



UNIVERSITÀ DEGLI STUDI DI CAGLIARI

Facoltà di Ingegneria e Architettura

Corso di Laurea in Ingegneria Elettrica ed Elettronica

Metodi numerici per il deblurring

Docente relatore

Prof. Giuseppe Rodriguez

Candidato

Claudio Usai

(Matr. 70/87/65232)

Anno Accademico 2017/2018

Indice

Introduzione	5
1 Richiami di algebra lineare	7
1.1 Buona posizione e condizionamento	7
1.2 Problemi ai minimi quadrati	7
1.3 Metodo delle equazioni normali	8
1.4 Metodo del gradiente coniugato	9
1.5 Regolarizzazione di Tikhonov	10
1.6 Fattorizzazione SVD e GSVD	10
1.6.1 SVD	10
1.6.2 GSVD	11
1.6.3 Applicazione nei problemi ai minimi quadrati	12
1.7 La L-curve	13
1.8 L'operatore derivata lineare	14
2 Le immagini digitali	15
2.1 Rappresentazione digitale	15
2.2 La convoluzione	17
2.3 La FFT nell'applicazione dei filtri	19
2.4 La PSF	20
3 Il problema del blur	21
3.1 Presentazione del problema	21
3.2 Caso monodimensionale	22
3.2.1 $L = I$	22
3.2.2 $L = D_n$	25
3.3 Caso bidimensionale	29
3.3.1 $L = I$	31
3.3.2 $L = D_1$	34
3.3.3 $L = R$	37
4 Conclusioni	39
Bibliografia	41

Introduzione

In questa tesi viene affrontato il problema del deblurring in casi di filtri mal condizionati e vengono analizzati dei metodi numerici riconducendo il problema a un problema ai minimi quadrati. Oltre a un capitolo riguardante dei richiami di algebra lineare è presente un breve capitolo riguardante le immagini digitali in modo da fornire le premesse per affrontare il problema. Gli approcci utilizzati sono principalmente: il metodo del gradiente coniugato applicato al sistema ortogonale relativo al problema e, attraverso l'ausilio della decomposizione a valori singolari, del metodo di Tikhonov. Entrambi gli approcci vengono prima presentati nel caso più semplice monodimensionale e poi nel caso bidimensionale e cioè il deblur di immagini.

Capitolo 1

Richiami di algebra lineare

In questa sezione sono presenti dei richiami a concetti e definizioni di algebra lineare necessari per un trattamento sufficientemente esaustivo del tema trattato.

1.1 Buona posizione e condizionamento

Per trovare soluzioni a dei problemi pratici, nell'ambito ingegneristico e non, ci si avvale di modelli matematici che permettono di prevedere in maniera più o meno esatta l'andamento dei fenomeni fisici e delle variabili in gioco. Un requisito fondamentale per la risoluzione di un qualsiasi problema e quindi per sfruttare il modello matematico adatto è quello della buona posizione del problema. Un problema è detto ben posto se esso possiede in un prefissato campo di definizione, una e una sola soluzione e questa dipenda con continuità dai dati. In caso contrario viene detto mal posto. In una situazione reale ci si trova a utilizzare dati misurati sperimentalmente e cioè affetti da errore, se la soluzione del problema disturbato non dipende con continuità dai dati ovviamente si tratta di un problema mal posto. L'influenza del disturbo sulla soluzione viene detta condizionamento del problema. Viene detto numero di condizionamento assoluto $K = K(d)$

$$\|\delta x\| \leq K \|\delta d\|$$

dove $\delta(d)$ rappresenta la perturbazione dei dati d e δx la corrispondente perturbazione sulla soluzione x [1]. Analogamente viene definito il numero di condizionamento relativo come:

$$\frac{\|\delta x\|}{\|x\|} \leq K \frac{\|\delta d\|}{\|d\|}$$

1.2 Problemi ai minimi quadrati

Spesso è possibile ricondurre problemi di grosse dimensioni a dei sistemi lineari del tipo $A\mathbf{x} = \mathbf{b}$, talvolta però il numero delle equazioni ottenute è differente dal numero di incognite. In particolare siano: A una matrice $m \times n$, $\mathbf{b} \in \mathbb{R}^m$ e $\mathbf{x} \in \mathbb{R}^n$. Nel caso $m > n$ il problema si dice sovradeterminato, vi sono più equazioni che incognite e il sistema potrebbe non avere soluzioni. Nel caso $n > m$ si dice sottodeterminato, vi sono cioè più incognite che equazioni e il sistema potrebbe

avere infinite soluzioni. Il caso sovradeterminato è una situazione molto frequente, nel caso ideale in cui i dati non siano affetti da errore significa che nel sistema sono presenti $m - n$ equazioni linearmente dipendenti. Nel caso reale, in cui i dati sono affetti da errore, potrebbe capitare che nessuna equazione sia verificata esattamente ed è preferibile utilizzare un numero sovrabbondante di dati per ricavare una soluzione più accurata. Essendo il problema mal posto (non possiede una e una soluzione che dipende con continuità dai dati) si ricorre alla riformulazione di esso, non andando più alla ricerca della soluzione esatta ma quella in cui il residuo è minimo. Il problema viene così riformulato:

$$\min_{x \in \mathbb{R}^n} \|Ax - \mathbf{b}\|_2$$

1.3 Metodo delle equazioni normali

Una delle tecniche qui presentate per la risoluzione dei problemi ai minimi quadrati è il metodo delle equazioni normali, che risolve il sistema normale $A^T A \mathbf{x} = A^T \mathbf{b}$ associato al problema $A \mathbf{x} = \mathbf{b}$. Data la formula del gradiente di funzioni da \mathbb{R}^n in \mathbb{R} :

$$\nabla f(\mathbf{x}) = \left(\frac{\partial f}{\partial x_1}, \dots, \frac{\partial f}{\partial x_n} \right)^T$$

Considerando $f(\mathbf{x}) = \mathbf{b}^T \mathbf{x}$

$$\nabla(\mathbf{b}^T \mathbf{x}) = \mathbf{b}$$

mentre considerando $f(\mathbf{x}) = \mathbf{x}^T A \mathbf{x}$

$$\nabla(\mathbf{x}^T A \mathbf{x}) = A \mathbf{x} + A^T \mathbf{x}$$

Dove in caso A sia simmetrica si ha $\nabla(\mathbf{x}^T A \mathbf{x}) = 2A \mathbf{x}$. Applicando quanto visto al quadrato della norma del residuo

$$\|A \mathbf{x} - \mathbf{b}\|^2 = \mathbf{x}^T A^T A \mathbf{x} - 2\mathbf{x}^T A^T \mathbf{b} + \mathbf{b}^T \mathbf{b}$$

si ottiene:

$$\frac{1}{2} \nabla(\|A \mathbf{x} - \mathbf{b}\|^2) = A^T A \mathbf{x} - A^T \mathbf{b}$$

Nel caso in cui A è a rango pieno, la matrice $A^T A$ è invertibile e la soluzione del sistema lineare è unica

$$\mathbf{x}_{LS} = (A^T A)^{-1} A^T \mathbf{b}$$

Se A non è a rango pieno la matrice $A^T A$ risulta singolare, tuttavia il sistema resta consistente poiché $A^T \mathbf{b}$ appartiene all'immagine di $A^T A$. In questo caso vi sono infiniti vettori che soddisfano l'equazione ma solitamente viene considerata come soluzione \mathbf{x}_{LS} , dove però l'inversa di $A^T A$ viene sostituita dalla pseudo-inversa o inversa di Moore-Penrose che è un'inversa sinistra di A , ma non un'inversa destra.

1.4 Metodo del gradiente coniugato

Un metodo iterativo per risolvere i problemi ai minimi quadrati è quello del gradiente coniugato applicato alle equazioni normali, a cui è applicabile poiché la matrice $A^T A$ è simmetrica definita positiva. L'idea di base del metodo è quella di trovare il punto in cui il gradiente della funzione viene minimizzato attraverso un metodo iterativo che converge in n iterazioni.

Sia A una matrice simmetrica definita positiva, consideriamo la forma quadratica

$$\phi(\mathbf{y}) = \frac{1}{2} \mathbf{y}^T A \mathbf{y} - \mathbf{y}^T \mathbf{b}$$

tale funzione è minima nel punto in cui si annulla il suo gradiente

$$\nabla \phi(\mathbf{y}) = \frac{1}{2} (A + A^T) \mathbf{y} - \mathbf{b} = A \mathbf{y} - \mathbf{b} = 0$$

La soluzione è quindi equivalente a quella del sistema lineare $A \mathbf{x} = \mathbf{b}$. Il metodo è del tipo

