Processamento de Imagens

Prof. MSc. Daniel Menin Tortelli

e-mail: danielmenintortelli@gmail.com
Skype: daniel.menin.tortelli

Site: http://sites.google.com/site/danielmenintortelli/home

Transformadas de Fourier e Processamento no Domínio da Frequência

Histórico

- Século XVII: matemático e físico francês Jean Baptiste Joseph Fourier (1768-1830) demonstrou que qualquer forma de onda pode ser representada por uma somatória de senóides e cossenóides de diferentes frequências, amplitudes e fases.

- Transformada de Fourier: decompõe um sinal em suas componentes elementares seno e cosseno.

- Aplicação inicial: problemas da condução do calor (lei da condução térmica).

Conceitos

- A Transformada de Fourier (FT) decompõem uma imagem em seus componentes real e imaginário (número complexo) que é a representação da imagem no domínio da frequência.

- Se o sinal de entrada é uma imagem, então o número de frequências no domínio da frequência é igual ao número de pixels na imagem ou no domínio espacial.

- A saída da Transformada de Fourier é um número complexo que possui uma faixa de valores muito maior do que os valores da imagem no domínio espacial.

- Dessa forma, para representar esses valores de maneira correta, eles são armazenados como valores de ponto flutuante.

Conceitos

- A Transformada Discreta de Fourier (DFT) é uma forma específica da análise de Fourier para converter uma função (frequentemente no domínio do tempo ou espacial) em outro (domínio da frequência).

- A Transformada Rápida de Fourier (FFT) é uma implementação mais eficiente da DFT e é usada no processamento de imagens.

- FFT é usada para converter uma imagem no domínio espacial para o domínio da frequência.

- Aplicar filtros em imagens no domínio da frequência é computacionalmente mais rápido do que no domínio espacial.

Conceitos

- Os filtros de frequência processam uma imagem no domínio da frequência.

- *A imagem é transformada para o domínio da frequência usando a Transformada de Fourier, multiplicada pela função de filtro e, depois, retransformada (usando a Transformada Inversa de Fourier) para o domínio espacial.*

- Atenuar altas frequências resulta em uma imagem mais suave (borrada, desfocada) no domínio espacial.

- Atenuar baixas frequências evidencia as bordas da imagem.

Conceitos

- Isso é baseado na propriedade da Transformação de Fourier em que a convolução no domínio espacial se torna multiplicação no domínio da frequência.

- A transformação e a transformação inversa podem ser feitas como processos independentes.

$$G(u,v)=F(u,v)H(u,v),$$

$$g(m,n)\overset{\Im}{\Leftrightarrow}G(u,v),$$

G(u,v) é a imagem resultante no domínio da frequência.
F(u,v) é a imagem original no domínio da frequência (espectro de Fourier).
H(u,v) é a função de filtragem (filtro/máscara) no domínio da frequência.

g(m,n) é a imagem resultante no domínio espacial, após aplicação da Transformada Inversa de Fourier.

Filtros no Domínio da Frequência

- Esse tipo de filtragem requer que a imagem seja convertida para um outro tipo de dado antes da aplicação do filtro em si.
- A imagem formada por pixels é transformada (usando a Transformada de Fourier) em uma soma de senos e cossenos de diferentes frequências e amplitudes.
- Somente depois pode-se aplicar o filtro desejado sobre a imagem.
- O filtro então atua selecionando as frequências e amplitudes que deseja-se manter ou remover na imagem.
- Usa-se a Transformada Inversa de Fourier sobre o conjunto de frequências resultante para obter a imagem filtrada novamente no domínio espacial.

A Transformada de Fourier

- A transformação permite representar uma curva ou sinal em termos de funções sinusoidal.

- Ela realiza uma soma de funções de senos e cossenos de diferentes frequências e amplitudes.

- A TF pode ser usada nas formas **contínua** e **discreta**.

- A forma discreta é a mais utilizada no Processamento de Imagens.

A Transformada Discreta de Fourier 2D e sua inversa

- A Transformada Discreta de Fourier (DFT) 2D é dada pela equação:

$$F(u,v)=\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)e^{-j2\pi(ux/M+vy/N)}$$

sendo ***f(x, y)*** uma imagem digital de tamanho **M** x **N**.

- Dada a transformada ***F(u, v)***, podemos obter ***f(x, y)*** utilizando a Transformada Discreta de Fourier Inversa (IDFT):

$$f(x,y)=\frac{1}{MN}\sum_{u=0}^{M-1}\sum_{v=0}^{N-1} F(u,v)e^{j2\pi(ux/M+vy/N)}$$

(u,v) são as coordenadas no domínio da frequência

(x,y) são as coordenadas do pixel no domínio espacial.

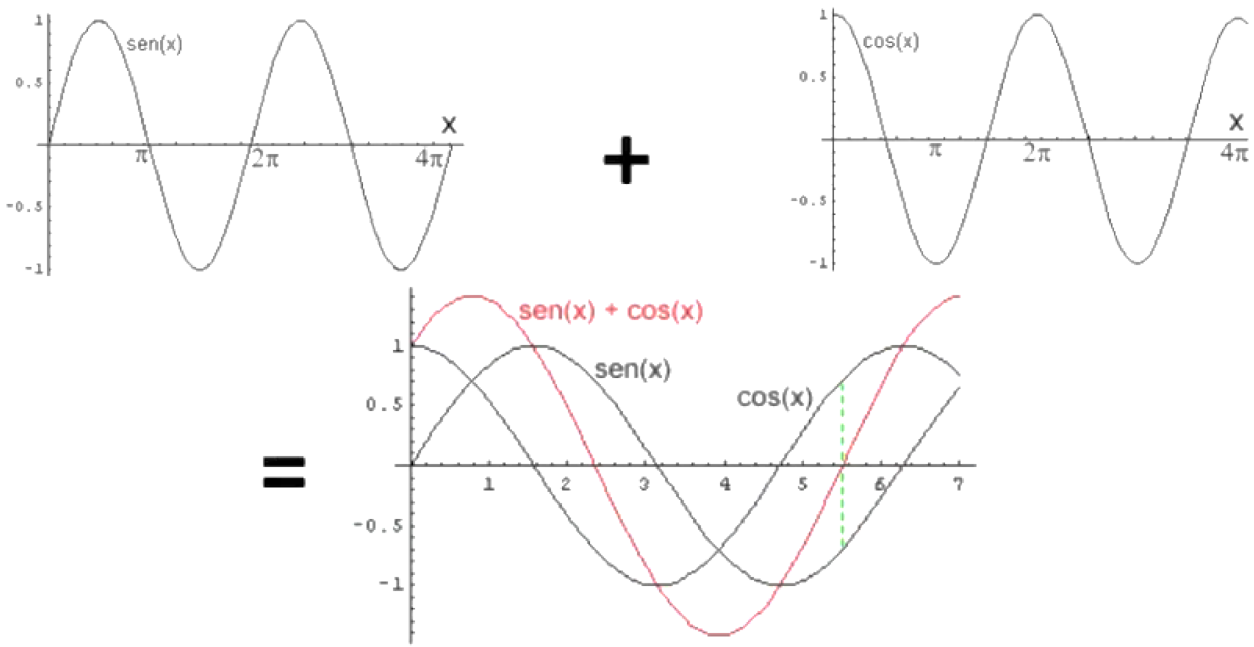
- As equações acima constituem o par de T*ransformadas Discretas de Fourier* 2D.

