Calcolo Numerico - Esercitazione 7

Laurea Triennale in Informatica a.a. 2023-2024

1 Image deblur

Il problema di deblur consiste nella ricostruzione di un immagine a partire da un dato acquisito mediante il seguente modello:

$$y = Ax + \eta \tag{1}$$

dove :

- y rappresenta l'immagine corrotta,
- x rappresenta l'immagine originale che vogliamo ricostruire
- A rappresenta l'operatore che applica il blur Gaussiano
- $\eta \sim \mathcal{N}(0, \sigma^2)$ rappresenta una realizzazione di rumore additivo con distribuzione Gaussiana di media $\mu = 0$ e deviazione standard σ

Exercise 1.1. Problema test

- Caricare l'immagine camera dal modulo skimage.data rinormalizzandola nel range $[0, 1]$.
- Applicare un blur di tipo gaussiano con deviazione standard 3 il cui kernel ha dimensioni 24×24. utilizzando la funzione. Utilizzare prima `cv2` (`open-cv`) e poi la trasformata di Fourier.
- Aggiungere rumore di tipo gaussiano, con $\sigma = 0.02$, usando la funzione `np.random.normal()`.
- Calcolare le metriche Peak Signal Noise Ratio (PSNR) e Mean Squared Error (MSE) tra l'immagine degradata e l'immagine esatta usando le funzioni `peak_signal_noise_ratio` e `mean_squared_error` disponibili nel modulo `skimage.metrics`.

Exercise 1.2. Soluzione naive Una possibile ricostruzione dell'immagine originale x partendo dall'immagine corrotta y è la soluzione naive data dal minimo del seguente problema di ottimizzazione:

$$x^* = \underset{x}{\operatorname{argmin}} \frac{1}{2} \|Ax - y\|_2^2 \tag{2}$$

- Utilizzando il metodo del gradiente coniugato implementato dalla funzione `minimize` della libreria `scipy`, calcolare la soluzione naive.
- Analizza l'andamento del PSNR e dell'MSE al variare del numero di iterazioni

Exercise 1.3. Soluzione regolarizzata Si consideri il seguente problema regolarizzato secondo Tikhonov

$$x^* = \underset{x}{\operatorname{argmin}} \frac{1}{2} \|Ax - y\|_2^2 + \lambda \|x\|_2^2 \tag{3}$$

- Utilizzando sia il metodo del gradiente che il metodo del gradiente coniugato calcolare la soluzione del problema regolarizzato.
- Analizzare l'andamento del PSNR e dell'MSE al variare del numero di iterazioni.
- Facendo variare il parametro di regolarizzazione λ, analizzare come questo influenza le prestazioni del metodo analizzando le immagini.

- Scegliere λ con il metodo di discrepanza.
- Scegliere λ attraverso test sperimentali come il valore che minimizza il valore del PSNR. Confrontare il valore ottenuto con quella della massima discrepanza.

Exercise 1.4. • Ripetere i punti precedenti utilizzando anche l'operatore downsampling con i seguenti fattori di scaling $sf = 2, 4, 8, 16$.

- Testare i punti precedenti su due immagini in scala di grigio con caratteristiche differenti (per esempio, un'immagine tipo fotografico e una ottenuta con uno strumento differente, microscopio o altro).
- Degradare le nuove immagini applicando, mediante le funzioni `gaussian_kernel()`, `psf_fft()`, l'operatore di blur con parametri:
  - $\sigma = 0,5$ dimensione del kernel 7×7 e 9×9
  - $\sigma = 1,3$ dimensione del kernel 5×5
  - Aggiungendo rumore gaussiano con deviazione standard nell' intervallo $(0, 0, 05]$.

Exercise 1.5 (Facoltativo TV)**.** Un'altra funzione adatta come termine di regolarizzazione è la Variazione Totale. Data x immagine di dimensioni $m \times n$ la variazione totale TV di x è definita come:

$$TV(x) = \sum_{i}^{n} \sum_{j}^{m} \sqrt{||\nabla x(i,j)||_2^2 + \epsilon^2} \tag{4}$$

Come nei casi precedenti il problema di minimo che si va a risolvere è il seguente:

$$x^* = \arg\min_x \frac{1}{2}||Ax - b||_2^2 + \lambda TV(x) \tag{5}$$

il cui gradiente ∇f è dato da

$$\nabla f(x) = (A^T Ax - A^T b) + \lambda \nabla TV(x) \tag{6}$$

- Utilizzando il metodo del gradiente e la funzione `minimize`, calcolare la soluzione del precendente problema di minimo regolarizzato con la funzione TV per differenti valori di λ, utilizzando le funzioni `totvar` e `grad_totvar`.

- Per calcolare il gradiente dell'immagine ∇u usiamo la funzione 'np.gradient' che approssima la derivata per ogni pixel calcolando la differenza tra pixel adiacenti. I risultati sono due immagini della stessa dimensione dell'immagine in input, una che rappresenta il valore della derivata orizzontale e l'altra della derivata verticale . Il gradiente dell'immagine nel punto (i, j) è quindi un vettore di due componenti, uno orizzontale contenuto e uno verticale.

- Per risolvere il problema di minimo è necessario anche calcolare il gradiente della variazione totale che è definito nel modo seguente

  $$\nabla TV(u) = -div\left(\frac{\nabla u}{\sqrt{||\nabla u||_2^2 + \epsilon^2}}\right) \tag{7}$$

  dove la divergenza è definita come

  $$div(F) = \frac{\partial F_x}{\partial x} + \frac{\partial F_y}{\partial y} \tag{8}$$

  $div(F)$ è la divergenza del campo vettoriale F, nel nostro caso F ha due componenti dati dal gradiente dell'immagine ∇u scalato per il valore $\frac{1}{\sqrt{||\nabla u||_2^2+\epsilon^2}}$. Per calcolare la divergenza bisogna calcolare la derivata orizzontale $\frac{\partial F_x}{\partial x}$ della componente x di F e sommarla alla derivata verticale $\frac{\partial F_y}{\partial y}$ della componente y di F. Per specificare in quale direzione calcolare la derivata con la funzione 'np.gradient' utilizziamo il parametro 'axis = 0' per l'orizzontale e 'axis = 1' per la verticale.

—