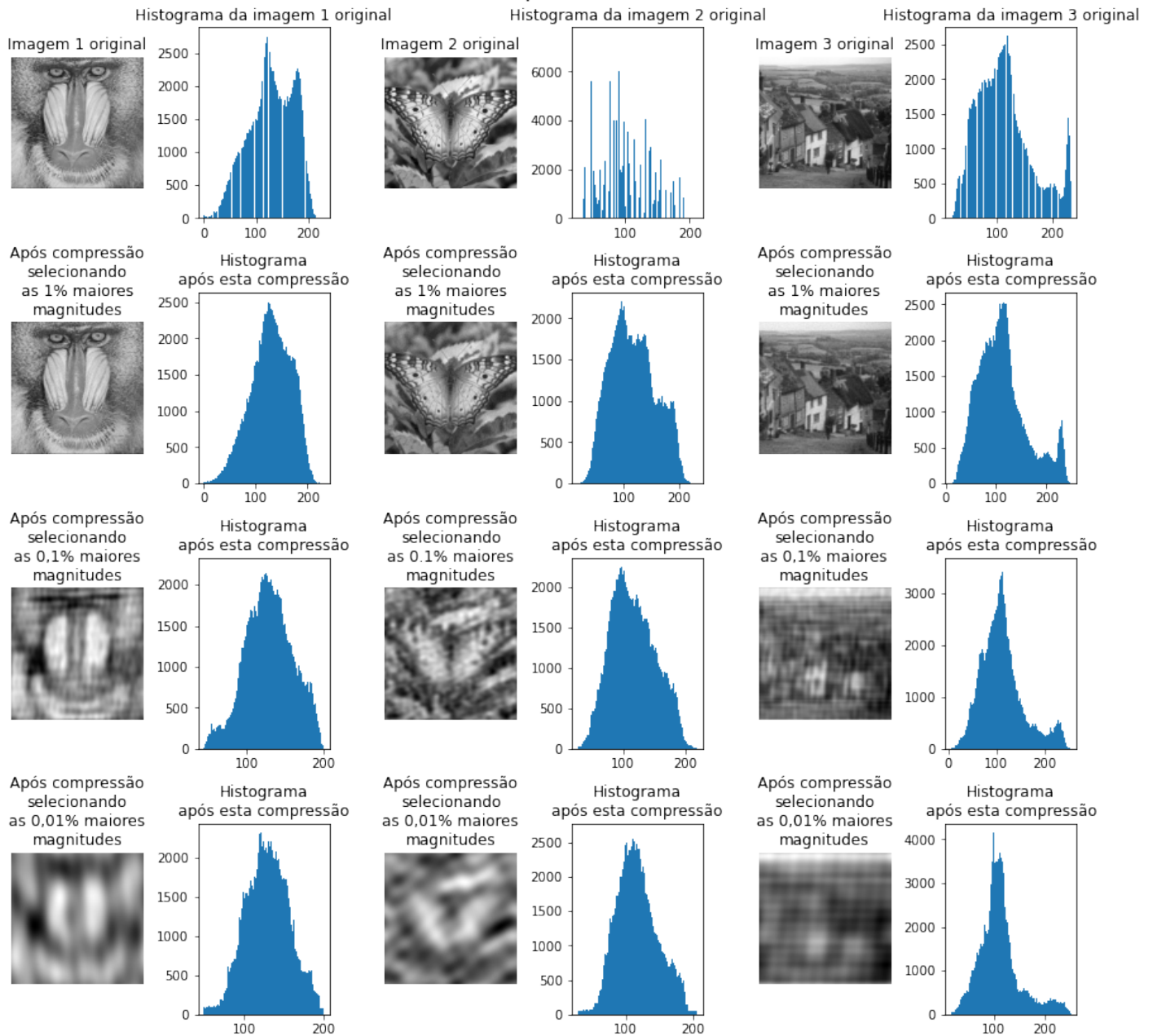


Figura 6: Compressão de três imagens para diferentes seleções das magnitudes no espectro de Fourier.

Como esperado, percebe-se que as imagens são quase totalmente recuperadas ao usar um limiar de 1%, onde alguns detalhes perdidos podem ser notados somente caso um zoom seja feito. Conforme este limiar vai ficando mais restrito, detalhes das imagens vão sendo perdidos, fazendo com que as imagens fiquem extremamente embaçadas quando o limiar é de 0,01%. Este resultado faz sentido, já que ao definir o limiar de magnitude como um número relativamente alto, mesmos os poucos valores que possuem uma magnitude alta, normalmente responsáveis pelos detalhes das imagens, estarão sendo atenuados. Em especial, nota-se como a imagem da cidade fica borrada com mais facilidade, o que se deve ao fato dela possuir uma alta frequência mais abundante.

Os histogramas, por sua vez, mostram como a compressão diminui o número de "picos" do histograma,

fazendo com que os tons de cinza fiquem concentrados ao redor de um único valor, ou seja, representa uma atenuação das regiões de borda e o borramento da imagem.