A Transformada de Fourier

- Qualquer função ***f(x)*** pode, segundo Fourier, ser escrita na forma da soma de uma série de funções seno e cosseno da seguinte forma:

$$f(x) = a_0 + a_1 sen(x) + a_2 sen(2x) + a_3 sen(3x) + \ \ldots b_1 cos(x) + b_2 cos(2x) + b_3 cos(3x) + \ \ldots$$

onde **a** e **b** são amplitudes das funções seno e cosseno, respectivamente.



A Transformada Rápida de Fourier 2D

- No Matlab, a Transformada de Fourier pode ser feita em sinais 2D, isto é, matrizes (imagens), usando a função ***fft2(f)***:

F = fft2(f)	
Entrada	**f** – é uma matriz M x N representando uma imagem.
Saída	**F** – é uma matriz M x N representando uma imagem após a aplicação da Transformada de Fourier.

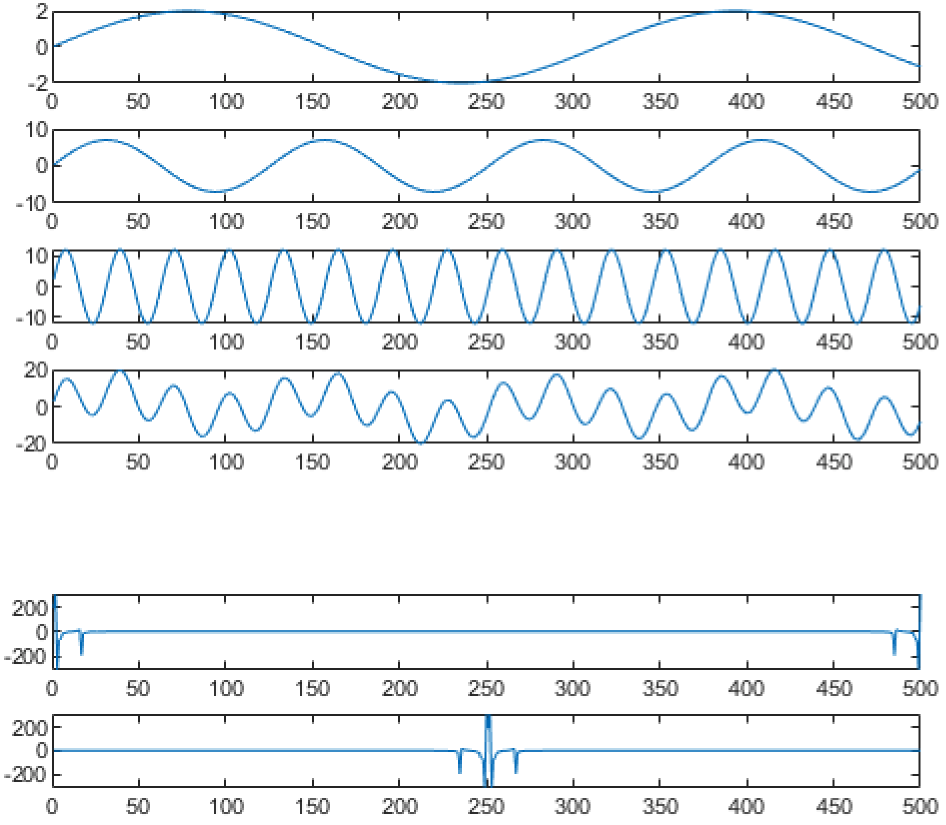
A Transformada Rápida de Fourier 2D

- No Matlab, a função que centraliza a Transformada de Fourier é a função ***fftshift(F1)***:

F2 = fftshift(F1)	
Entrada	**F1** – é uma matriz M x N representando uma imagem após a aplicação da Transformada de Fourier. A função irá calcular o centro do vetor e trocar de lugar as porções da direita e da esquerda.
Saída	**F2** – é uma matriz M x N representando uma imagem após a aplicação da operação ***shift***.

A Transformada Rápida de Fourier 2D

```
mn = 1:500;
sen1 =  2 * sin(mn/50);
sen2 =  7 * sin(mn/20);
sen3 = 12 * sin(mn/5);

sen_Soma = sen1 + sen2 + sen3;
Hseno = fft(sen_Soma);

subplot(6,1,1); plot(mn, sen1);
subplot(6,1,2); plot(mn, sen2);
subplot(6,1,3); plot(mn, sen3);
subplot(6,1,4); plot(mn, sen_Soma);
figure;
subplot(6,1,5); plot(mn, Hseno);
subplot(6,1,6); plot(mn, fftshift(Hseno));
figure;
```



A Transformada Inversa de Fourier 2D

- No Matlab, a Transformada Inversa de Fourier pode ser feita em sinais 2D, isto é, matrizes (imagens), usando a função ***ifft2(f)***:

f = ifft2(F)	
Entrada	**F** – é uma matriz M x N representando uma imagem na qual já foi aplicada a Transformada de Fourier.
Saída	**f** – é uma matriz M x N representando uma imagem após a aplicação da Transformada Inversa de Fourier.

Filtragem usando a Transformada de Fourier

- O teorema da convolução estabelece que a convolução de uma máscara (filtro) em uma imagem (filtragem espacial) equivale a multiplicação da Transformada de Fourier pela transformada da máscara (filtragem no espectro).

- A localização das frequências após a aplicação da Transformada de Fourier na imagem, dizem respeitos às Baixas e Altas frequências.

- As baixas frequências localizam-se no centro da imagem e representam as mudanças suaves de intensidade na imagem.

- As altas frequências localizam-se afastadas do centro da imagem e representam as mudanças bruscas (bordas) de intensidade na imagem.