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \alpha_k \mathbf{p}^{(k)} \quad (1.1)$$

a partire da un vettore iniziale $\mathbf{x}^{(0)}$, lungo direzioni \mathbf{p} e con passi di lunghezza α_k . Qualunque sia la direzione di discesa \mathbf{p}^k , possiamo determinare il minimo di $\phi(\mathbf{x}^{(k+1)})$ rispetto alla variazione di $\alpha = \alpha_k$. Infatti da

$$\begin{aligned} \phi(\mathbf{x}^{(k+1)}) &= \frac{1}{2} (\mathbf{x}^{(k)} + \alpha \mathbf{p}^{(k)})^T A (\mathbf{x}^{(k)} + \alpha \mathbf{p}^{(k)}) - (\mathbf{x}^{(k)} + \alpha \mathbf{p}^{(k)})^T \mathbf{b} \\ &= \phi(\mathbf{x}^{(k+1)}) + \frac{1}{2} (\mathbf{p}^{(k)})^T A \mathbf{p}^{(k)} \cdot \alpha^2 - (\mathbf{p}^{(k)})^T \mathbf{r}^{(k)} \cdot \alpha \end{aligned}$$

quindi

$$\frac{d}{d\alpha} \phi(\mathbf{x}^{(k+1)}) = (\mathbf{p}^{(k)})^T A \mathbf{p}^{(k)} \cdot \alpha - (\mathbf{p}^{(k)})^T \mathbf{r}^{(k)} = 0$$

il cui minimo si ottiene per

$$\alpha_k = \frac{(\mathbf{p}^{(k)})^T \mathbf{r}^{(k)}}{(\mathbf{p}^{(k)})^T A \mathbf{p}^{(k)}}$$

Definizione 1.1 Definiamo un vettore $\mathbf{x}^{(k)}$ **ottimale** rispetto ad una direzione \mathbf{p} se

$$\phi(\mathbf{x}^{(k)}) \leq \phi(\mathbf{x}^{(k)} + \lambda \mathbf{p}), \quad \forall \lambda \in \mathbb{R}.$$

Se $\mathbf{x}^{(k)}$ è ottimale rispetto ad ogni direzione \mathbf{p} di un sottospazio V , diremo che è ottimale rispetto a V .

In particolare è possibile dimostrare che il vettore $\mathbf{x}^{(k)}$ è ottimale rispetto a \mathbf{p} se e solo se la direzione \mathbf{p} è ortogonale al residuo $\mathbf{r}^{(k)}$, cioè

$$\mathbf{p}^T \mathbf{r}^{(k)} = 0$$

Il fatto che il vettore $\mathbf{x}^{(k)}$ (1.1) sia ottimale rispetto alle direzioni $\mathbf{p}^{(i)}$, ($i = 0, \dots, k$) significa che lungo tali direzioni non è possibile far diminuire ulteriormente il valore della funzione $\phi(\mathbf{y})$ e spiega la convergenza in un numero finito di iterazioni. [3]

1.5 Regolarizzazione di Tikhonov

La difficoltà nella risoluzione di problemi mal posti risiede nel decadimento a zero dei valori singolari nella matrice A . Per riuscire a isolare una soluzione che possa essere accettabile e coerente con il problema è quindi necessario aggiungere una quantità di informazioni che facilitano la risoluzione del problema accettando il compromesso di avere uno scostamento più o meno grande rispetto alla soluzione ideale in funzione del grado di regolarizzazione che viene utilizzato. In generale l'approccio alla regolarizzazione richiede che la seguente quantità sia minimizzata

$$\Omega(x) = \|L(\mathbf{x} - \mathbf{x}^*)\|_2^2$$

dove L solitamente è la matrice identità I o la matrice dell'operatore derivata, mentre con \mathbf{x} e \mathbf{x}^* vengono indicati rispettivamente la soluzione e una stima iniziale di essa. La regolarizzazione di Tikhonov sfrutta un parametro di regolarizzazione λ con il quale si può dosare il grado di regolarizzazione. [2]

$$\mathbf{x}_\lambda = \arg \min \{ \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 + \lambda^2 \|L(\mathbf{x} - \mathbf{x}^*)\|_2^2 \} \quad (1.2)$$

Con valori di λ grandi si ha un alto grado di regolarizzazione e si ha una norma della soluzione molto piccola al prezzo di una norma del residuo grande e viceversa per valori piccoli di λ . Va inoltre precisato che l'assunzione alla base dell'equazione precedente è che gli errori non siano di tipo sistematico e che la loro matrice di covarianza siano proporzionali alla matrice identità.

1.6 Fattorizzazione SVD e GSVD

La fattorizzazione SVD (Singular Value Decomposition, decomposizione a valori singolari) è un potente strumento di analisi dei problemi mal posti. La GSVD è la generalizzazione per due matrici A ed L invece ha importanti implicazioni nella regolarizzazione come nel caso della equazione (1.2).

1.6.1 SVD

Sia $A \in \mathbb{R}^{m \times n}$ una matrice rettangolare con $m \geq n$, la fattorizzazione SVD è data da

$$A = U \Sigma V^T = \sum_{i=1}^n \mathbf{u}_i \sigma_i \mathbf{v}_i^T \quad (1.3)$$

dove U e V sono matrici ortogonali, quindi $U^T U = V^T V = I$, e dove Σ è una matrice diagonale i cui elementi, detti valori singolari della matrice A , sono positivi e decrescenti all'aumentare dell'indice: $\sigma_1 \geq \dots \geq \sigma_n \geq 0$. La matrice V è composta dagli autovettori della matrice $A^T A$ mentre la matrice U è composta dagli autovettori della matrice $A A^T$, si nota inoltre che

$$K(A) = \frac{\sigma_1}{\sigma_n}$$

L'implicazione è notevole: essendo strettamente collegata alla decomposizione in autovettori di AA^T e $A^T A$ la SVD è unica. Inoltre empiricamente si nota che i valori singolari decrescono senza un particolare intervallo e che i cambi di segno all'interno delle matrici U e V aumentano all'aumentare dell'indice i . Consideriamo adesso le seguenti relazioni

$$\begin{cases} A\mathbf{v}_i = \sigma_i \mathbf{u}_i \\ \|A\mathbf{v}_i\|_2 = \sigma_i \end{cases} \quad i = 1, \dots, n$$

Confrontando un valore piccolo σ_i con $\|A\|_2 = \sigma_1$ è immediato notare che esiste una combinazione lineare delle colonne di A , tale che $\|A\mathbf{v}_i\|_2 = \sigma_i$, questo implica che A non è a rango pieno e che i vettori \mathbf{v}_i sono numericamente molto piccoli. Se ne deduce che una matrice A relativa a un problema discreto mal posto è necessariamente altamente mal condizionata. Dalla (1.3) si nota inoltre che all'aumentare dell'indice il fattore σ_i pesa le componenti della matrice finale completa partendo dalla più significativa (valore di σ_i maggiore) fino alla meno significativa, questo aspetto è di grande importanza come si vedrà per la TSVD (Truncated SVD).

1.6.2 GSVD

La GSVD di una coppia di matrici (A, L) è la generalizzazione della SVD dove però i valori singolari sono relativi alla coppia di matrici $(A^T A, L^T L)$, assumendo $A \in \mathbb{R}^{m \times n}$ e $L \in \mathbb{R}^{p \times n}$ dove $m \geq n \geq p$, ciò è sempre verificato nei problemi mal posti e discreti. La GSVD è quindi la decomposizione a valori singolari di A e L :

$$A = U \begin{pmatrix} \Sigma & 0 \\ 0 & I_{n-p} \end{pmatrix} X^{-1}, \quad L = V \begin{pmatrix} M & 0 \end{pmatrix} X^{-1}$$

dove $U \in \mathbb{R}^{m \times n}$ e $V \in \mathbb{R}^{p \times p}$ sono ortonormali, $X \in \mathbb{R}^{n \times n}$ è invertibile e Σ e M sono matrici diagonali $p \times p$:

$$\Sigma = \text{diag}(\sigma_1, \dots, \sigma_p), \quad M = \text{diag}(\mu_1, \dots, \mu_p)$$

dove gli elementi sono negativi e ordinati in modo tale che:

$$0 \leq \sigma_1 \leq \dots \leq \sigma_p \leq 1, \quad 1 \geq \mu_1 \geq \dots \geq \mu_p > 0$$

e normalizzati tali che:

$$\sigma_i^2 + \mu_i^2 = 1, \quad i = 1, \dots, p$$

I valori singolari generalizzati sono decrescenti e definiti

$$\gamma_i = \frac{\sigma_i}{\mu_i}, \quad i = 1, \dots, p$$

Va precisato che le matrici U, Σ e V sono matrici differenti rispetto alle stesse relative alla SVD. Se si considera la GSVD della coppia (A, L) tuttavia i valori singolari risulteranno uguali tranne che per l'ordine invertito degli elementi. Nel caso generale non c'è nessuna relazione tra i valori singolari di una data SVD e di una GSVD, tuttavia nei problemi discreti, è possibile notare delle analogie. Infatti dato che la matrice L viene scelta ben condizionata, risulta ben condizionata anche X e ciò

implica che $\gamma_i = \sigma_i(1 - \sigma_i^2)^{1/2} \approx \sigma_i$, per valori piccoli di σ_i deve rappresentare il mal condizionamento della matrice A decadendo a 0 come nella SVD, in quanto L e quindi X sono ben condizionate. Inoltre se nella SVD vi sono cambi di segno nella matrice destra degli autovettori V , nella GSVD si verifica lo stesso per la matrice X : al decrescere di γ_i aumentano i cambi di segno nel vettore x_i della matrice X . [4]

