

## ***Esercitazione su filtro di Sobel per l'elaborazione delle immagini***

Il filtro di Sobel opera sulle immagini come un gradiente lungo una direzione.

In particolare detta  $f(x, y)$  l'intensità dell'immagine nei punti  $x$  e  $y$ , il gradiente è dato da

$$\nabla f = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} \quad \text{Equazione 1}$$

Tramite il filtro di Sobel è possibile realizzare quindi le operazioni  $\frac{\partial f}{\partial x}$  e  $\frac{\partial f}{\partial y}$ .

Il modulo del gradiente si trova come

$$|\nabla f| = \sqrt{G_x^2 + G_y^2} \quad \text{Equazione 2}$$

e da informazioni sulle regione a massima velocità di variazione nei livelli dell'immagine, in ogni direzione. Questo ultimo operatore è molto utile nel caso di applicazioni quali la rilevazione di bordi.

Il filtro di Sobel che opera un gradiente lungo  $y$  è descritto dalla matrice bidimensionale (maschera) 3x3 seguente

$$\text{sob\_y} = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

in Matlab questa si ottiene come

```
sob_y=fspecial('sobel');
```

La trasformata discreta di Fourier di questa maschera si ottiene come

```
sob_y_f=fft2(sob_y);
```

in questo caso la TDF ha dimensioni 3x3.

N.B. Si fa notare che, come nel caso di segnali monodimensionali, la TDF è definita per immagini periodiche e andrebbe normalizzata per il numero di campioni dell'immagine. D'altronde, come nel caso monodimensionale, è possibile utilizzare la TDF per l'analisi spettrale delle immagini: in questo caso allora l'uscita della `fft2`, non normalizzata, fornisce una stima in alcuni punti frequenziali ( $k_x / (N_x d_x)$  e  $k_y / (N_y d_y)$ ), della trasformata di Fourier della *sequenza bidimensionale discreta*, ovvero dell'immagine. Per questo motivo nel seguito parleremo di trasformata di Fourier dell'immagine (omettendo "discreta").

Se poniamo il campionamento spaziale dell'immagine pari a  $dx=1$  e  $dy=1$ , la risoluzione in frequenza si ottiene come

```
dfx=1/(Nx*dx);
```

$dfy=1/(Ny*dy);$

da notare che le risoluzioni, sia spaziali che in frequenza, possono avere valori differenti.

In questo caso la taratura dell'asse frequenziale si ottiene come

$fx=dfx*[-Nx/2:Nx/2-1]; fy=dfy*[-Ny/2:Ny/2-1];$  per  $Nx$  e  $Ny$  pari

o come

$fx=dfx*[-(Nx-1)/2:(Nx-1)/2]; fy=dfy*[-(Ny-1)/2:(Ny-1)/2];$  per  $Nx$  e  $Ny$  dispari.

In questo caso per  $Nx=Ny=3$  si ottiene

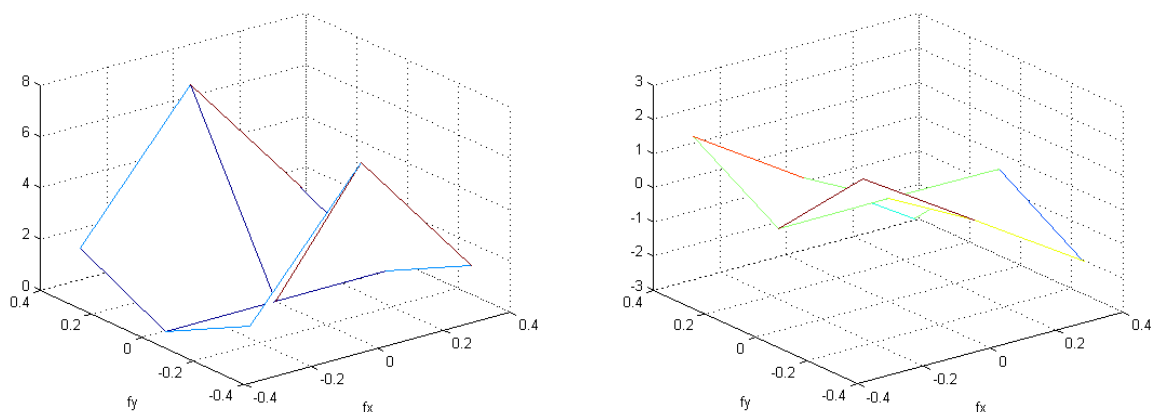
```
fx =[-0.3333    0    0.3333];  
fy =[-0.3333    0    0.3333];
```

