Universidade Tecnológica Federal do Paraná - UTFPR
Departamento Acadêmico de Eletrônica – DAELN
Prof. Gustavo B. Borba

2013/2
IF69D

ATIVIDADE 12
26.fev.2014

ATENÇÃO
- Vale nota.
- Em dupla ou individual.
- Apresentar quando todos estiverem funcionando ou até 22h30.

12a) Equação da DFT

A transformada de Fourier é a operação que recebe como entrada um sinal no domínio do tempo (ou do espaço, no caso de uma imagem) e apresenta na saída o sinal no domínio da frequência. Na maioria dos textos, a transformada discreta de Fourier (DFT) de um sinal 1D é definida pela equação a seguir [[BB] Eq. 13.45].

$$G(m) = \frac{1}{\sqrt{M}} \sum_{u=0}^{M-1} g(u) \cdot e^{-i2\pi \frac{mu}{M}}$$

To-do: inversa

$g(u)$ é o sinal original, de comprimento M ($u = 0 \ldots M\text{-}1$) e $G(m)$ é o espectro (domínio da frequência), também de comprimento M. Ambos são vetores de números complexos, isto é, cada valor de $g(u)$ e de $G(m)$ possui uma parte real e uma parte imaginária [[BB] Eq. 13.47]:

$$g(u) = g_{Re}(u) + i \cdot g_{Im}(u)$$

$$G(m) = G_{Re}(m) + i \cdot G_{Im}(m)$$

Uma das conveniências desta equação é que ela é concisa. Mas se fosse pra implementar a DFT computacionalmente, teríamos que expandir esta equação usando a fórmula de Euler para números complexos, mostrada a seguir.

$$e^{-ix} = cos(x) - i \cdot sin(x)$$

Após a substituição e mais uma manipuladinha, obtém-se as equações a seguir [[BB] Eq 13.49 e 13.50]. No entanto, é importante lembrar que, computacionalmente, não se realiza a implementação destas equações para obter a DFT. O método utilizado para a obtenção da DFT é o da FFT (Fast Fourier Transform).

$$G_{Re}(m) = \frac{1}{\sqrt{M}} \sum_{u=0}^{M-1} g_{Re}(u) \cdot C_m^M(u) + g_{Im}(u) \cdot S_m^M(u)$$

$$G_{Im}(m) = \frac{1}{\sqrt{M}} \sum_{u=0}^{M-1} g_{Im}(u) \cdot C_m^M(u) - g_{Re}(u) \cdot S_m^M(u)$$

$$C_m^M(u) = cos\left(2\pi \frac{mu}{M}\right), \quad S_m^M(u) = sin\left(2\pi \frac{mu}{M}\right)$$

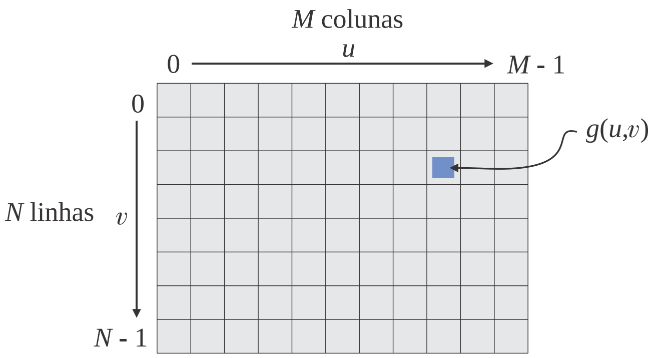
To-do: inversa

12b) Algumas propriedades importantes para a aplicação da DFT em imagens

1. Equação da DFT 2D

A definição da DFT para um sinal de duas dimensões, com é o caso de imagens, é mostrada a seguir [[BB] Eq 14.1]. A figura que mostra a convenção para o sistema de coordenadas foi redesenhada de [[BB] Figura 2.6].

$$G(m,n) = \frac{1}{\sqrt{MN}} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} g(u,v) \cdot e^{-i2\pi \frac{mu}{M}} \cdot e^{-i2\pi \frac{nv}{M}}$$