Filtragem no Domínio da Frequência

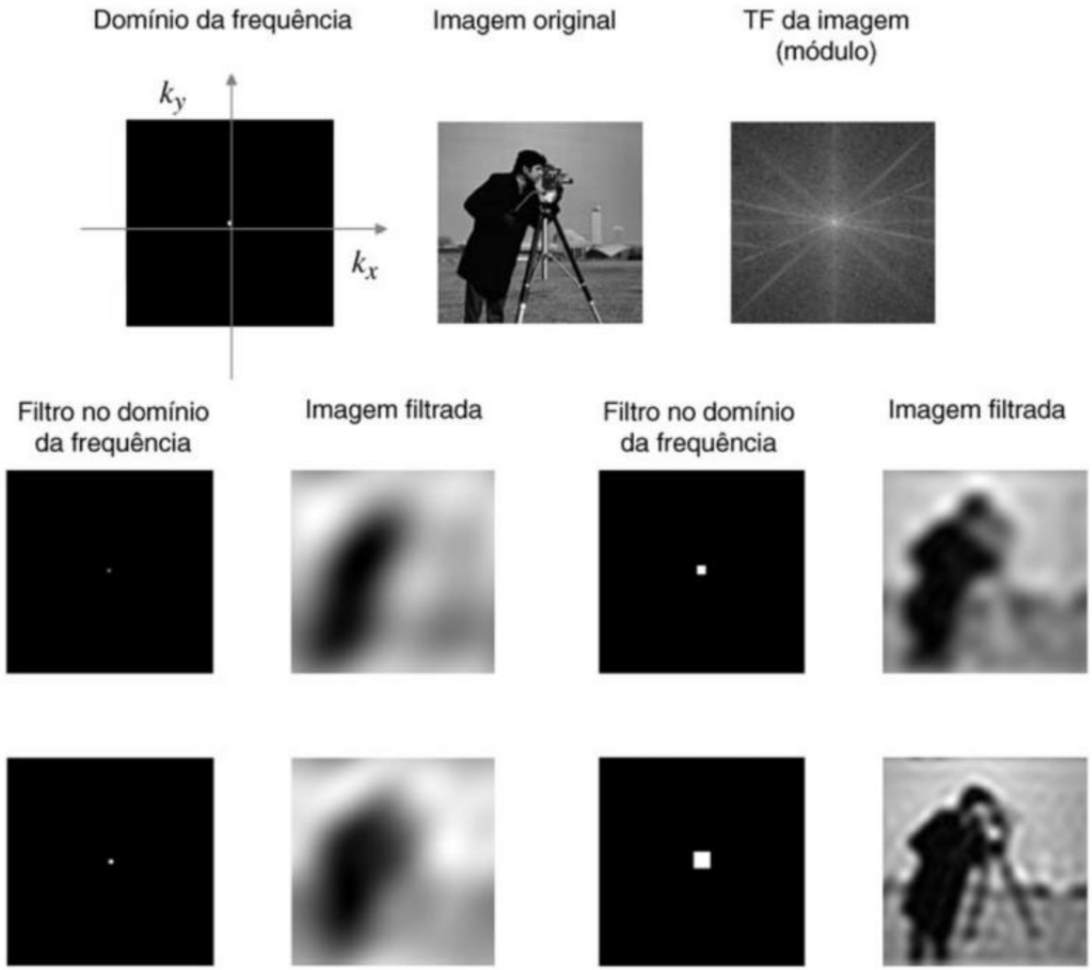
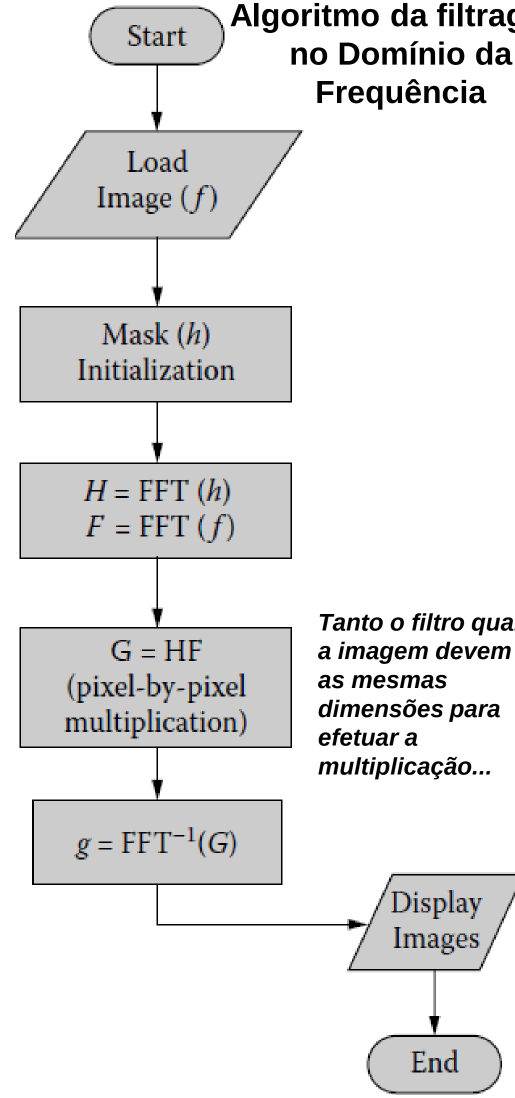


Ilustração simples de filtragem no domínio da frequência. A transformada de Fourier (TF) da imagem original é multiplicada pelos filtros indicados (branco indica pares de frequência que são preservados, e preto, totalmente removidos). Uma transformada de Fourier inversa nos leva de volta ao domínio espacial e a imagem filtrada é exibida no lado direito.

Algoritmo da filtragem no Domínio da Frequência



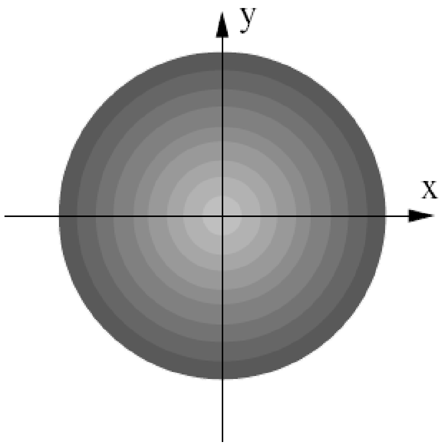
Tanto o filtro quanto a imagem devem ter as mesmas dimensões para efetuar a multiplicação...

O Espectro de Fourier

- O **Espectro de Fourier** obtido a partir da DFT bidimensional, além de prover informações sobre a orientação das estruturas presentes na imagem de entrada, efetua um mapeamento das variações nos tons de cinza dos pixels.

- Um número pequeno de variações de intensidade de cinza em um determinado espaço indica a presença de regiões de baixa frequência, enquanto um numero maior de variações indica a presença de regiões de frequência alta na imagem.

- Pode-se resumir a interpretação do espectro de Fourier para uma função bidimensional por meio do plano mostrado a seguir:



- A presença de componentes em regiões mais claras indica a existência de frequências baixas na imagem de entrada, enquanto a presença de componentes em regiões mais escuras indica a existência de frequências altas.