I grafici del modulo e della fase della trasformata di Fourier della maschera si ottengono in questo modo

```
sob_y_shift=fftshift(sob_y_f);
```

```
figure  
mesh(fx,fy,abs(sob_y_shift));  
xlabel('fx');  
ylabel('fy');
```

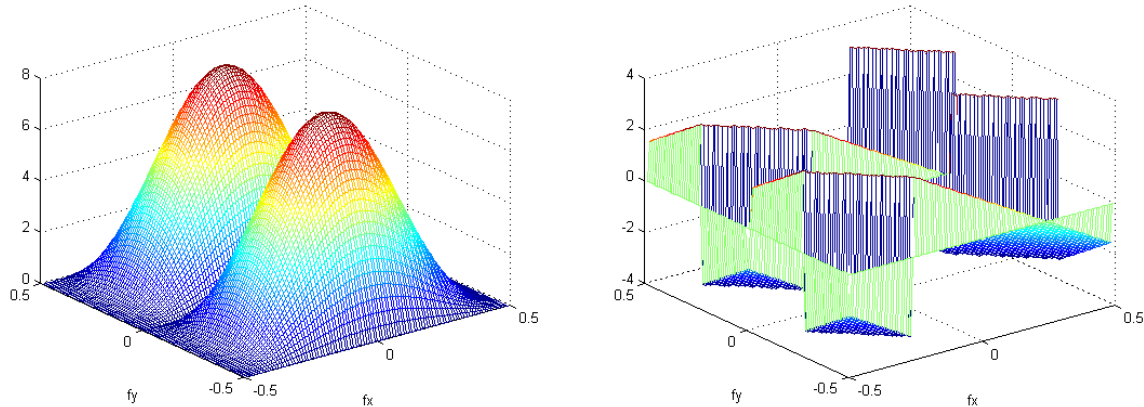
```
figure  
mesh(fx,fy,angle(sob_y_shift));  
xlabel('fx');  
ylabel('fy');
```



**Figura 1.** Sinistra: modulo della trasformata di Fourier del filtro di Sobel lungo y. Destra: fase della trasformata di Fourier del filtro di Sobel lungo y.

Se vogliamo migliorare la risoluzione frequenziale della trasformata possiamo operare l'operazione di zero padding. Scegliamo di effettuare uno zero padding in modo da portare  $N_x = N_y = 100$

Modificando di conseguenza la taratura degli assi frequenziali (dfx, dfy, fx e fy) si ottengono le seguenti figure.