2. Separabilidade

A equação anterior, da definição da DFT em 2D pode ser reescrita conforme mostrado a seguir [[BB] Eq. 14.7].

$$G(m,n) = \frac{1}{\sqrt{N}} \sum_{v=0}^{N-1} \left[\frac{1}{\sqrt{M}} \sum_{u=0}^{M-1} g(u,v) \cdot e^{-i2\pi \frac{mu}{M}} \right] \cdot e^{-i2\pi \frac{nv}{M}}$$

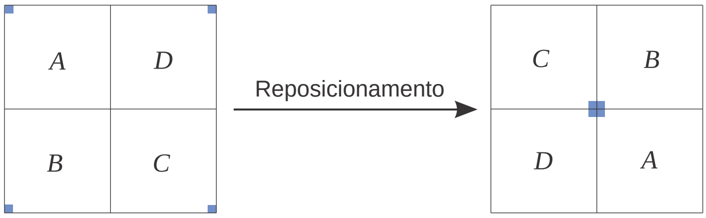
A parte dentro dos colchetes é a DFT 1D da linha v da imagem, pois v é fixo e o somatório é feito para todos os valores de u (todas as colunas). Matematicamente, pode-se utilizar a notação $g(\cdot,v)$ (lê-se: todas as colunas da linha v). Isto significa que, para calcular a DFT 2D, pode-se fazer primeiramente a DFT 1D de cada linha da imagem, e em seguida aplicar uma nova DFT 1D em cada coluna da DFT anterior, conforme expresso pelas equações a seguir [[BB] Tópico 14.1.2]. $g''(u,\cdot)$ é a própria DFT 2D, $G(m,n)$.

$$g'(\cdot,v) \leftarrow DFT\{g(\cdot,v)\} \quad para \;\; 0 \le v < N$$

$$g''(u,\cdot) \leftarrow DFT\{g'(u,\cdot)\} \quad para \;\; 0 \le u < M$$

3. Visualização

Na transformada de Fourier, o componente correspondente à frequência zero é denominado coeficiente DC. Na saída da DFT 2D, os componentes DC estão localizados nos cantos da matriz e, consequentemente, os componentes de frequência mais alta no centro. No entanto, para a visualização e também para facilitar a aplicação de filtros, convenciona-se posicionar o componente DC no centro da matriz de saída. Isso pode ser conseguido multiplicando-se os pixels por $(-1)^{u+v}$ antes de submeter a imagem à transformada. A outra maneira é reposicionar cada quadrante da saída da transformada, conforme mostrado na figura a seguir [[BB] Figura 14.3]. No MATLAB, este repocisionamento pode ser obtido utilizando-se a função `fftshift`. Com isso, na DFT as componentes de frequência mais baixa estão posicionadas na região central, e as de frequência mais alta, nas bordas.



Cada elemento da saída da DFT é um número complexo. Para visualizar esses valores, deve-se calcular as suas *magnitudes* (módulo de um número complexo). A magnitude do espectro, também chamada de *espectro de potência*, é obtida a partir da equação a seguir. No MATLAB, a função que calcula a magnitude do espetro é a `abs`.

$$|G(m)| = \sqrt{G_{Re}^2(m) + G_{Im}^2(m)}$$

Ainda, é necessário levar em consideração que os valores numéricos na saída da DFT apresentam diferenças de amplitude muito grandes, especialmente do componente DC em relação aos demais (o componente DC é a soma de todos os pixels da imagem). Com isso, ao tentar visualizar diretamente a saída a DFT, o que pode ser feito reescalonando os valores para a faixa de 0 até 1, por exemplo (função `mat2gray`), o resultado é um único ponto de intensidade alta no centro (componente DC). Assim, costuma-se utilizar o logaritmo para mostrar os valores. No MATLAB, basta fazer `mat2gray(log(1+dftmag))`, onde `dftmag` é a magnitude do espectro.

```
clear all, close all

g = imread('cameraman.tif');
gd = double(g);

% DFT 2D
dft = fft2(gd);
% Reposicionamento (shifiting)
dfts = fftshift(dft);
% Magnitude
dftsm = abs(dfts);
% Visualização
dftsmv = mat2gray(log(1+dftsm));

%Display
figure
imshow(g)
title('Imagem de entrada')
figure
imshow(dftsmv)
title('DFT 2D')
```