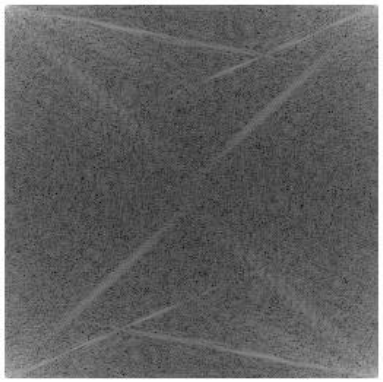
Interpretação do espectro de Fourier resultante da aplicação da DFT bidimensional

Filtragem no Domínio da Frequência

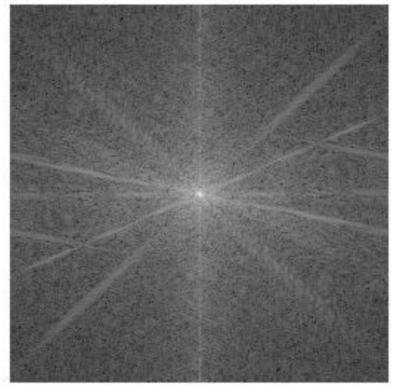
```
A = imread('cameraman.tif'); % Lê a imagem

FT = fft2(A); % Calcula FT
FT_centred = fftshift(FT); % Obtém a versão centrada

figure; imshow(A); title('Imagem Original'); % Exibe a imagem original
figure; imshow(log(1+abs(FT)),[]); title('Espectro da TF'); % Exibe o módulo da TF (escala log)
figure; imshow(log(1+abs(FT_centred)),[]); title('Espectro centrado da TF'); % Exibe o módulo da TF centrada (escala log)


Im1 = abs(ifft2(FT)); % Calcula FFT inversa
figure; imshow(Im1,[]); title('FFT Inversa');

Im2 = abs(ifft2(FT_centred)); % Calcula FFT inversa da imagem centrada
figure; imshow(Im2,[]); title('FFT Inversa centrada');

% Constrói filtro no domínio da frequência
figure;
[xd,yd] = size(A);
x = -xd./2:xd./2-1;
y = -yd./2:yd./2-1;
[X,Y] = meshgrid(x,y);
sigma = 32;
arg = (X.^2+Y.^2)./sigma.^2;
frqfilt = exp(-arg);

imfilt1 = abs(ifft2(frqfilt.*FT)); % Filtro centrado & espectro não centrado
imfilt2 = abs(ifft2(frqfilt.*FT_centred)); % Filtro centrado em espectro centrado

%Exibe resultados
figure; imshow(frqfilt,[]); title('Filtro');
figure; imshow(imfilt1,[]); title('Imagem não centrada filtrada');
figure; imshow(imfilt2,[]); title('Imagem centrada filtrada');
```

Filtragem no Domínio da Frequência

Espectro da TF



Espectro centrado da TF



Filtro

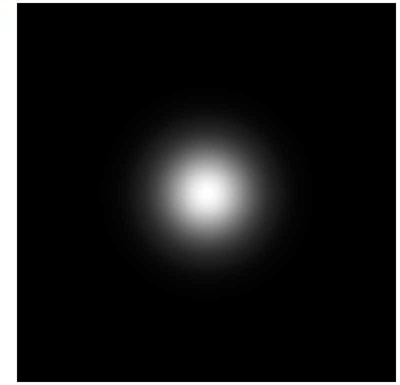
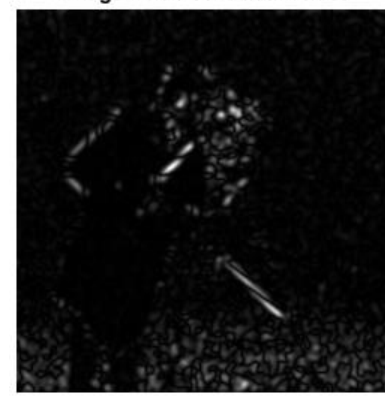


Imagem Original

Imagem não centrada filtrada



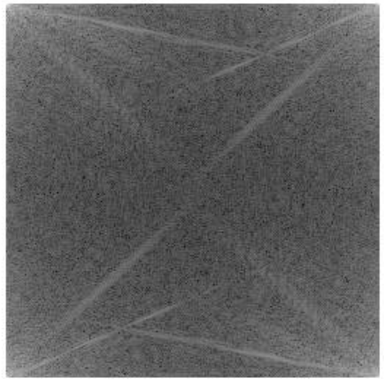
FFT Inversa

FFT Inversa centrada

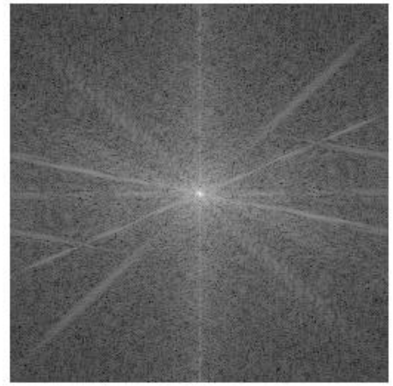
Imagem centrada filtrada

Filtragem no Domínio da Frequência

Espectro da TF



Espectro centrado da TF



Filtro

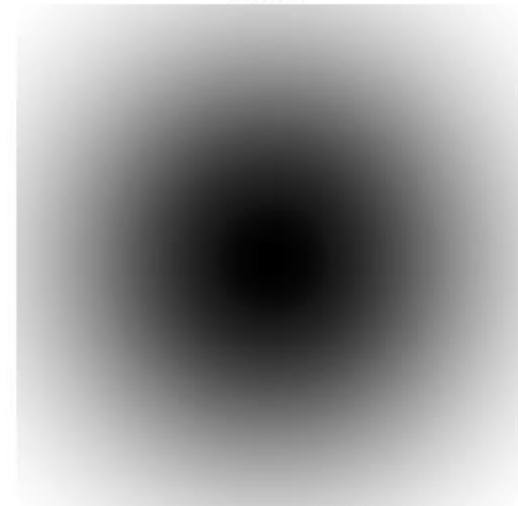


Imagem Original

FFT Inversa

FFT Inversa centrada

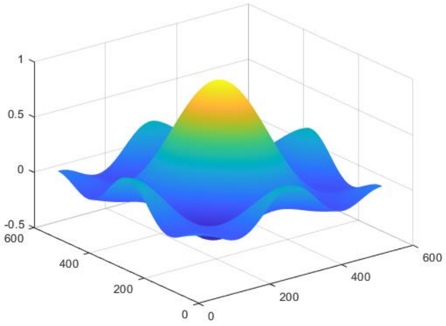
Imagem não centrada filtrada

```
% Constrói filtro no domínio da frequência
figure;
[xd,yd] = size(A);
x = -xd./2:xd./2-1;
y = -yd./2:yd./2-1;
[X,Y] = meshgrid(x,y);
sigma = 100; % Alterar o raio
arg = (X.^2+Y.^2)./sigma.^2;
frqfilt = exp(-arg).*-1; % Inverte o filtro
```