1.6.3 Applicazione nei problemi ai minimi quadrati

Nel caso particolare dei problemi ai minimi quadrati la fattorizzazione SVD (e GSVD) risulta particolarmente utile per la facilità con cui è possibile applicare la regolarizzazione di Tikhonov descritta precedentemente. La matrice inversa di una matrice fattorizzata è calcolabile in modo rapido grazie alle proprietà delle matrici che compongono la fattorizzazione stessa, infatti nel caso la matrice A sia quadrata

$$A = U\Sigma V^T \quad A^{-1} = V\Sigma^{-1}U^T$$

Nel caso invece in cui A sia rettangolare la formula analoga per la pseudoinversa è $A^+ = V\Sigma^+U^T$. Considerando il problema ai minimi quadrati

$$\min_{x \in \mathbb{R}^n} \|Ax - \mathbf{b}\|_2$$

è possibile ottenere la soluzione

$$\mathbf{x}_{LSQR} = \sum_{i=1}^n \frac{\mathbf{u}_i^T \mathbf{b}}{\sigma_i} \mathbf{v}_i \quad (1.4)$$

dove definiamo $|\mathbf{u}_i^T \mathbf{b}|$ come i coefficienti di Fourier. Sia $\mathbf{b} = \bar{\mathbf{b}} + \mathbf{e}$ dove $\bar{\mathbf{b}}$ è la soluzione non perturbata e \mathbf{e} è la perturbazione o errore, introduciamo quindi la condizione di Picard discreta di un problema mal posto:

Definizione 1.2 (Condizione di Picard discreta) *La soluzione non perturbata $\bar{\mathbf{b}}$ di un problema mal posto con matrice di regolarizzazione L soddisfa la condizione di Picard discreta se i coefficienti di Fourier $|\mathbf{u}_i^T \mathbf{b}|$ decadono a zero più velocemente dei valori singolari γ_i .*

Se ciò non è verificato infatti la soluzione data dalla 1.4 presenta errori di ordine elevato dovuto ai valori singolari γ_i infinitesimi che producono un rapporto sempre più elevato per indici crescenti. Le possibilità sono quindi di "troncare" i valori singolari oltre un certo indice oppure applicare la 1.2 ovvero la regolarizzazione di Tikhonov. Con quest'ultima la soluzione è data da:

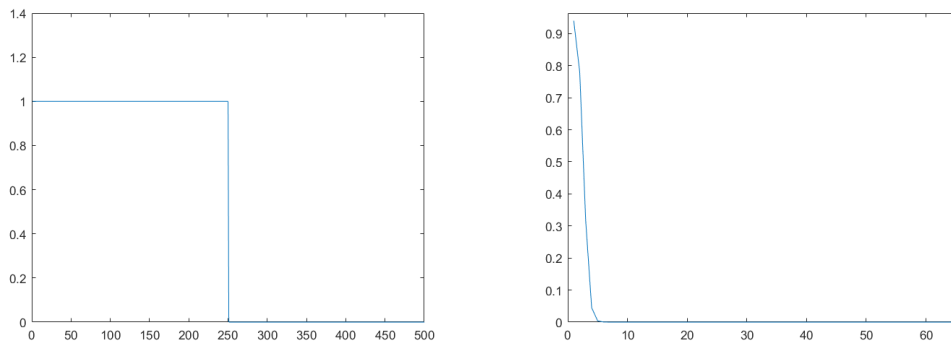
$$\mathbf{x}_{\text{reg}} = \sum_{i=1}^n f_i \frac{\mathbf{u}_i^T \mathbf{b}}{\sigma_i} \mathbf{v}_i \quad \text{se } L = I_n$$

$$\mathbf{x}_{\text{reg}} = \sum_{i=1}^p f_i \frac{\mathbf{u}_i^T \mathbf{b}}{\sigma_i} \mathbf{x}_i + \sum_{i=p-1}^n (\mathbf{u}_i^T \mathbf{b}) \mathbf{x}_i \quad \text{se } L \neq I_n$$

dove f_i sono i fattori di filtro che fanno sì che l'azione delle componenti della soluzione dovuti ai σ_i piccoli vengano filtrati. Nel caso della regolarizzazione di Tikhonov i fattori di filtro sono così definiti:

$$f_i = \frac{\sigma_i^2}{\sigma_i^2 + \lambda^2} \quad \text{se } L = I_n, \quad f_i = \frac{\gamma_i^2}{\gamma_i^2 + \lambda^2} \quad \text{se } L \neq I_n$$

dal valore di λ dipende quindi il filtraggio, perciò è estremamente importante la scelta di tale costante. [5] È possibile visualizzare i fattori di filtro che nel caso della TSVD risultano andare a zero in modo drastico come una funzione a gradino, mentre con l'uso di Tikhonov la variazione è più graduale (Figura 1.1).



(a) Andamento dei fattori di filtro con l'uso di TSVD (b) Andamento dei primi 60 fattori di filtro con l'uso di Tikhonov

Figura 1.1

1.7 La L-curve

La L-curve (o curva a L) è uno strumento grafico utilizzato per visualizzare l'effetto sulla soluzione della quantità di regolarizzazione e per sceglierne il parametro λ . Il grafico viene rappresentato in scala logaritmica e presenta appunto una forma a L (Figura 1.2).

Nell'asse delle ascisse viene indicata la norma del residuo mentre in quello delle ordinate la norma della soluzione regolarizzata, al variare di λ . Più la curva diventa verticale maggiore sarà il residuo e quindi l'errore "non regolarizzato", più il grafico diventa orizzontale e maggiore sarà la norma della soluzione regolarizzata. [6] Attraverso la curva è possibile individuare il "corner" ovvero il punto in cui la curva passa da verticale a orizzontale che altro non è che il valore nel quale le due norme vengono dosate nella maniera ottimale. Non è ovviamente possibile, partendo da un termine noto perturbato, ottenere una soluzione esatta e il valore di λ nel punto di "corner" rappresenta spesso il compromesso migliore in termini di risultato affidabile e coerente.

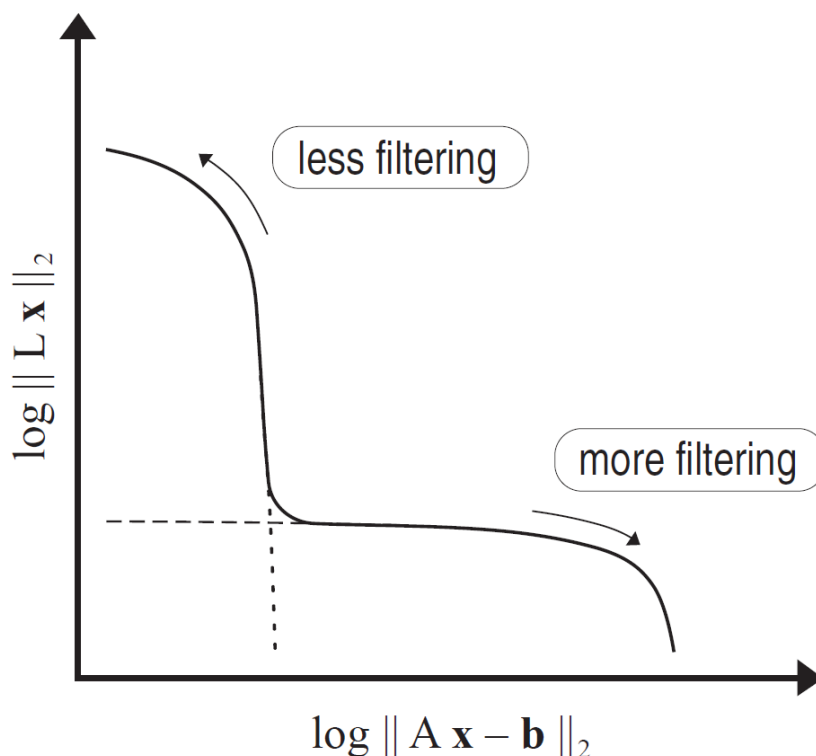


Figura 1.2: L-curve

1.8 L'operatore derivata lineare

Nel campo discreto l'operatore di derivazione lineare viene descritto attraverso le differenze finite come una matrice di Toeplitz in cui ogni riga applica il metodo delle differenze finite centrato in un elemento del vettore per cui è moltiplicato. Ad esempio per il secondo ordine la derivata viene così approssimata attraverso le differenze finite.

$$f''(x) \approx \frac{\delta_h^2[f](x)}{h^2} = \frac{\frac{f(x+h)-f(x)}{h} - \frac{f(x)-f(x-h)}{h}}{h} = \frac{f(x+h) - 2f(x) + f(x-h)}{h^2}$$

Sia n il numero delle righe del vettore da derivare e d l'ordine della derivata, la matrice D_d sarà di dimensione $(n-d) \times n$ del tipo:

$$D_2 = \begin{bmatrix} 1 & -2 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & -2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & -2 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & -2 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & -2 & 1 \end{bmatrix}, \text{ con } d = 2, n = 7$$

L'operatore si presta ad essere utilizzato come matrice di regolarizzazione L nel metodo di Tikhonov (eq. 1.2) in quanto tende a rendere coerente la continuità della derivata di ordine scelto rispetto alla soluzione, situazione che si verifica nel caso in cui i dati sono misurazioni reali.

Capitolo 2

Le immagini digitali

In questo capitolo verranno affrontate alcuni concetti basilari dell' Image Processing come la rappresentazione delle immagini in formato digitale e l'applicazione di filtri in termini matematici.

2.1 Rappresentazione digitale

Un'immagine digitale è un insieme di dati strutturati, i dati in questione sono segnali o misurazioni, solitamente del mondo reale, acquisiti attraverso uno o più sensori ovvero fotocamere. La rappresentazione in formato digitale può avvenire in diversi modi a seconda delle necessità e del tipo di immagine in questione, il formato può essere più o meno compresso, la risoluzione è spesso variabile ed è possibile che sia a colori (con diversa profondità), in bianco e nero o in scala di grigi. Nel corso di questo elaborato verranno utilizzate immagini in scala di grigi per semplicità di esposizione, pur essendo i metodi equivalenti per le immagini a colori.



