

PSI-3432 — Processamento de Áudio e Imagem

Lista de Exercícios 4 - Gabarito

Vítor H. Nascimento Thiago Yuji Aoyagi

4 de novembro de 2019

Processamento de Sinais Multidimensionais

1. Na TDF unidimensional, uma raia $X[k]$, juntamente com seu conjugado em $X[N - k]$, contém a informação de módulo e fase da componente senoidal de frequência $\omega = 2\pi k/N$ presente no sinal $x[n]$. Para sinais de duas dimensões, a interpretação é análoga. As raias $X[k_1, k_2]$ e $X[N_1 - k_1, N_2 - k_2]$, juntas, representam o módulo e fase da parte real do sinal

$$e^{j\frac{2\pi k_1}{N_1}n_1} e^{j\frac{2\pi k_2}{N_2}n_2} = e^{j\left(\frac{2\pi k_1}{N_1}n_1 + \frac{2\pi k_2}{N_2}n_2\right)} = e^{j\boldsymbol{\omega}^T \mathbf{n}}$$

onde

$$\boldsymbol{\omega} = \begin{pmatrix} \frac{2\pi k_1}{N_1} \\ \frac{2\pi k_2}{N_2} \end{pmatrix} \quad \text{e} \quad \mathbf{n} = \begin{pmatrix} n_1 \\ n_2 \end{pmatrix}$$

Qual é a direção da onda $\Re\{e^{j\boldsymbol{\omega}^T \mathbf{n}}\}$ (para uma raia espectral fixa)? Faça um esboço desse sinal.

(lembre, de álgebra linear, que todo vetor pode ser decomposto em $\mathbf{a} = \mathbf{a}_1 + \mathbf{a}_2$, onde \mathbf{a}_1 e \mathbf{a}_2 são ortogonais entre si, e que o produto interno $\mathbf{a}^T \mathbf{b}$ é zero quando \mathbf{a} e \mathbf{b} são ortogonais)

Resposta 1. Podemos escrever o vetor de frequências como $\boldsymbol{\omega} = |\omega| \mathbf{e}$, onde \mathbf{e} é um vetor unitário que contém a direção de $\boldsymbol{\omega}$. Decompomos então o vetor de índices espaciais em $\mathbf{n} = m_1 \mathbf{e} + m_2 \mathbf{f}$, com \mathbf{f} ortogonal a \mathbf{e} . Assim, podemos escrever o sinal 2D $e^{j\boldsymbol{\omega}^T \mathbf{n}}$ como:

$$\begin{aligned} e^{j\boldsymbol{\omega}^T \mathbf{n}} &= e^{j|\omega| \mathbf{e}^T (m_1 \mathbf{e} + m_2 \mathbf{f})} = e^{j|\omega| m_1 \mathbf{e}^T \mathbf{e}} e^{j|\omega| m_2 \mathbf{e}^T \mathbf{f}} \\ &= e^{j|\omega| m_1 \times 1} e^{j|\omega| m_2 \times 0} = e^{j|\omega| m_1} \end{aligned}$$

ou seja, o sinal só depende de m_1 , a componente espacial na direção \mathbf{e} . Portanto, o sinal $\Re\{e^{j\boldsymbol{\omega}^T \mathbf{n}}\}$ se “propaga” na direção de $\boldsymbol{\omega}$, com frequência igual ao seu módulo $|\omega|$. A Figura 1 ilustra isso para o caso em que $N_1 = N_2$.

2. Mostre que a convolução linear entre os sinais $x[n_1, n_2]$ (comprimento N_{x1}, N_{x2}) e $h[n_1, n_2]$ (comprimento N_{h1}, N_{h2}) pode ser implementada usando a TDF se esta última for calculada usando-se $N_1 \geq N_{x1} + N_{h1} - 1$, $N_2 \geq N_{x2} + N_{h2} - 1$ pontos.