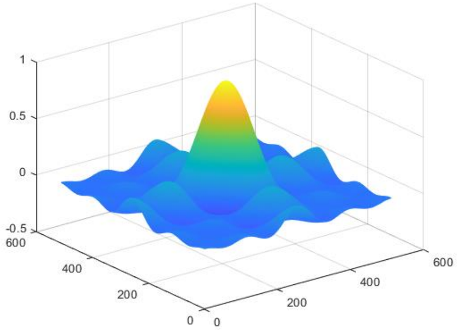
Filtragem no Domínio da Frequência (Remoção de ruído Gaussiano)

```
h1 = ones(3)/9; % Filtro da Média (Dominio Espacial)
h2 = fspecial('gaussian', 5,5); % Filtro Gaussiano (Dominio Espacial)

x = imread('lenna.jpg'); % Lê uma imagem (Dominio Espacial)
x = rgb2gray(x); % Converte imagem em escala de cinza
[r,c] = size(x); % Salva dimensões da imagem
figure; imshow(x);

% Aplica TF nos filtros (Domínio da Frequência)
% Os filtros devem ser do mesmo tamanho da imagem
H1 = fft2(h1, r, c); % Filtro da Média (Dominio Frequência)
H2 = fft2(h2, r, c); % Filtro Gaussiano (Dominio Frequência)

% Representação gráfica dos filtros
figure; mesh(fftshift(real(H1)));
figure; mesh(fftshift(real(H2)));

% Exibe os filtros no domínio da frequência
figure; imshow(fftshift(abs(H1))*20);
figure; imshow(fftshift(abs(H2))*20);

% Adiciona ruído Gaussiano na imagem original
xng = imnoise(x, 'gaussian', 0, 0.1);
figure; imshow(xng);

% Aplica TF na imagem com ruído (Espectro de Fourier)
XNG = fft2(xng, r, c);
figure; imshow(fftshift(abs(XNG))*0.00001);

G1 = XNG .* H1; % Filtra a imagem com a máscara de Média
g1 = ifft2(G1); % Aplica Transformada Inversa de Fourier (Domínio Espacial)
g1 = g1(1:r, 2:c); % Exibe a imagem resultante
figure; imshow(g1*0.004);

G2 = XNG .* H2; % Filtra a imagem com a máscara gaussiana
g2 = ifft2(G2); % Aplica Transformada Inversa de Fourier (Domínio Espacial)
g2 = g2(1:r, 2:c); % Exibe a imagem resultante
figure; imshow(g2*0.004);
```

- Filtragem por Média:



- Filtragem Gaussiana:



Filtragem no Domínio da Frequência (Remoção de ruído Gaussiano)

Imagem Original

Imagem com ruído Gaussiano

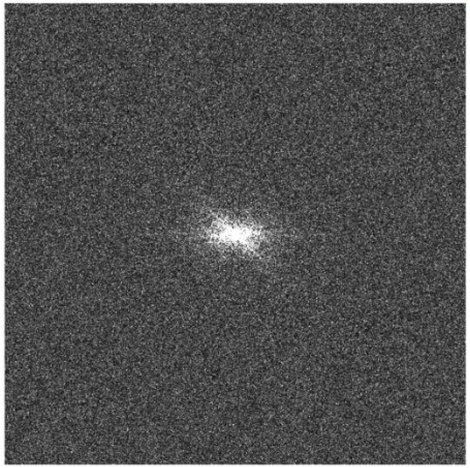


Imagem com ruído no Domínio da Frequência (espectro de Fourier)

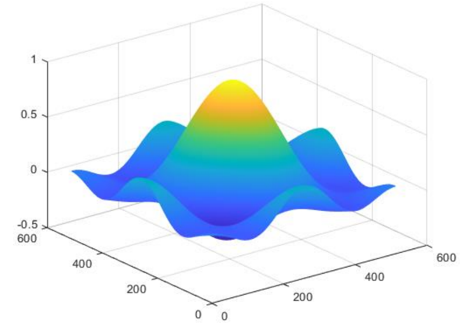
Imagem filtrada com máscara de Média

Imagem filtrada com máscara Gaussiana

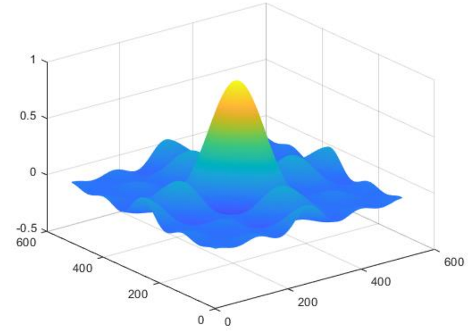
Filtragem no Domínio da Frequência (Remoção de ruído Sal e Pimenta)

```
h1 = ones(3)/9; % Filtro da Média (Dominio Espacial)
h2 = fspecial('gaussian', 5,5); % Filtro Gaussiano (Dominio Espacial)

x = imread('lenna.jpg'); % Lê uma imagem (Dominio Espacial)
x = rgb2gray(x); % Converte imagem em escala de cinza
[r,c] = size(x); % Salva dimensões da imagem
figure; imshow(x);

% Aplica TF nos filtros (Domínio da Frequência)
% Os filtros devem ser do mesmo tamanho da imagem
H1 = fft2(h1, r, c); % Filtro da Média (Dominio Frequência)
H2 = fft2(h2, r, c); % Filtro Gaussiano (Dominio Frequência)

% Representação gráfica dos filtros
figure; mesh(fftshift(real(H1)));
figure; mesh(fftshift(real(H2)));

% Exibe os filtros no domínio da frequência
figure; imshow(fftshift(abs(H1))*20);
figure; imshow(fftshift(abs(H2))*20);

% Adiciona 10% de ruído Salt & Pepper na imagem original
xns = imnoise(x,'salt & pepper', 0.1);
figure; imshow(xns);

% Aplica TF na imagem com ruído (Espectro de Fourier)
XNS = fft2(xns, r, c);
figure; imshow(fftshift(abs(XNS))*0.00001);

G3 = XNS .* H1; % Filtra a imagem com a máscara de Média
g3 = ifft2(G3); % Aplica Transformada Inversa de Fourier (Domínio Espacial)
g3 = g3(1:r, 2:c); % Exibe a imagem resultante
figure; imshow(g3*0.004);

G4 = XNS .* H2; % Filtra a imagem com a máscara gaussiana
g4 = ifft2(G4); % Aplica Transformada Inversa de Fourier (Domínio Espacial)
g4 = g4(1:r, 2:c); % Exibe a imagem resultante
figure; imshow(g4*0.004);
```

- Filtragem por Média:



- Filtragem Gaussiana:



Filtragem no Domínio da Frequência (Remoção de ruído Sal e Pimenta)

Imagem Original

Imagem com ruído Sal e Pimenta

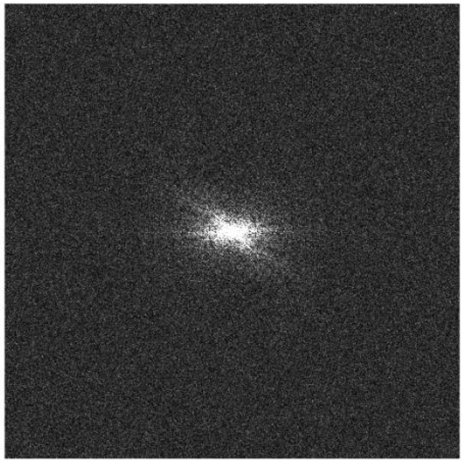


Imagem com ruído no Domínio da Frequência (espectro de Fourier)

Imagem filtrada com máscara de Média

Imagem filtrada com máscara Gaussiana

Filtragem usando a Transformada de Fourier

- Todos os filtros de frequência também podem ser implementados no domínio espacial e, se houver um *kernel* simples para o efeito de filtro desejado, é computacionalmente menos custoso executar a filtragem no domínio espacial.