Figura 2.1: Cameraman

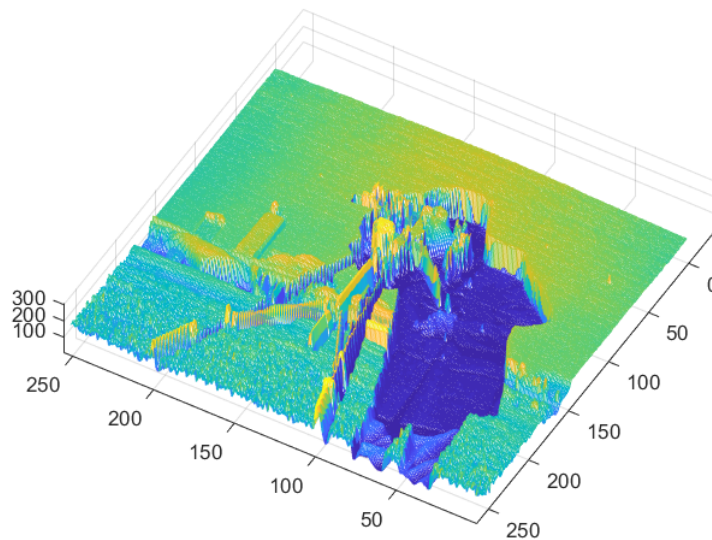


Figura 2.2: Visualizzazione in 3 dimensioni dell'immagine 2.1

Nell'immagine 2.1 in scala di grigi ogni pixel assume un valore compreso tra 0 e 255 (o fra 0 e 1 a seconda dell'uso di `int` o `float`). Utilizzando le routine `mesh` di MATLAB è possibile visualizzare l'immagine come funzione in 3 dimensioni: altezza, larghezza, intensità del pixel. Ogni riga (o colonna) è quindi un segmento della curva data dall'intensità dei pixel in essa, presa singolarmente una riga dell'immagine si presenta come una qualsiasi funzione matematica.

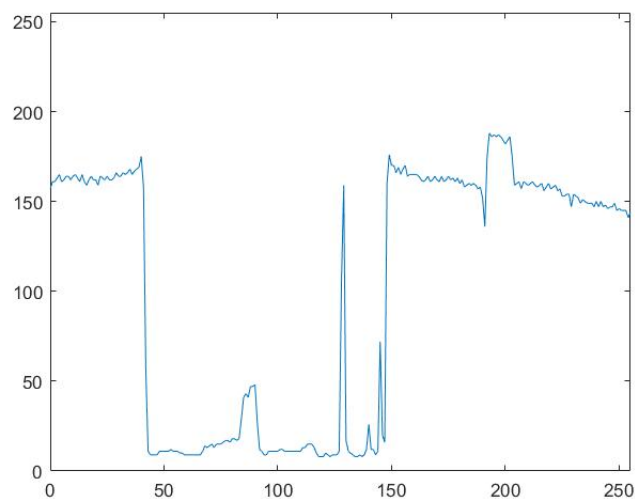


Figura 2.3: Riga 120 della figura 2.1

2.2 La convoluzione

La convoluzione permette di applicare un filtro a una funzione:

$$K(x) * f(x) = \int_{-\infty}^{+\infty} K(x-y)f(y)dy = g(x) \quad (2.1)$$

che in forma discreta diventa

$$g[x] = \sum_{-\infty}^{+\infty} K[x-y] \cdot f[y] \quad (2.2)$$

È immediato notare che il primo fattore della sommatoria dipende solo dalla differenza $x - y$, utilizzando gli indici (i, j) al posto di (x, y)

$$\sum_j K_{i-j} \cdot f_j = g_i$$

In questo modo è facile ricondurre l'operazione di convoluzione ad una moltiplicazione matriciale del tipo

$$K\mathbf{f} = \mathbf{g}$$

dove \mathbf{f} è la funzione da filtrare, \mathbf{g} quella filtrata e K la matrice di convoluzione, cioè una matrice a bande di Toeplitz del tipo

$$K = \begin{bmatrix} \mathbf{k}_0 & \mathbf{k}_{-1} & \cdot & \cdot & \cdot & \cdot & \mathbf{k}_{-n+1} \\ \mathbf{k}_1 & \mathbf{k}_0 & \mathbf{k}_{-1} & \cdot & \cdot & \cdot & \cdot \\ \cdot & \mathbf{k}_1 & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \mathbf{k}_{-1} \\ \mathbf{k}_{n-1} & \cdot & \cdot & \cdot & \cdot & \mathbf{k}_1 & \mathbf{k}_0 \end{bmatrix} \quad (2.3)$$

Prendendo come esempio la curva in figura 2.3 e operando una convoluzione con una gaussiana come quella in figura 2.4 si ottiene una variazione più morbida tra valori adiacenti, che se riportata in 2 dimensioni corrisponde ad un effetto di sfocatura.

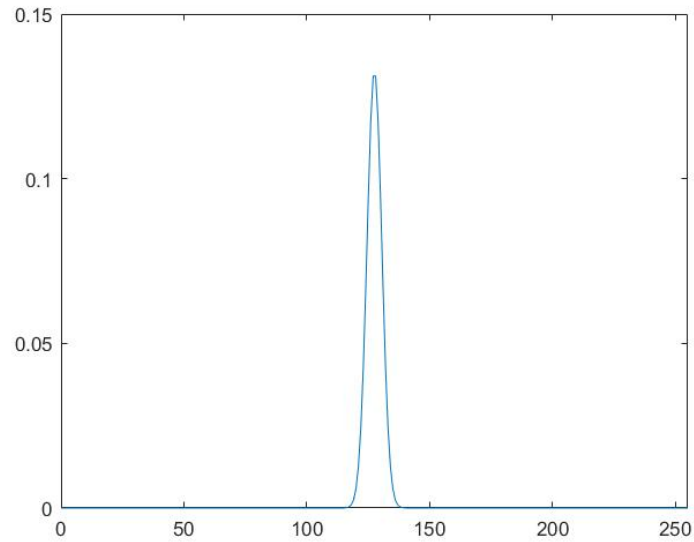


Figura 2.4: Curva gaussiana $\sigma = 3$

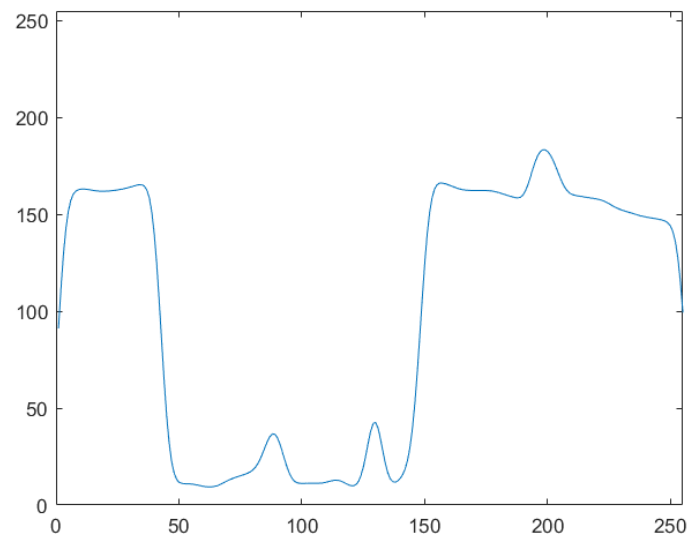


Figura 2.5: Risultato della convoluzione

L'operazione che permette di applicare un filtro ad un'immagine è la convoluzione applicata in 2 dimensioni:

$$K(x, y) * f(x, y) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} K(x-z, y-t) f(z, t) dz dt = g(x, y) \quad (2.4)$$

che in forma discreta diventa

$$\sum_r \sum_s K_{i-r, j-s} f_{r,s} = g_{i,j} \quad (2.5)$$

Analogamente al ragionamento precedente la sommatoria dipende dalle differenze degli indici $i-r, j-s$ e sebbene non sia possibile ricondurre K ad una matrice di Toeplitz in quanto la dipendenza è a doppio indice, è tuttavia possibile ricondurla a un tipo di matrice strutturata. In particolare questo tipo di matrice è chiamata BTTB che sta per "Block-Toeplitz with Toeplitz Block" appunto perchè è una matrice di Toeplitz come descritta in 2.6 in cui però ogni elemento T_k è una matrice Toeplitz del tipo descritto precedentemente in 2.3.

$$T = \begin{bmatrix} T_0 & T_{-1} & \cdot & \cdot & \cdot & T_{-n+1} \\ T_1 & T_0 & T_{-1} & \cdot & \cdot & \cdot \\ \cdot & T_1 & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & T_{-1} \\ T_{m-1} & \cdot & \cdot & \cdot & T_1 & T_0 \end{bmatrix} \quad (2.6)$$

Supponendo di trasformare la funzione discreta f in un vettore colonna composto dalle colonne della matrice f è possibile ricondurre l'operazione di convoluzione ad un prodotto matriciale e quindi ad un sistema lineare:

$$Tf = g \iff K(x, y) * f(z, t) = g(x, y)$$

2.3 La FFT nell'applicazione dei filtri

La FFT (*Fast Fourier Transform*) ricopre un ruolo importante nell'applicazione di filtri, il motivo sta nella semplificazione del numero di operazioni dovuto alla rapidità dell'algoritmo e in particolare al teorema di convoluzione.