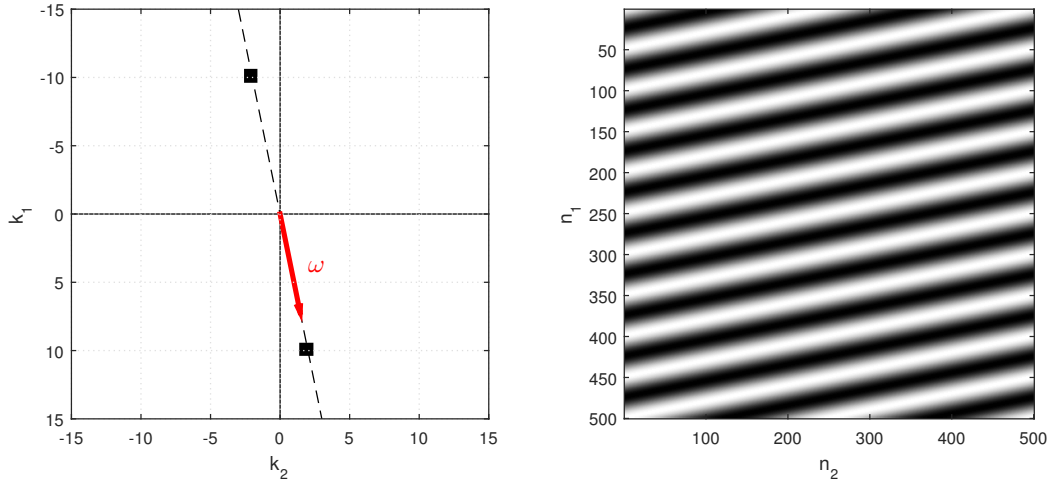


Figura 1: Esquerda: espectro $X[k_1, k_2]$ com duas raias não nulas; o vetor ω está com módulo fora de escala; Direita: imagem correspondente ao espectro.

Resposta 2.

O suporte da convolução linear entre $x[n_1, n_2]$ e $h[n_1, n_2]$ é o retângulo $R = \{(n_1, n_2) \in \mathbb{Z}^2, 0 \leq n_1 \leq N_{x_1} + N_{h_1} - 2, 0 \leq n_2 \leq N_{x_2} + N_{h_2} - 2\}$. Definimos as sequências periódicas

$$\tilde{x}[n_1, n_2] = \begin{cases} x[n_1, n_2], & 0 \leq n_1 \leq N_{x_1} - 1, 0 \leq n_2 \leq N_{x_2} - 1, \\ 0, & \text{caso contrário,} \end{cases}$$

$$\tilde{h}[n_1, n_2] = \begin{cases} h[n_1, n_2], & 0 \leq n_1 \leq N_{h_1} - 1, 0 \leq n_2 \leq N_{h_2} - 1, \\ 0, & \text{caso contrário,} \end{cases}$$

ambas com período N_1 e N_2 . A convolução periódica desses sinais é dado por:

$$\tilde{x}[n_1, n_2] \circledast \tilde{h}[n_1, n_2] = \sum_{k_1=0}^{N_1-1} \sum_{k_2=0}^{N_2-1} \tilde{x}[k_1, k_2] \tilde{h}[n_1 - k_1, n_2 - k_2]$$

e, se definirmos que $N_1 \geq N_{x_1} + N_{h_1} - 1$ e $N_2 \geq N_{x_2} + N_{h_2} - 1$, a expressão resultará no mesmo resultado da convolução linear. Logo, pela propriedade

$$\tilde{x}[n_1, n_2] \circledast \tilde{h}[n_1, n_2] \rightarrow X[\lambda_1, \lambda_2] \cdot H[\lambda_1, \lambda_2]$$

podemos usar a TDF para calcular a convolução linear entre $x[n_1, n_2]$ e $h[n_1, n_2]$.

3. Projete um filtro de duas dimensões, com fase nula e com janela de Kaiser, com as especificações:

- Banda-passante em $0 \leq |\omega_1| \leq \pi/3$ e $0 \leq |\omega_2| \leq \pi/4$.
- Banda de rejeição em $\pi/2 \leq |\omega_1| \leq \pi$ e $\pi/3 \leq |\omega_2| \leq \pi$.
- Oscilação máxima na banda passante e na banda de rejeição de 0,002.

- (a) Quais as frequências de corte do filtro?
 (b) Quais os parâmetros A , $\Delta\omega_1$ e $\Delta\omega_2$ utilizados no projeto da janela de Kaiser?
 (c) Quais são os comprimentos dos filtros e o parâmetro β ?