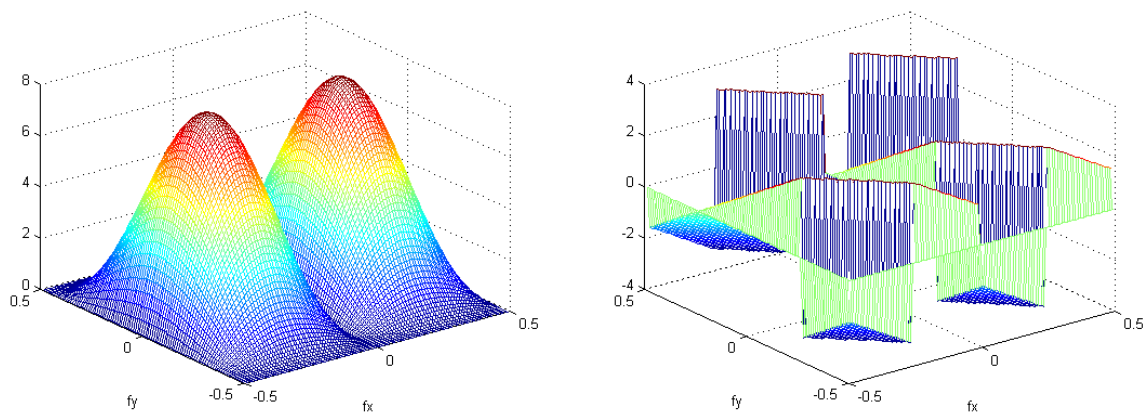


**Figura 2.** Sinistra: modulo della trasformata di Fourier del filtro di Sobel lungo y, con zero padding  $N_x=N_y=100$ . Destra: fase della trasformata di Fourier del filtro di Sobel lungo y, con zero padding  $N_x=N_y=100$ .

Il filtro di Sobel che opera un gradiente lungo x è descritto dalla matrice bidimensionale (maschera) 3x3 seguente, trasposta di sob\_y

$$\text{sob\_x} = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$$

Operando come nel caso precedente si ottiene la trasformata della maschera



**Figura 3.** Sinistra: modulo della trasformata di Fourier del filtro di Sobel lungo x, con zero padding  $N_x=N_y=100$ . Destra: fase della trasformata di Fourier del filtro di Sobel lungo x, con zero padding  $N_x=N_y=100$ .

L'immagine filtrata con un filtro bidimensionale può essere ottenuta tramite l'operazione di convoluzione bidimensionale

$$im\_filt = \sum_{k_1=-\infty}^{+\infty} \sum_{k_2=-\infty}^{+\infty} h(k_1, k_2) im(x - k_1, y - k_2)$$

In Matlab questa operazione può essere realizzata con il comando `conv2(.)`.

Quindi

```
im_sob_x=conv2(sob_x,im)
```

esegue la convoluzione bidimensionale tra le due matrici `sob_x` e `im`. Se `im` è una immagine, la matrice `im_sob_x` è l'immagine ottenuta dopo l'applicazione del gradiente lungo x ovvero

$$im\_sob\_x = \frac{\partial im}{\partial x}.$$

Consideriamo ad esempio un immagine che rappresenti un quadrato.

```
im=zeros(100);
im(26:74,26:74)=1;
```

```
imagesc(im);
colormap gray;
```

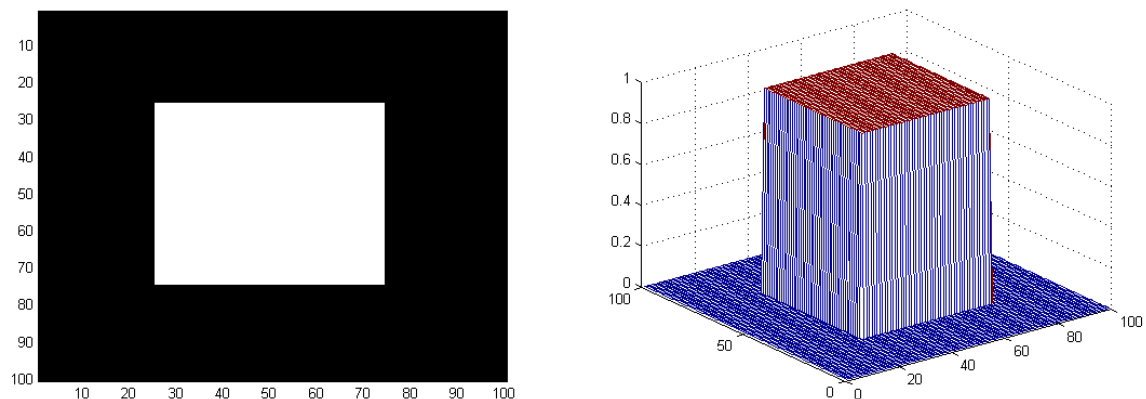
Il comando `imagesc(.)` riporta la matrice `im` come immagine, utilizzando una codifica dei colori descritta dalla variabile `colormap`. variabile è una matrice di dimensione `N_colx3`, e permette di codificare `N_col` diversi. Ogni riga contiene la descrizione in codifica RGB (Red Green Blue) di un colore.