Teorema 2.3.1 (Teorema di convoluzione) *Siano f e g due funzioni dotate di trasformata di Fourier la cui convoluzione è indicata da $f * g$ anche la loro convoluzione è dotata di trasformata di Fourier e tra esse vale la seguente relazione*

$$\mathcal{F}\{f * g\} = F(k)G(k)$$

Segue che

$$\mathcal{F}^{-1}\{F(k)G(k)\} = f * g$$

da cui ne deriva un miglioramento della velocità computazionale nell'applicazione di filtri a segnali, soprattutto nei casi di applicazioni in tempo reale. Le routine di MATLAB `fft` e `fft2` sono rispettivamente le FFT in una e due dimensioni, ciò permette applicazioni anche a segnali bidimensionali, come le immagini.

2.4 La PSF

PSF dall'inglese "point spread function" rappresenta la risposta all'impulso di un sensore, solitamente in condizioni di disturbo sistematico, la funzione ricavata permette di avere una stima della funzione di degradazione dell'immagine ottenuta sperimentalmente. L'impulso nella realtà viene rappresentato da un punto o da un oggetto puntiforme nelle condizioni nella quale sono state acquisite le immagini degradate. Nell'acquisizione di immagini di tipo astronomico viene spesso utilizzato questo metodo, puntando il sensore verso una stella isolata che funge da impulso. La conoscenza della funzione di degradazione permette di poter risolvere il problema inverso e ricavare quindi l'immagine restaurata. Nella realtà, essendo la risposta impulsiva solo un'approssimazione della funzione di degradazione ed essendo presente un fattore di rumore casuale l'immagine ottenuta attraverso la risoluzione del problema inverso, non è possibile ottenere l'immagine pura. Tuttavia i risultati spesso permettono, a seconda dello scopo, un sufficiente sfruttamento dell'immagine restaurata.

Capitolo 3

Il problema del blur

In questo capitolo verranno presentati i risultati degli esperimenti di risoluzione del problema del blur, attraverso metodi numerici associato a diversi tipi di immagine con diversi livelli di dettaglio.

3.1 Presentazione del problema

L'effetto di blur, o sfocatura, in un'immagine si presenta in condizioni di acquisizione non ottimali, nel caso di acquisizione di immagini astronomiche dalla Terra l'effetto di sfocatura è dovuto all'atmosfera terrestre, fenomeno noto come "seeing". L'approccio al problema presuppone la conoscenza della PSF e presenta dei metodi numerici di risoluzione prima in una dimensione e poi in due dimensioni, con uso di regolarizzazione. In tutti i casi trattati è stata aggiunta una componente di rumore utilizzando una matrice di opportune dimensioni generata dalla funzione `randn` moltiplicata per un fattore di 10^{-2} . Come PSF è stata utilizzata una gaussiana con $\sigma = 3$. Le routine di regolarizzazione utilizzate sono quelle presenti nel toolbox "Regularization Tools" [9]:

Tabella 3.1: Routine di regolarizzazione utilizzate

<code>csvd</code>	<code>cgsvd</code>	<code>get_l</code>	<code>tsvd</code>	<code>cgl</code>
<code>tikhonov</code>	<code>l_curve</code>	<code>plot_lc</code>	<code>l_corner</code>	<code>std_form</code>

Inizialmente il problema è stato affrontato utilizzando la convoluzione nel dominio della frequenza, tuttavia la scelta della PSF altamente mal condizionata ha reso impossibile l'utilizzo di tale metodo, restituendo infatti matrici di NaN. Il problema è stato quindi affrontato con i metodi per i minimi quadrati presentati nel Capitolo 1, è stato quindi analizzato il sistema lineare $A\mathbf{x} = \mathbf{b}$. Dove A è la matrice di Toeplitz generata a partire dalla gaussiana in una e in due dimensioni, \mathbf{x} il vettore di partenza e \mathbf{b} il risultato della convoluzione. Per il problema in due dimensioni è stata generata una BTTB partendo dalla gaussiana in due dimensioni e linearizzata la matrice dell'immagine di partenza in modo da ottenere un vettore colonna, il risultato è stato poi riportato alle dimensioni originali.

3.2 Caso monodimensionale

3.2.1 $L = I$

Per il problema in una dimensione è stata utilizzata una riga dell'immagine rappresentante il cameraman (2.1) come curva campione. Alla curva è stata effettuata l'operazione di convoluzione con la gaussiana utilizzando il metodo descritto nel capitolo precedente utilizzando la matrice di Toeplitz, con i seguenti risultati. I metodi di risoluzione applicati sono: metodo del gradiente coniugato, TSVD, Tikhonov. Tutti i metodi sono stati utilizzati con matrice di regolarizzazione $L = I$ e successivamente utilizzando l'operatore di derivata lineare.

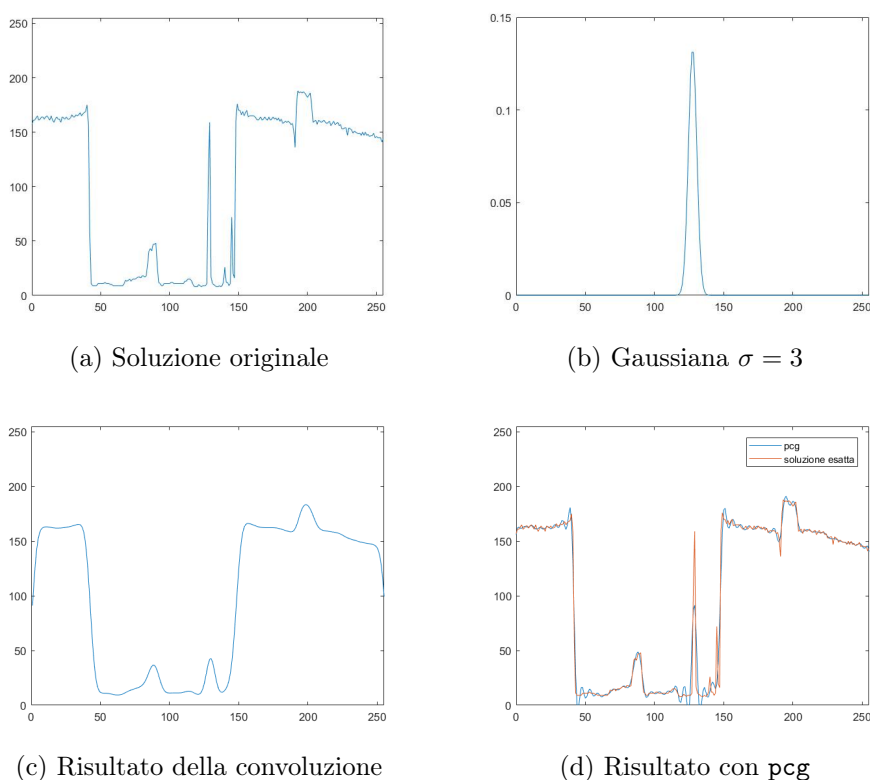
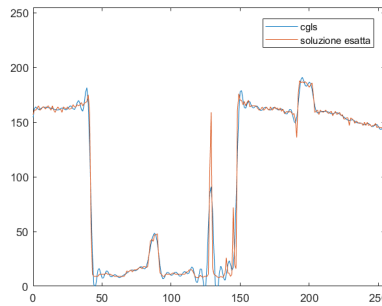


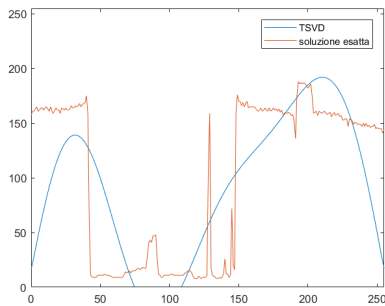
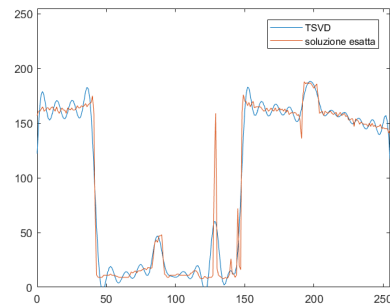
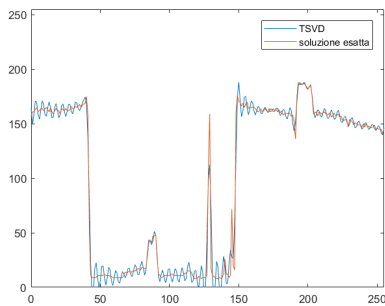
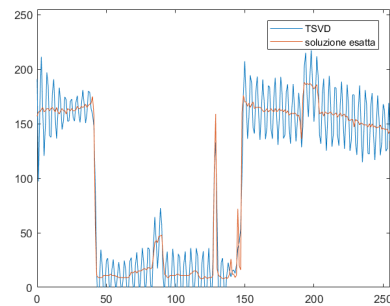
Figura 3.1

Come primo approccio è stato applicato il metodo del gradiente coniugato, utilizzando la routine di MATLAB `pcg`, al sistema $A^T Ax = A^T b$, dove A è la matrice di Toeplitz costruita partendo dalla gaussiana e b il vettore risultante dalla convoluzione. Il metodo converge in 141 iterazioni con un residuo relativo $\bar{r} = 9.473 \cdot 10^{-7}$.

