Elaborazione di Segnali Multimediali
a.a. 2017/2018

Rappresentazione nel dominio della frequenza
Soluzioni

1 La trasformata discreta 1D

Calcoliamo lo spettro di $x(n) = \mathcal{R}_L(n)$ al variare di $L = 2, 5, 10, 20, 30$ attraverso interpolazione lineare dei campioni della DFT:

```
L = [2 5 10 20 30];
N = 500;
for i=1:length(L),
   x = [zeros(1,4) ones(1,L(i)) zeros(1,4)];
   X=fft(x,N);
   v = [-1/2:(1/N):1/2-1/N];
   X = fftshift(X);
   plot(v,abs(X));
   title(sprintf('Lunghezza della finestra pari a %d',L(i)));
   pause;
end;
```

Potete facilmente notare come all'aumentare della lunghezza della finestra rettangolare il lobo principale si stringe, coerentemente col fatto che più il segnale varia lentamente più lo spettro di ampiezza risulta essere concentrato intorno all'origine, ovvero le componenti frequenziali rilevanti che compongono il segnale sono quelle relative alle basse frequenze. Viceversa se un segnale varia velocemente ci sarà un contributo non trascurabile alle alte frequenze. Ripetiamo adesso l'esperimento con il segnale $x(n) = (-1)^n \mathcal{R}_L(n)$ fissando $L = 10$ e confrontando gli spettri:

```
L = 10;
N = 500;
x = [zeros(1,4) ones(1,L) zeros(1,4)];
n = [0:length(x)-1];
X = fftshift(fft(x,N));
y = x.*((-1).^n);
Y = fftshift(fft(y,N));
v = [-1/2:(1/N):1/2-1/N];
subplot(211); plot(v,abs(X)); title('impulso rettangolare');
subplot(212); plot(v,abs(Y)); title('impulso rettangolare alternato');
```

Confrontando i due spettri di ampiezza si può concludere che risultano identici eccetto che per una traslazione di $1/2$, in particolare lo spettro dell'impulso rettangolare è concentrato intorno all'origine (comportamento passabasso) mentre quello dell'impulso rettangolare alternato intorno a $\pm 1/2$ (comportamento passaalto). Ciò si spiega notando che risulta:

$$y(n) = (-1)^n \, x(n) = e^{j\pi n} \, x(n)$$

Dal momento che ad un prodotto nel tempo per $e^{j2\pi\nu_0 n}$ corrisponde la traslazione dello spettro proprio in ν_0, si può concludere che cambiare il segno dei coefficienti di posto dispari di un segnale è un'operazione che permette di traslare lo spettro di $1/2$ e quindi per esempio può essere usata per trasformare un filtro passabasso in un passaalto.

2 La trasformata discreta 2D

1. *Importanza dello spettro di fase*. Di seguito si presenta il codice per visualizzare modulo e fase di un'immagine usando le funzioni matlab `fft2` e `ifft2`:

```
close all; clear all; clc;
x = double(imread('volto.tif'));
X = fft2(x);
Xf = fftshift(X);
figure(1);
Z = log(1+abs(Xf));       % enhancement per visualizzazione
subplot(121); imshow(Z,[]); title('Spettro di ampiezza');
subplot(122); imshow(angle(Xf),[]); title('Spettro di fase');
```

Proviamo adesso a ricostruire un'immagine con la sola informazione di ampiezza o con la sola informazione di fase:

```
ym = real(ifft2(abs(X)));           % ricostruzione solo modulo
yf = real(ifft2(exp(i*angle(X))));  % ricostruzione solo fase
figure(2);
z = log(ym-min(ym(:))+1);     % enhancement per la visualizzazione
subplot(121); imshow(z,[]);
title('Ricostruzione spettro di ampiezza');
subplot(122); imshow(yf,[]);
title('Ricostruzione spettro di fase');
```

Infine, proviamo a scambiare spettri di ampiezza e fase di due immagini:

```
close all; clear all; clc;
x1 = double(imread('volto.tif'));
x2 = double(imread('rettangolo.jpg'));
X1 = fft2(x1);
X2 = fft2(x2);
```

```
y = real(ifft2(abs(X2).*exp(i*angle(X1))));
figure(1); imshow(y,[]); title('Modulo rettangolo, Fase volto')
y = real(ifft2(abs(X1).*exp(i*angle(X2))));
figure(2); imshow(y,[]); title('Modulo volto, Fase rettangolo')
```

2. *Risposta in frequenza dei filtri*. Visualizziamo la risposta in frequenza del filtro media aritmetica per $k = 5, 10, 15$.

```
P=500; Q=500;
dv1=1/P; dv2=1/Q;
n=-1/2:dv2:1/2-dv2;
m=-1/2:dv1:1/2-dv1;
[k,l] = meshgrid(n,m);

for i=5:5:15,
   h = fspecial('average',[i i]);
   H=fft2(h,P,Q);
   H=abs(fftshift(H));
   figure(1); mesh(k,l,H);
   figure(2); freqz2(h,500,500);    % verifica
   pause; close all;
end
```

La trasformata discreta di Fourier può anche essere implementata facilmente tramite prodotto matriciale, grazie al fatto che è una trasformata separabile, nel seguente modo:

```
function X = dft2D(x);

x = double(x);
[M N] = size(x);
n = [0:N-1]; k = [0:N-1];
WN = exp(-j*2*pi*k'*n/N);
m = [0:M-1]; l = [0:M-1];
WM = exp(-j*2*pi*l'*m/M);
X = WM*x*WN;
X = fftshift(X);
figure; imshow(log(1+abs(X)),[]); title('Spettro di ampiezza');
```

Questa funzione fornisce lo stesso risultato di `fft2`, solo che presenta tempi di esecuzione molto più rapidi, grazie all'implementazione mediante algoritmi veloci (Fast Fourier Transform). Confrontate i tempi con i comandi `tic` e `toc`.