In particolare il comando `imagesc (.)`, a differenza del comando `image(.)`, aggiusta l'intervallo di valori dell'immagine in modo da utilizzare tutta la `colormap`: questo significa che il valore più piccolo dell'intensità dell'immagine è descritto dalla prima riga della mappa di colori, mentre il valore più grande dall'ultima riga della mappa.

Ad un dato valore di un pixel dell'immagine è associata una riga della `colormap` e quindi un colore. Questo tipo di immagine si chiama immagine di intensità.

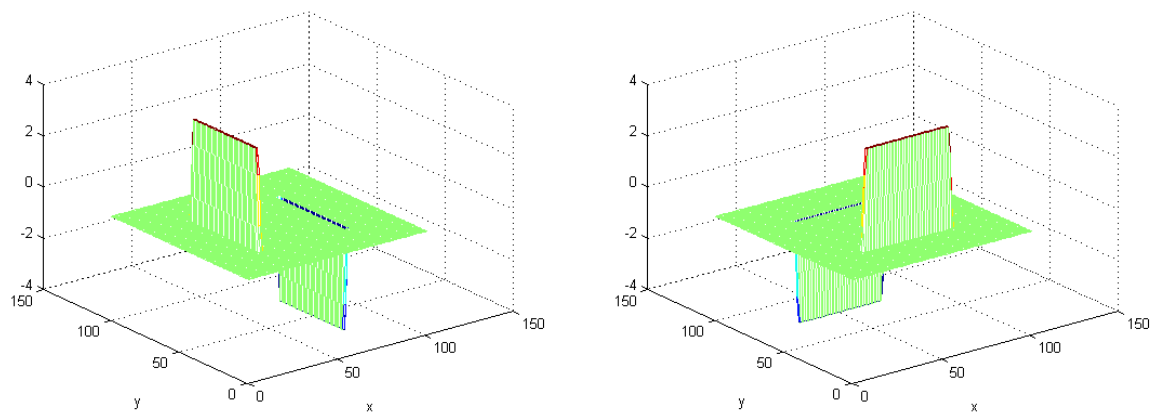
L'utente può definire una propria `colormap` o sceglierla (come nell'esempio la `colormap grey`) da una lista di `colormap` predefinite (vedi *doc colormap*).

Ecco l'immagine in esame, come si ottiene da comando `imagesc(.)` e `mesh(.)`.



**Figura 4.** Sinistra: immagine del quadrato utilizzando la colormap grey e imagesc(.). Destra: grafico del quadrato utilizzando mesh(.) e la colormap jet.

Ecco le immagini (visualizzate col comando mesh(.)) ottenute dalla convoluzione tra il quadrato e i due filtri di Sobel.