Utilizzando il metodo del gradiente coniugato implicito, routine `cglis`, cambia l'ordine delle operazioni che diventa $A^T(A\mathbf{x})$ e la convergenza è più rapida di circa la metà delle iterazioni. È comunque presente una componente di rumore che fa oscillare la soluzione trovata rispetto a quella esatta in un intervallo tuttavia accettabile.

Figura 3.2: Risultato `cgls` 70 iterazioni

Dopo aver decomposto la matrice A con la SVD nelle matrici U , S e V è stata applicata la routine `tsvd` con diversi parametri. La funzione restituisce oltre che il risultato della SVD troncata, anche i residui `rho` e `eta` che vengono utilizzati dalla routine `plot_lc` per rappresentare la L-curve e dalla routine `l_corner` che ricava il parametro di regolarizzazione ottimale, che corrisponde allo "spigolo" della L nel grafico.

(a) $k = 5$ (b) $k = 50$ (c) $k = 110$ (d) $k = 120$ Figura 3.3: Soluzioni ottenute con `tsvd`

Dai grafici in figura 3.3 si può dedurre che la TSVD aumenta la precisione della soluzione all'aumentare del parametro, i valori singolari aggiungono precisione al crescere dell'indice fino a un punto in cui diventano rumore. In particolare è facile notare come per $k = 5$ (Fig 3.3a) la curva approssimata assume una forma che va-

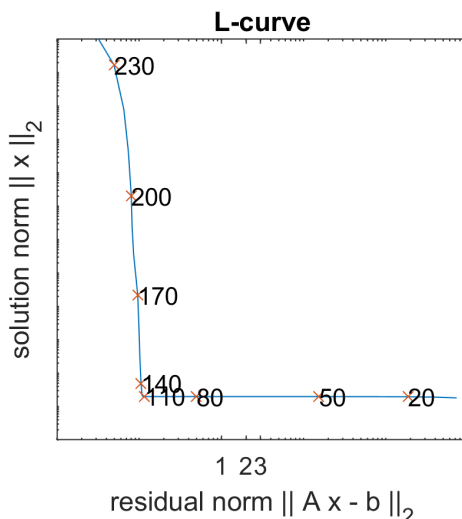
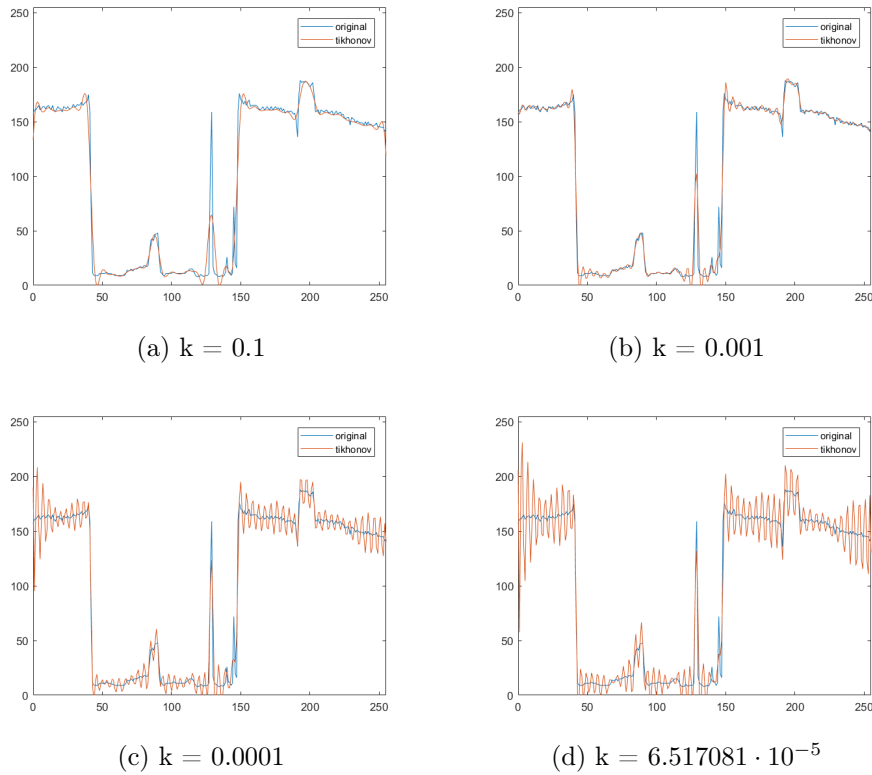
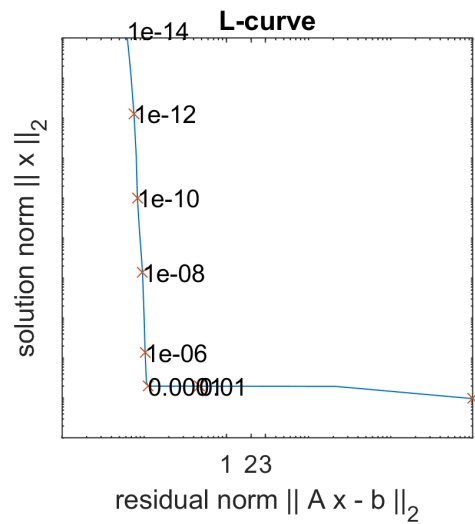


Figura 3.4: L-curve relativa al metodo `tsvd`, curva con parametro ottimale in 3.3c

gamente ricorda quella originale ma che ricalca le caratteristiche zone in cui i valori sono alti rispetto a quelle con valori bassi. Andando avanti la curva diventa sempre più precisa, oscillando intorno al valore esatto fino a che le oscillazioni sono talmente ampie da alterare completamente l'andamento. Diventa quindi necessario trovare il valore giusto in cui troncare i valori singolari e per fare questo la L-curve si rivela uno strumento molto utile, a seconda del caso non per determinare il punto esatto ma per inquadrare almeno l'intervallo in cui i valori singolari non aggiungono più informazione utile alla soluzione del problema.

Sfruttando ancora la decomposizione a valori singolari è stata poi applicata la routine `tikhonov` che applica l'omonimo metodo e, con lo stesso procedimento descritto per la TSVD, è stato ricavato il grafico della L-curve e il parametro di regolarizzazione ottimale.

Utilizzando `tikhonov` (Figura 3.5), come nel caso della TSVD, si nota un progressivo aumento di precisione della curva approssimante, in questo caso al decrescere del parametro k poiché i valori singolari decrescono all'aumentare dell'indice. Nel punto ottimale calcolato dalla L-curve si nota tuttavia che la componente del rumore sovrasta la precisione ottenuta ed è quindi una soluzione da scartare. La soluzione migliore in questo caso risulta essere la 3.5b, scegliendo un parametro di regolarizzazione minore di un ordine di grandezza rispetto a quello calcolato con il metodo della L-curve.

Figura 3.5: Soluzioni ottenute con `tikhonov`Figura 3.6: L-curve relativa al metodo `tikh`, curva con parametro ottimale in 3.5c

3.2.2 $L = D_n$

Per utilizzare la matrice L anche nei metodi iterativi si è ricorso alla routine `std_form` del toolbox di regolarizzazione che riconduce una forma del tipo

$$\mathbf{x}_\lambda = \arg \min \left\{ \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 + \lambda^2 \|L(\mathbf{x} - \mathbf{x}^*)\|_2^2 \right\}$$

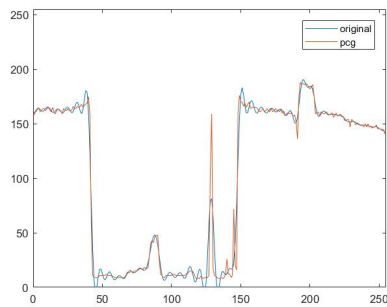
a una del tipo

$$\mathbf{x}_\lambda = \arg \min \{ \|\mathbf{A}\mathbf{x} - \mathbf{b}\|_2^2 + \lambda^2 \|I(\mathbf{x} - \mathbf{x}^*)\|_2^2 \}$$

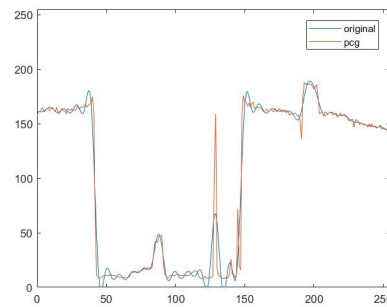
restituendo la matrice A_s , il vettore b_s , la matrice inversa di Penrose L_p e il vettore x_o , le ultime due sono necessarie per ottenere la soluzione di $\mathbf{A}\mathbf{x} = \mathbf{b}$ che è data da

$$x_\lambda = L_p x_s + x_o$$

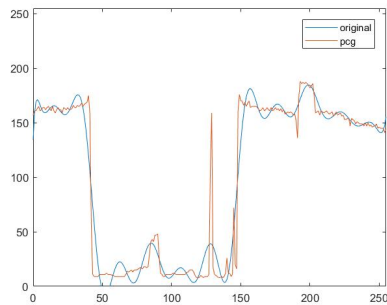
(per approfondire [7]).



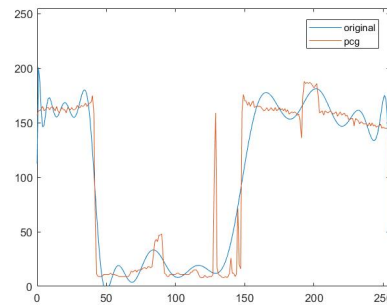
(a) $L = D_1$, convergenza a 225 iterazioni.