- Filtragem no domínio de frequência é mais apropriada se não existir um *kernel* simples no domínio espacial e, em alguns casos, pode obter melhores resultados.

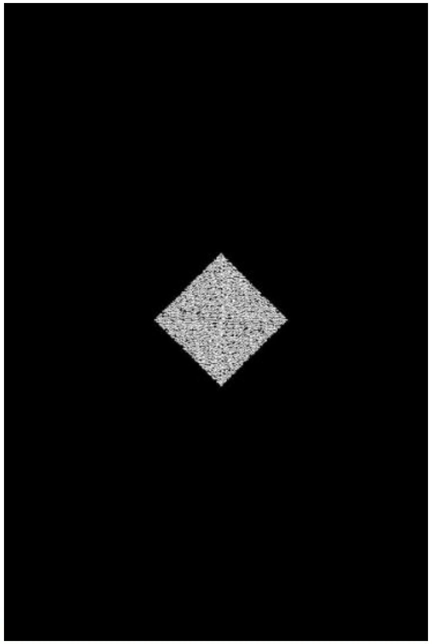
- Existem basicamente três tipos de filtros:
  - Passa-Baixa;
  - Passa-Alta;
  - Passa-Banda/Faixa.

Filtros Passa-Baixa

- Um **filtro Passa-Baixa** corresponde a atenuar as frequências mais altas, permitindo que as frequências mais baixas fiquem em evidência.

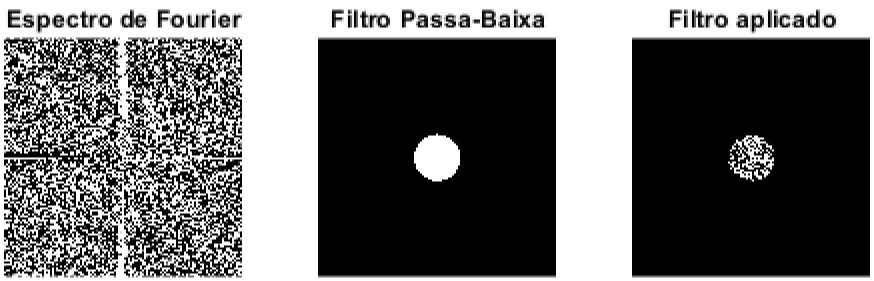
- No diagrama de módulo de uma imagem, as frequências altas se localizam nas bordas da imagem.

- O efeito equivalente no domínio espacial é o **desfoque ou suavização da imagem**.



Filtros Passa-Baixa

```
I = imread('Lenna.jpg'); % Lê uma imagem
I = rgb2gray(I); % Converte para escala de cinza
Id = im2double(I); % Converte para double

I_dft = fft2(Id); % Aplica a Transformada de Fourier
[M, N] = size(I); % Recupera o tamanho da imagem

% Retorna uma matriz bidimensional que deve ser do
% mesmo tamanho da imagem que está sendo processada.
% Os valores da matriz representam a distância de cada
% pixel com relação ao centro da imagem.
dist = distmatrix(M, N);

% Criação do filtro (máscara) Passa-Baixa ideal:
H = zeros(M, N); % Cria matriz de zeros
radius = 50; % Define o raio do círculo centralizado
% Define todos os pixels para cor branca a partir do
% centro da imagem, num raio definido na variável 'radius'
ind = dist <= radius;
H(ind) = 1;
Hd = double(H); % Converte o filtro para double

% Para aplicar o filtro, deve-se multiplicar seus pixels
% pelos equivalentes na imagem transformada (espectro)
DFT_filt = Hd .* I_dft;

% Após a aplicação do filtro, aplica a Transformada
% Inversa de Fourier para converter a imagem novamente
% para o domínio espacial
I2 = real(ifft2(DFT_filt));

subplot(1,3,1); imshow(fftshift(I_dft)); title('Espectro de Fourier');
subplot(1,3,2); imshow(fftshift(H)); title('Filtro Passa-Baixa');
subplot(1,3,3); imshow(fftshift(DFT_filt)); title('Filtro aplicado');
figure;
subplot(1,2,1); imshow(Id); title('Imagem Original');
subplot(1,2,2); imshow(I2); title('Imagem Filtrada');
```



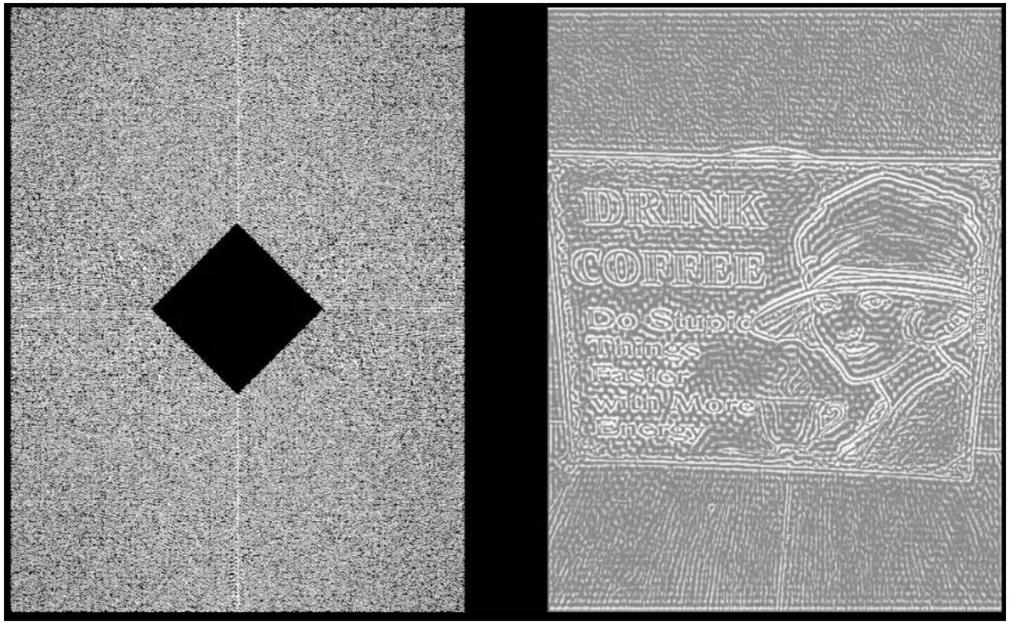
- Altere o valor da variável **radius** para outros resultados.

Filtros Passa-Alta

- Um **filtro Passa-Alta** corresponde a atenuar as frequências mais baixas, permitindo que as frequências mais altas fiquem em evidência.

- No diagrama de módulo de uma imagem, as frequências baixas se localizam no centro da imagem.

- O efeito equivalente no domínio espacial é o **realce das bordas da imagem**, porque as bordas contém as frequências altas.