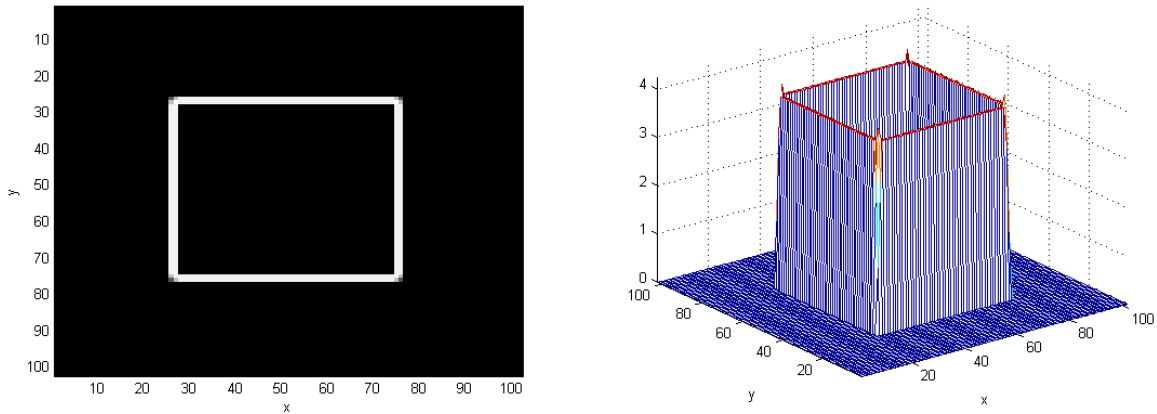


**Figura 5.** Sinistra: immagine del gradiente del quadrato rispetto a x. Destra: immagine del gradiente del quadrato rispetto a y.

Per ottenere il modulo del gradiente, in tutte le direzioni come da equazione 2, si utilizzano le due immagini precedenti in questo modo

```
im_mod_grad=sqrt(im_sob_x.^2+im_sob_y.^2);
```

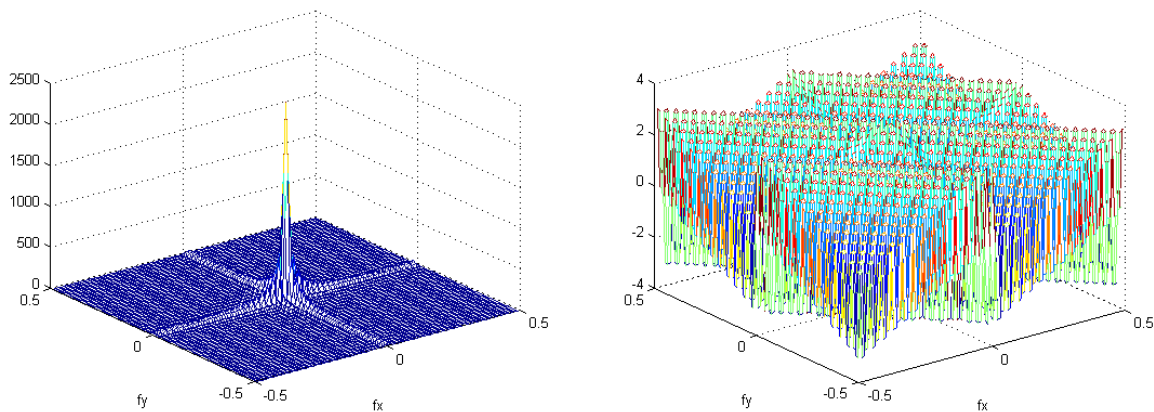
nella figura successiva è mostrato il risultato. Questa operazione ha evidenziato i bordi del quadrato. Come esempio di applicazione si veda l'esercizio *esercizio\_2\_5\_4* contenuto nel CD allegato al libro di testo del corso (L. Landini, Fondamenti di analisi di segnali biomedici, Plus editore).



**Figura 6.** Sinistra: immagine del modulo del gradiente in tutte le direzioni applicato al quadrato utilizzando la colormap grey e imagesc(.). Destra: grafico del modulo del gradiente in tutte le direzioni applicato al quadrato utilizzando mesh(.) e la colormap jet.