(b) $L = D_2$, convergenza a 189 iterazioni



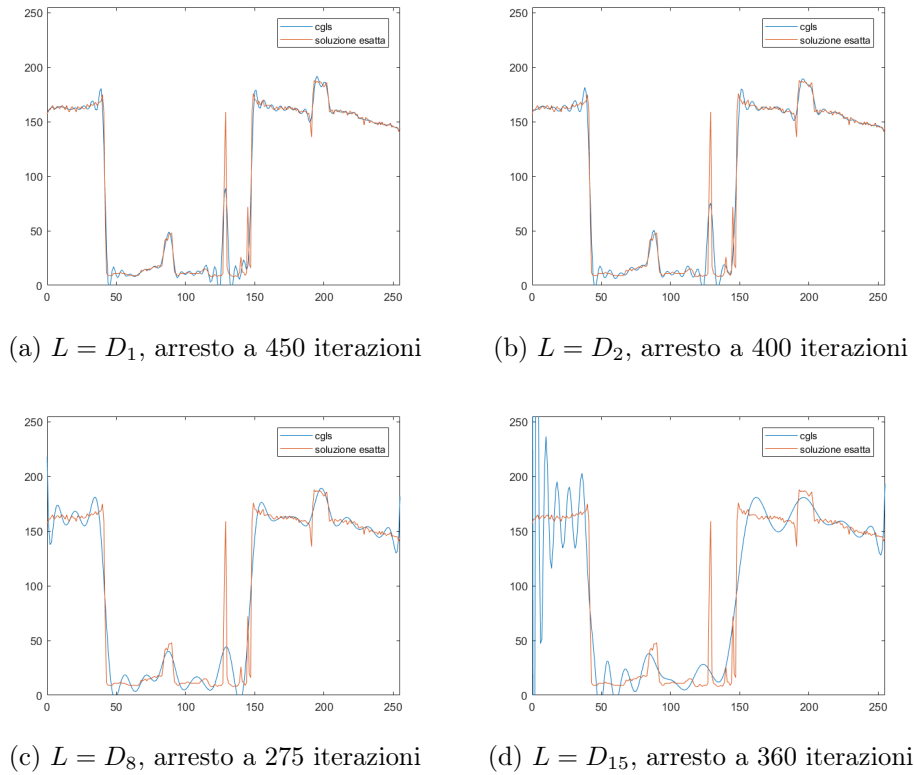
(c) $L = D_8$, convergenza a 135 iterazioni



(d) $L = D_{15}$, convergenza a 19 iterazioni

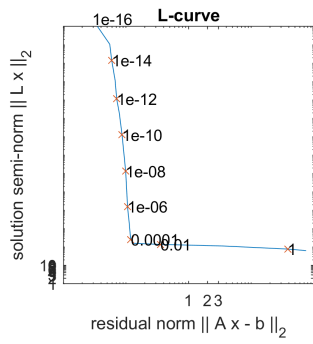
Figura 3.7: Soluzioni ottenute con pcg

Risolvendo il problema ai minimi quadrati con pcg è possibile notare come varia la rapidità di convergenza al variare della matrice L . Assumendo come matrice L un operatore derivata si applica una condizione che rende la soluzione più coerente con la derivata dell'ordine scelto del vettore soluzione. All'aumentare dell'ordine aumenta la velocità di convergenza e, restando in ordini relativamente bassi, si ha anche una soluzione sufficientemente coincidente con quella esatta. Utilizzando invece ordini maggiori i residui aumentano visibilmente e, seppure la convergenza è molto rapida (come nel caso di 3.7d), la soluzione ottenuta non raggiunge una precisione comparabile con 3.7a e 3.7b.

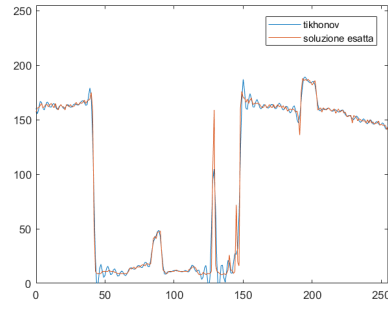
Figura 3.8: Soluzioni ottenute con `cgls`

Utilizzando il metodo del gradiente coniugato implicito `cgls` [8] variando L come con `pcg`, i risultati sono notevolmente diversi. Come prima fatto con nessuna delle matrici di regolarizzazione L utilizzate vi è stata una convergenza con residuo relativo inferiore a 10^{-6} . Il metodo impiega inoltre un numero maggiore di iterazioni anche con nel caso di ordine 15 e con risultati peggiori del corrispondente con `pcg`. Utilizzando il metodo `tikhonov` è stato scelto il parametro ottenuto utilizzando la L-curve relativo a diverse matrici di regolarizzazione. Una importante osservazione riguarda l'andamento delle diverse L-curve al variare della matrice di regolarizzazione: aumentando l'ordine di derivazione la curva diminuisce la convessità che infatti risulta meno marcata per ordini maggiori. I risultati sono comunque di buona precisione con residui relativi sotto l'unità.

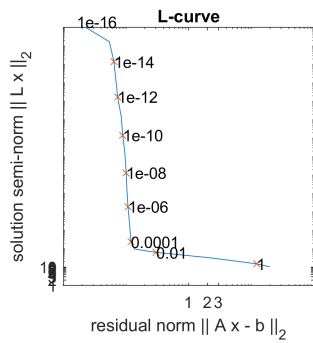
Utilizzando le matrici di regolarizzazione con la routine `tikh` si nota come la convessità della L curve diminuisce gradualmente all'aumentare dell'ordine di D_n aumentando il residuo relativo alla matrice di regolarizzazione. Ciò risulta in soluzioni meno fedeli all'originale, dal punto di vista numerico, che tuttavia mantenendo residui nell'ordine o al di sotto dell'unità, si sovrappongono con una buona precisione alla soluzione esatta, come è possibile notare in Figura 3.9



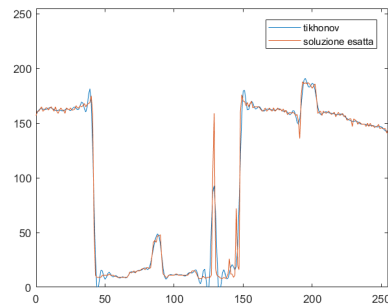
(a) L-curve per $L = D_1$



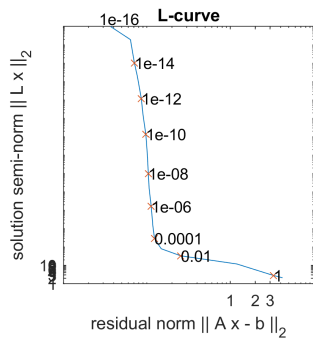
(b) $\rho = 0.1295$



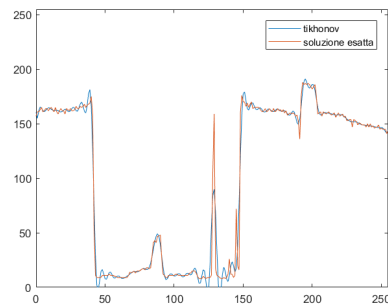
(c) L-curve per $L = D_3$



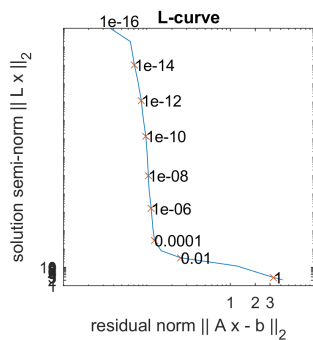
(d) $\rho = 0.1478$



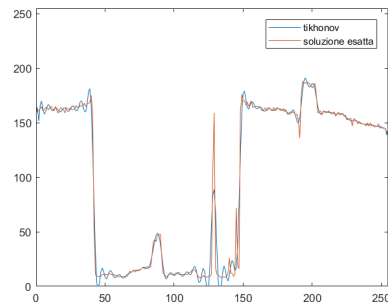
(e) L-curve per $L = D_8$



(f) $\rho = 0.2113$



(g) L-curve per $L = D_{14}$



(h) $\rho = 0.2341$

Figura 3.9: Soluzioni ottenute con tikhonov

3.3 Caso bidimensionale

Nell'applicazione dei metodi precedentemente visti nel caso in due dimensioni sono state utilizzate delle immagini campione con diversi soggetti e diversi livelli di dettaglio in modo da poter osservare i risultati in diverse condizioni. Le immagini di test utilizzate sono fornite dall' Image Processing Toolbox di MATLAB ad eccezione di "Lena". Dai campioni sono stati ritagliati dei frame di dimensione 128×128 in modo da poter gestire in memoria tutte le variabili. La BTTB infatti per le dimensioni 128×128 risulta avere dimensioni $(128 \times 128) \times (128 \times 128) = 16384 \times 16384$. Ne risulta quindi un consumo di memoria elevato (circa 2 GB) che può essere però ridotto utilizzando dei sistemi di gestione di matrici sparse come quello integrato in MATLAB. Le immagini scelte sono un frammento di un'immagine satellitare 3.11, un frammento dell'immagine di test "Lena" 3.12 e uno degli strati di un'immagine di una risonanza magnetica. Il filtro di sfocatura utilizzato è formata dalla BTTB generata a partire di una gaussiana in 2 dimensioni di $\sigma = 3$. Per generare la gaussiana è stata utilizzata la funzione `fspecial` presente nell'Image Processing Toolbox.