To-do: FT de um grating

12.1) Propriedade da separabilidade

Implemente a DFT 2D usando apenas DFTs 1D. Você vai usar apenas a função `fft`, e não `fft2`. Segundo o teorema da separabilidade, faça a DFT das linhas da imagem (`dftr = fft(img,[],1)`) e em seguida a DFT das colunas (`fft(dftr,[],2)`) . O resultado deve ser similar ao obtido a partir da função `fft2`.

4. Translação e rotação

Esta propridade afirma que o espectro não é sensível à translação da imagem, mas rotaciona no mesmo ângulo que a imagem [[GW] Tópico 4.6.5].

12.2) Invariabilidade do espectro à translação e resposta à rotação

Faça um script para testar esta propriedade. Crie três imagens sintéticas contendo um fundo preto (ausência de sinal) e um quadrado branco (imagem 'propriamente dita') e obtenha o espectro de cada uma, para comparar. Na primeira imagem o quadrado deve estar no centro. Na segunda, deslocado do centro (translação). Na terceira, rotacionado em relação à primeira.

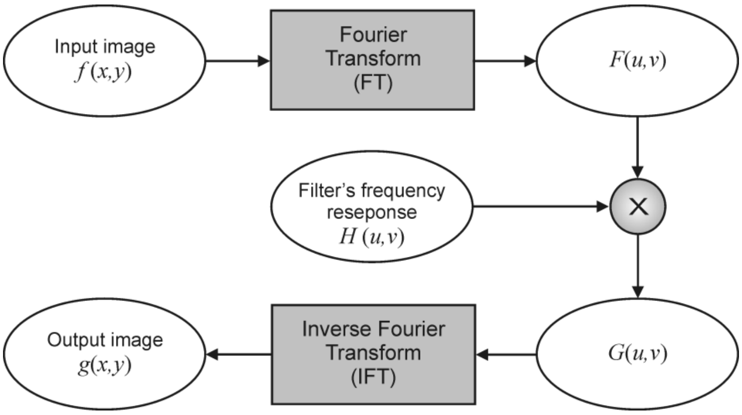
5. Teorema da convolução

Afirma que uma convolução no domínio do espaço (imagem original) é equivalente a uma multiplicação ponto-a-ponto no domínio da frequência (DFT). Com isso, é possível implementar, por exemplo, filtros passa-baixas e passa-altas no domínio da frequência.

$$g_f(u,v) = g(u,v) * h(u,v) = DFT^{-1}\{G(m,n) \cdot H(m,n)\}$$

12c) Filtragem no domínio da frequência

O diagrama a seguir representa o teorema da convolução [[OM] Figura 11.1].



Observe que quem multiplica a imagem ponto-a-ponto é transformada de Fourier da função de transferência do filtro. Porém, na prática, os filtros são projetados sem levar em consideração este detalhe, isto é, são elaborados "diretamente no domínio da frequência". Lembre que os componentes de baixa frequência então no centro da matriz da transformada e, as componentes de alta frequência, nas bordas.

No exemplo MATLAB a seguir são mostradas as operações de passa-altas e passa-baixas no domínio da frequência usando filtros ideais (o perfil é um degrau). Pode-se observar o efeito indesejado de *ringing*. Isto ocorre devido ao corte abrupto das componentes de frequência na transformada. Em sinais e sistemas e processamento digital de sinais isso é conhecido como *efeito Gibbs* [[SS] Capítulo 11].