Resposta 3. (a) As frequências de corte serão

$$\omega_{c1} = \frac{\pi/3 + \pi/2}{2} = \frac{5\pi}{12} \text{ e } \omega_{c2} = \frac{\pi/4 + \pi/3}{2} = \frac{7\pi}{24}$$

- (b) Aplicando um filtro de cada dimensão por vez, teremos:

$$2\delta_p + \delta_p^2 \leq 0,002 \rightarrow \delta_p \approx 0,001, \quad \delta_r = \sqrt{0,002} = 0,0447$$

e, portanto, $A = -20 \log(\min\{\delta_p, \delta_r\}) = 60$. Teremos também, para as duas frequências ω_1 e ω_2 ,

$$\Delta\omega_1 = \pi/6, \quad \Delta\omega_2 = \pi/12.$$

- (c) Com os parâmetros dados, obtemos assim janelas de Kaiser com $\beta = 5.6533$ e

$$N_1 \approx \frac{A - 8}{2,285\Delta\omega_1} + 1 \approx 45 \text{ (experimentalmente 47)}$$

$$N_2 \approx \frac{A - 8}{2,285\Delta\omega_2} + 1 \approx 88$$

4. Sabendo que a FFT de duas dimensões requer $N_1 N_2 \log_2(N_1 N_2)$ operações se N_1 e N_2 forem potências de dois, e lembrando que a implementação direta da TDF requer $(N_1 N_2)^2$ operações, compare o ganho de número de operações para implementar um filtro de 101×101 coeficientes aplicado a uma imagem de 128×128 pixels.

Resposta 4. A convolução da imagem com o filtro resultará em $(101 + 128 - 1) \times (101 + 128 - 1) = 228^2$ elementos. Portanto, a implementação direta da TDF resultará em $228^4 = 2702336256$ operações.

Para o uso da FFT, completaremos com zeros o número de elementos do resultado até obter uma potência de 2, para utilizar a máxima eficiência do algoritmo; portanto, teremos 256^2 elementos. Desta forma, teremos $256^2 \cdot \log_2(256^2) = 1048576$ operações. Enfim, o ganho de operações será $2702336256/1048576 \approx 2577$.

5. Neste exercício, vamos comparar trabalhar com a reconstrução da imagem sintética ‘phantom.mat’ de tamanho 100×100 chamada Fantasma de Shepp-Logan. Esta imagem representa os coeficientes de atenuação de um corte transversal de um fantasma que modela a cabeça humana. Cada pixel da imagem é um coeficiente de atenuação cujo valor varia entre 0 e 57.27, o coeficiente de atenuação do osso humano em cm^{-1} . Vamos supor que a imagem tem tamanho $20cm \times 20cm$, de forma que a distância entre dois pixels adjacentes seja de $2mm$.

- (a) Vamos simular uma tomografia de raios paralelos com o fantasma usando uma máquina que mede 720 projeções de raios-X em ângulos linearmente espaçados, isto é, que mede uma projeção a cada 0.5° . Considerando que as dimensões da imagem são de $20\text{cm} \times 20\text{cm}$, use a função `radon` para calcular a atenuação total para cada projeção e depois plote o sinograma.

Não se esqueça de colocar os eixos e a escala de cor. No Matlab, `imagesc(theta,u,R)` mostra a matriz `R` com eixos `theta` e `u`. O comando `colorbar` adiciona a escala de cores.

- (b) Na prática, o número de fótons em cada projeção é limitado, gerando um ruído de medida do tipo Poisson. Se N é o número de fótons recebidos por um detector e N_0 é o número de fótons emitido pela fonte na direção do detector, então o valor esperado do número de fótons recebidos segue a lei de Beer-Lambert:

$$E\{N\} = N_0 e^{-\int \mu(x,y) d\ell} = N_0 e^{-R(u,\theta)}$$

onde μ é a imagem e R sua transformada Radon. Suponha que sejam emitidos $N_0 = 100$ fótons na direção de cada detector em cada projeção. Usando a função `poissrnd`, simule o número de fótons recebidos por cada detector em cada projeção e então plote o sinograma medido. Compare com o sinograma sem ruído.

Dica: Basta gerar um ruído de Poisson N com média $N_0 e^{-R(u,\theta)}$ e depois calcular o sinograma ruidoso como $R_{noisy}(u,\theta) = -\ln(N/N_0)$.

- (c) Reconstrua as imagens com ruído e sem ruído usando a transformada Radon inversa sem a aplicação do filtro. Para isso, use a função `iradon` com o par de argumentos `'Filter'`, `'None'`. Depois, reconstrua usando o filtro de Ram-Lak. Compare as quatro imagens que você obteve.