Filtros Passa-Alta

```
I = imread('Lenna.jpg'); % Lê uma imagem
I = rgb2gray(I); % Converte para escala de cinza
Id = im2double(I); % Converte para double

I_dft = fft2(Id); % Aplica a Transformada de Fourier
[M, N] = size(I); % Recupera o tamanho da imagem

% Retorna uma matriz bidimensional que deve ser do
% mesmo tamanho da imagem que está sendo processada.
% Os valores da matriz representam a distância de cada
% pixel com relação ao centro da imagem.
dist = distmatrix(M, N);

% Criação do filtro (máscara) Passa-Alta ideal:
H = ones(M, N); % Cria matriz de preecnhidas com um (1)
radius = 30; % Define o raio do círculo centralizado
% Define todos os pixels para cor branca a partir do
% centro da imagem, num raio definido na variável 'radius'
ind = dist <= radius;
H(ind) = 0;

% Multiplica o filtro por uma constante (b), aplica um offset (a),
% e converte o filtro para double
a = 1; b = 1;
Hd = double(a + (b .* H));

% Para aplicar o filtro, deve-se multiplicar seus pixels
% pelos equivalentes na imagem transformada (espectro)
DFT_filt = Hd .* I_dft;

% Após a aplicação do filtro, aplica a Transformada
% Inversa de Fourier para converter a imagem novamente
% para o domínio espacial
I2 = real(ifft2(DFT_filt));

subplot(1,3,1); imshow(fftshift(I_dft)); title('Espectro de Fourier');
subplot(1,3,2); imshow(fftshift(H)); title('Filtro Passa-Alta');
subplot(1,3,3); imshow(fftshift(DFT_filt)); title('Filtro aplicado');
figure;
subplot(1,2,1); imshow(Id); title('Imagem Original');
subplot(1,2,2); imshow(I2); title('Imagem Filtrada');
```

- Altere os valores das variáveis **radius**, **a** e **b** para outros resultados.

Filtros Passa-Banda

- Um filtro **Passa-Banda** atenua frequências muito baixas e muito altas, mas mantém uma faixa média de frequências.

- A filtragem passa-banda pode ser usada para aprimorar as bordas (suprimir baixas frequências) enquanto reduz o ruído ao mesmo tempo (atenuar as altas frequências).

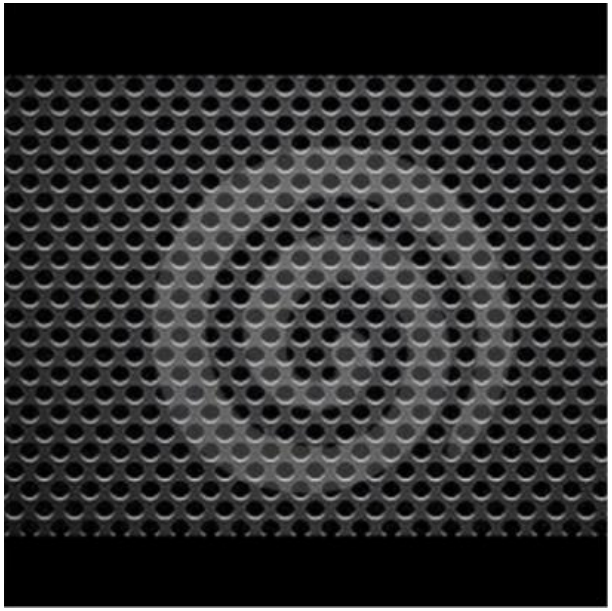
- Filtros passa-banda são uma combinação de ambos os filtros passa-baixa e passa-alta.

- Obtemos a função de filtro de um filtro passa-banda multiplicando as funções de filtro de um filtro passa-baixa e passa-alta no domínio da frequência, onde a frequência de corte do filtro passa-baixa é superior ao do filtro passa-alta.

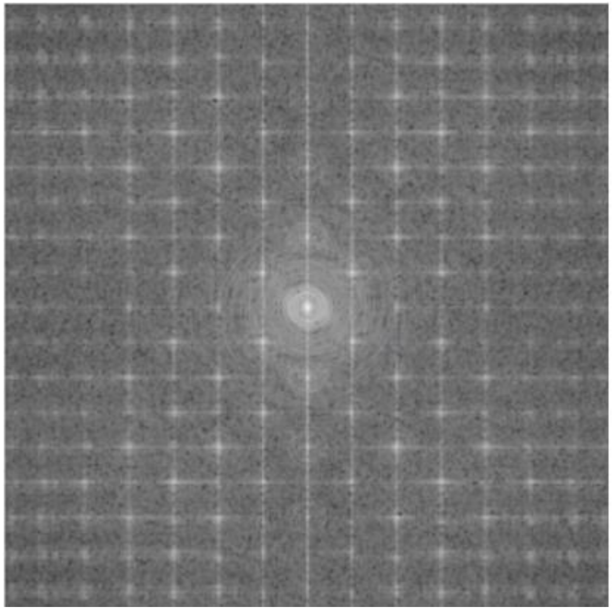
Filtragem no Domínio de Fourier

1) Imagem com ruído periódico (Domínio Espacial).

2) Espectro da imagem após a Transformada de Fourier (Domínio da Frequência).