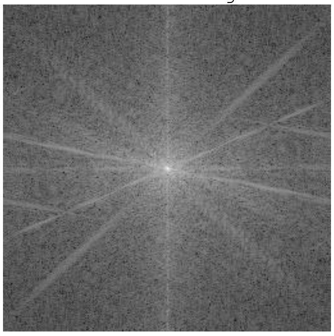
12.3) Passa-altas e passa-baixas Gaussiano no domínio da frequência

Faça as operações passa-altas e passa-baixas no domínio da frequência usando filtros Gaussianos. Teste com diferentes valores de σ. Você pode gerar o filtro usando a função `meshgrid` e a equação do Gaussiano 2D, como fizemos em aulas passadas. Outra possibilidade, mais fácil :-), é usando a função `fspecial`. Só lembre a `fspecial` gera kernels cuja soma dos coeficiente é sempre igual a 1, para não alterar o valor médio da imagem. Para usá-lo como um filtro no domínio da frequência, reescalone-o para a faixa de 0 até 1 (função `mat2gray`).

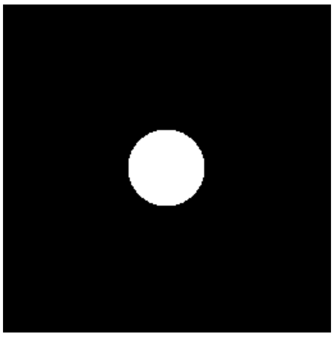
```
clear all, close all

f = imread('cameraman.tif');
fd = double(f);

% DFT 2D
F = fft2(fd);
% Reposicionamento (shifiting)
F = fftshift(F);

% Filtro passa-baixas ideal
[nr nc] = size(f);
rc = round(nr/2);
cc = round(nc/2);
bw = false(nr,nc);
bw(rc,cc) = 1;
d = bwdist(bw);
raio = 30;
Hpb = double(d < raio);
% Filtro passa-altas ideal
Hpa = 1 - Hpb;

% Multiplicação ponto-a-ponto
Gpb = F .* Hpb;
Gpa = F .* Hpa;

% DFT 2D inversa
gpb = ifft2(Gpb);
gpa = ifft2(Gpa);
% Magnitude e visualização
gpb = mat2gray(abs(gpb));
gpa = mat2gray(abs(gpa));

% Display
figure, imshow(f)
title('Imagem de entrada')
figure,
imshow(mat2gray(log(1+abs(F))))
title('DFT da imagem')
figure, imshow(mat2gray(Hpb))
title('Filtro PB ideal')
figure, imshow(mat2gray(Hpa))
title('Filtro PA ideal')
figure, imshow(gpb)
title('Saída PB')
figure, imshow(gpa)
title('Saída PA')
```

Imagem de entrada

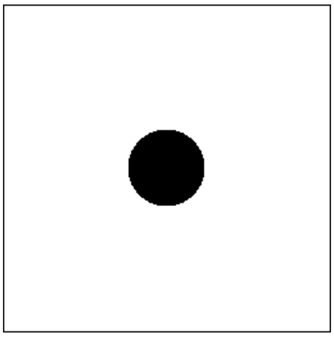
DFT da imagem



Filtro PB ideal



Filtro PA ideal



Saída PB

Saída PA

To-do: usar no diagrama a mesma notação da equação do teorema da convolução, deconvolution, Wiener, homomorphic, descriptors, exemplo Russ fig. 6.31, removing periodic noise.

Referências

[OM] Oge Marques, Practical image and video processing using MATLAB, Wiley, 2011.

[GW] Rafael C. Gonzalez, Richard E. Woods, Steven L. Eddins, Digital image processing, Pearson Prentice Hall, 3rd ed, 2008.

[BB] Wilhelm Burger, Mark Burge, Digital image processing – an algorithmic introduction using Java, Springer, 2010.

[SS] Steven W. Smith, The Scientist and Engineer's Guide to Digital Signal Processing, California Technical Pub, 1997. Disponível em http://www.dspguide.com/ch11/4.htm