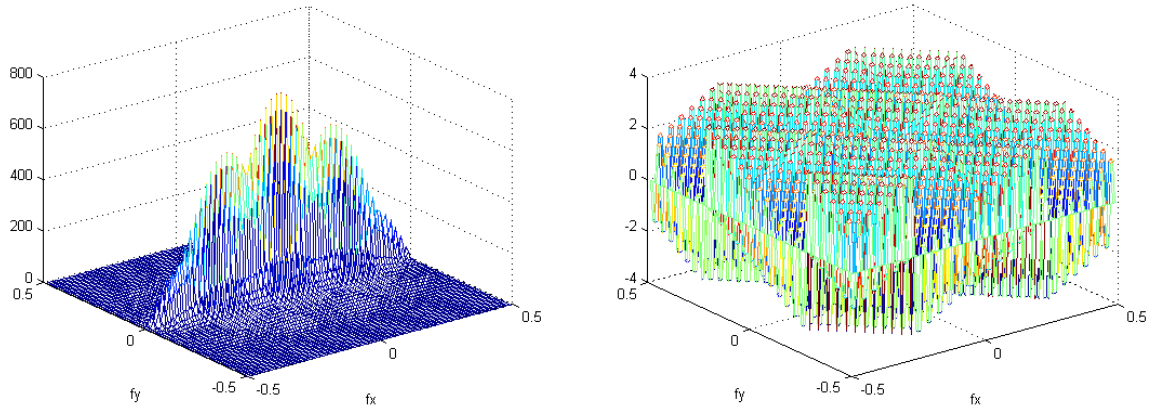
### **Trasformata di Fourier delle immagini filtrate**

Se calcoliamo la trasformata del cubo, prima e dopo l'operazione di gradiente lungo una direzione, si ottiene.

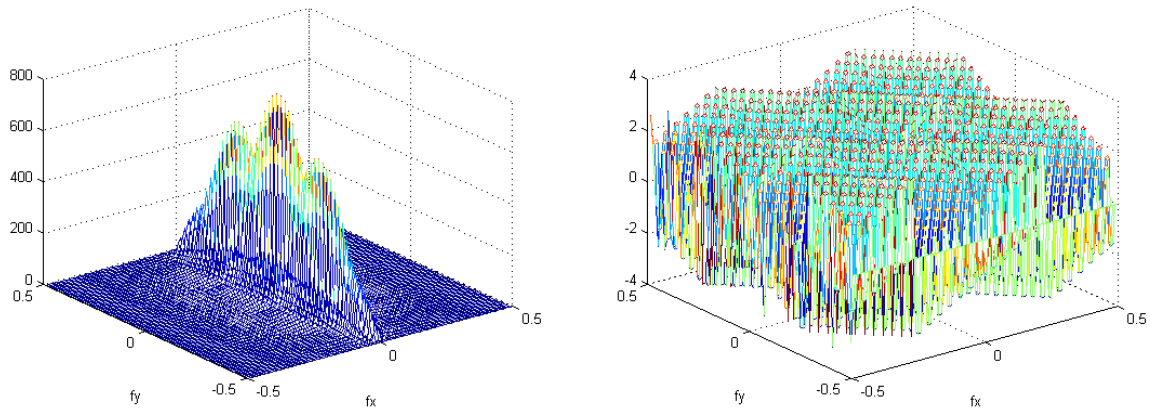


**Figura 7.** Sinistra: modulo della trasformata di Fourier del quadrato. Destra: fase della trasformata di Fourier del quadrato.

Le trasformate delle immagini gradiente lungo x e lungo y sono le seguenti. Si fa notare che le immagini ottenute dalla convoluzione hanno dimensione 102x102: di questo va tenuto conto nella taratura degli assi frequenziali.



**Figura 8.** Sinistra: modulo della trasformata di Fourier dell'immagine filtrata col filtro di Sobel lungo x, con  $N_x=N_y=102$ . Destra: fase della trasformata di Fourier del filtro di Sobel lungo x, con  $N_x=N_y=102$ .



**Figura 9.** Sinistra: modulo della trasformata di Fourier dell'immagine filtrata col filtro di Sobel lungo y, con  $N_x=N_y=102$ . Destra: fase della trasformata di Fourier del filtro di Sobel lungo y, con  $N_x=N_y=102$ .

Non è riportata la trasformata del modulo del gradiente.

N.B.

è possibile sfruttare le proprietà della TDF e ottenere le immagini filtrate, moltiplicando tra loro (punto per punto) le trasformate dell'immagine di partenza e della maschere e applicando la TDF inverse (ifft2) alla matrice risultante.

Perché l'operazione abbia successo le trasformate devono essere rappresentate sullo stesso numero di punti (in questo caso  $102 \times 102$ ) applicando uno zero padding opportuno.