1

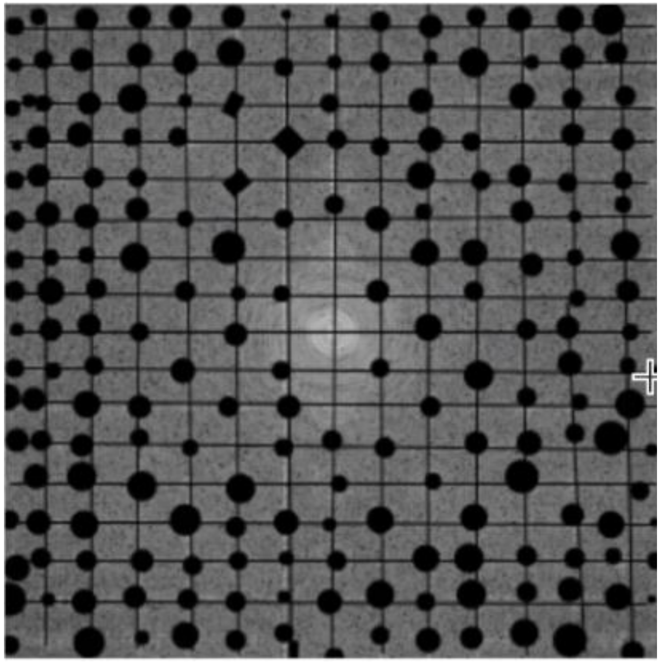


2

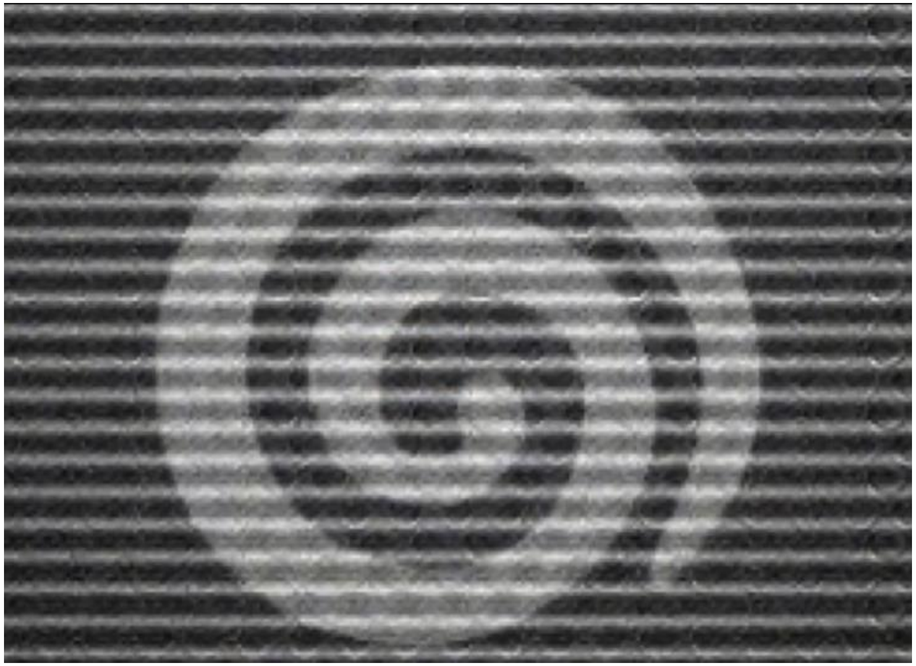
Filtragem no Domínio de Fourier

3) Remoção do ruído periódico.

4) Aplicação da Transformada Inversa de Fourier para recuperar a imagem do domínio da frequência para o domínio espacial.



3



4

Resumindo Filtragem no Domínio Espacial e de Frequência

- A filtragem é a operação de remoção de certos componentes de frequência de uma imagem.

- Os filtros de imagem podem funcionar diretamente no domínio espacial, bem como no domínio de frequência.

- No domínio espacial, o filtro pode ser visto como uma pequena matriz M × N, chamada *kernel* ou máscara, que é convoluída com a imagem de origem.

- Em uma operação de convolução, uma pequena vizinhança na imagem (proporcional ao *kernel*) é analisada e um novo valor de pixel é obtido como o pixel central correspondente da vizinhança.

Resumindo Filtragem no Domínio Espacial e de Frequência

- As imagens também podem ser processadas no domínio da frequência. Aqui uma máscara específica é necessária para remover ou permitir certas áreas do espectro da frequência.

- No domínio da frequência, tanto a imagem quanto a máscara devem possuir as mesmas dimensões.

- O ruído é geralmente o aspecto de maior frequência da imagem e pode ser removido pela máscara de filtro passa-baixa.

- No entanto, remover as frequências de ruído também remove com elas alguns componentes úteis de alta frequência, como bordas na imagem, produzindo assim uma imagem borrada.

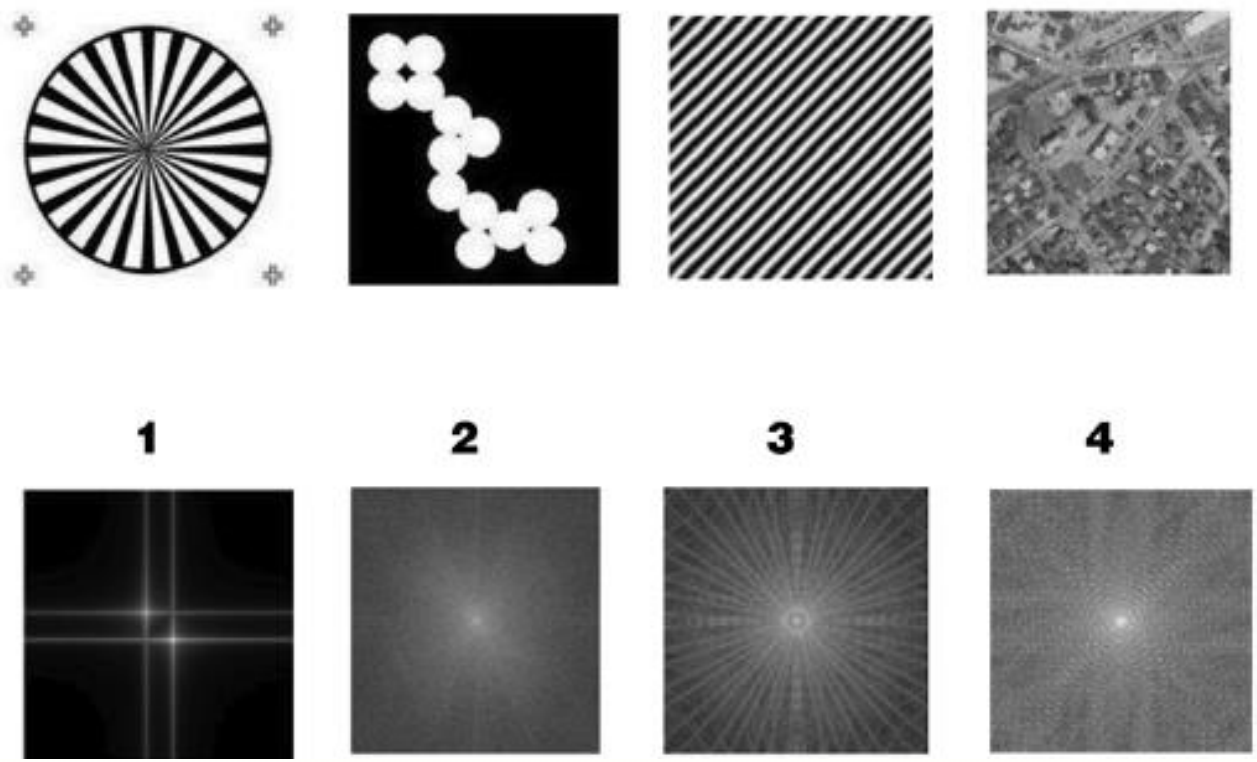
- A seleção cuidadosa da máscara e sua aplicação pode obter uma solução na qual a remoção de ruído e a conservação das bordas são otimizadas.

Exercício 1

Na figura adiante, são exibidas quatro imagens rotuladas com A, B, C e D. As imagens rotuladas com 1, 2, 3 e 4 mostram os espectros de Fourier das quatro anteriores imagens, mas em ordem aleatória.

Case cada imagem ao correspondente espectro de Fourier, justificando sua escolha.

Note que os espectros de Fourier são exibidos como $\log(1 + |F|)$, em que F é a transformada de Fourier da imagem.



Exercício 2

Três filtros de convolução linear discreta, A, B e C, são dados por

$$A = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad B = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & 1 & -1 \end{bmatrix} \quad C = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

A imagem I1 a seguir foi processada de três modos independentes, produzindo os resultados rotulados I2, I3 e I4, também mostrados adiante. Carregue a imagem 'trui.png' em Matlab e processe-a usando cada um dos três filtros, e case os resultados com as imagens processadas.

I1

I2

I3

I4