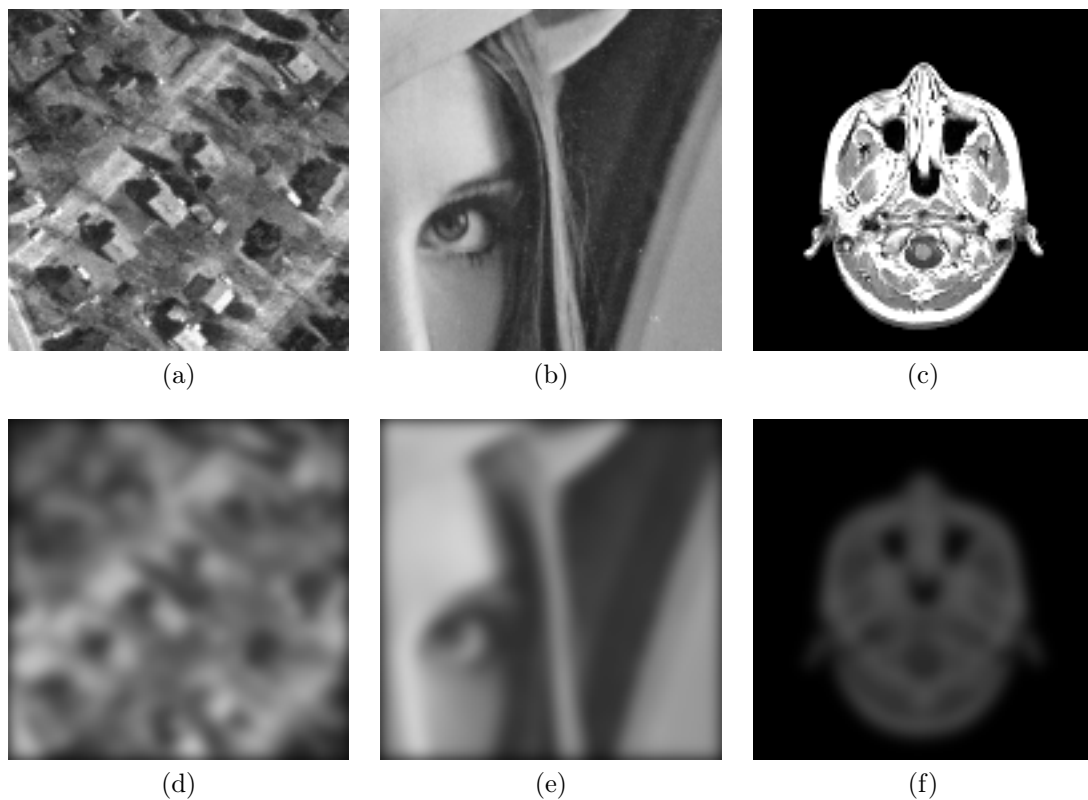


Figura 3.10: Frame delle immagini di test originali e sfocati di dimensione 128×128



Figura 3.11: Immagine satellitare (dimensione originale 2956 x 2215)



Figura 3.12: Immagine di test "Lena" (dimensione originale 512 x 512)

3.3.1 $L = I$

Come per il caso monodimensionale, si sono utilizzati i metodi iterativi `pcg` e `cgl`s e i metodi `tsvd` e `tikh`. Per utilizzare i metodi iterativi si è ricorsi anche in questo caso alla routine `std_form`, alla routine `csvd` per i metodi `tsvd` e `tikh`, alle routine `plot_lc` e `l_corner` per disegnare e ricavare il parametro di regolarizzazione ottimale. I risultati sono rappresentati in Figura 3.13,3.14 e nelle tabelle 3.2,3.3,3.4 e 3.5. Per utilizzare il metodo del gradiente coniugato esplicito, analogamente al caso monodimensionale, si è proceduto calcolando gli elementi del sistema normale $A^T A$ e $A^T \mathbf{b}$ per poi utilizzare `pcg`, il numero di iterazioni cambia sostanzialmente in base all'immagine utilizzata ma in tutti i casi si super il centinaio di iterazioni. Il tempo impiegato per calcolare le matrici $A^T A$ e $A^T \mathbf{b}$ è di circa 46 secondi mentre il tempo impiegato per eseguire le routine sono indicati in tabella 3.2. Utilizzando invece il metodo del gradiente coniugato implicito il numero di iterazioni tra le diverse immagine resta simile intorno alle 550, il tempo impiegato maggiore rispetto a `pcg`, dovuto al prodotto matriciale che viene ripetuto ad ogni passo, residuo relativo maggiore. Con le routine `tsvd` e `tikh` e parametri scelti utilizzando `l_corner` si notano degli artefatti dovuto a rumore in alta frequenza soprattutto nelle immagini 3.14d e 3.14e. L'effetto è analogo a quello del caso monodimensionale in 3.3d. L'assenza di una matrice di regolarizzazione non favorisce il raggiungimento di una soluzione coerente e priva di rumore. È interessante notare come l'effetto si nota solo nell'immagine satellitare e nel ritratto mentre nell'immagine della risonanza magnetica l'effetto è impossibile da notare. I risultati numerici sono descritti nelle tabelle 3.4 e 3.5. È importante notare che seppur i tempi di calcolo delle soluzioni sono molto brevi, è necessario eseguire l'operazione di decomposizione SVD, che in questo caso ha impiegato circa 80 minuti.

	pcg		
	Im. satell.	Lena	MRI
tempo op. (sec)	66.28	42.83	85.04
residuo rel.	9.9546e-07	9.5310-07	9.5384e-07
iterazioni	577	371	743

Tabella 3.2: Tabella riassuntiva `pcg`

	cgls		
	Im. satell.	Lena	MRI
tempo op. (sec)	136.72	130.58	129.34
residuo rel.	2.5286e-04	2.2719e-04	0.0157
iterazioni	583	553	549

Tabella 3.3: Tabella riassuntiva `cgl`s

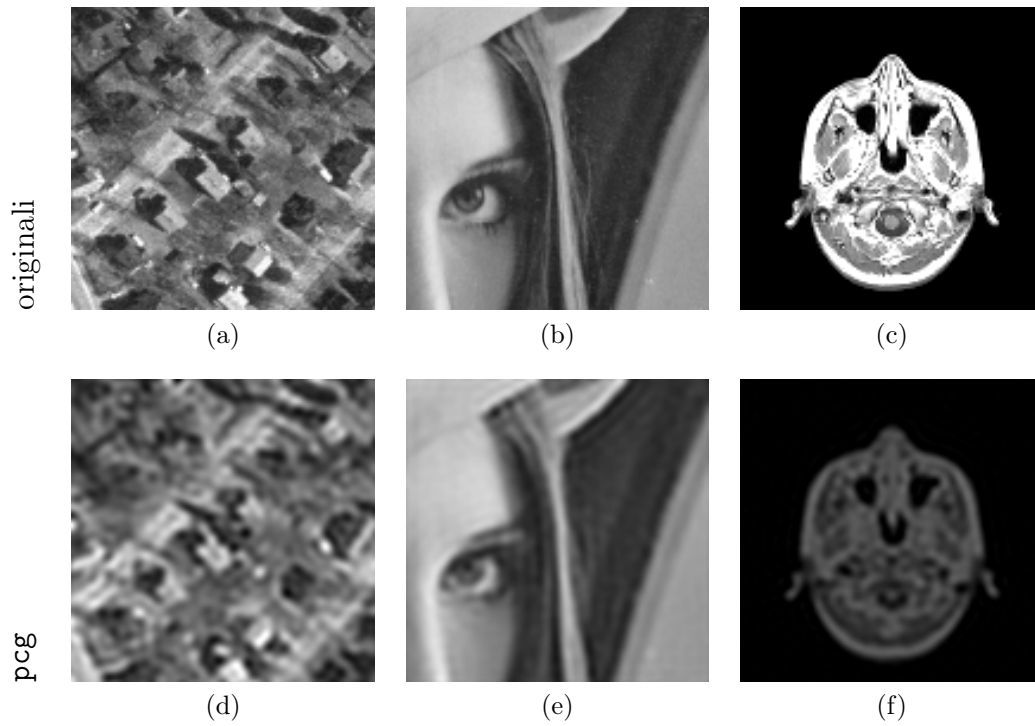


Figura 3.13: Confronto immagini originali con i risultati delle sperimentazioni

	tsvd		
	Im. satell.	Lena	MRI
tempo op. (sec)	2.10	2.13	1.88
residuo rel.	8.1520e-05	7.3715e-05	2.6542e-04
reg. corner	3000	3000	2000
rho	1.1538	1.2232	1.1539
eta	1.5355e+04	16778e+04	5.0223e+03

Tabella 3.4: Tabella riassuntiva tsvd

	tikh		
	Im. satell.	Lena	MRI
tempo op. (sec)	3.63	3.65	3.74
residuo rel.	8.1662e-05	7.3824e-05	2.5350e-04
reg. corner	9.9268e-05	9.9268e-05	2.4371e-04
rho	1.1559	1.1555	1.1683
eta	1.5218e+04	1.6658e+04	5.0557e+03

Tabella 3.5: Tabella riassuntiva tikh

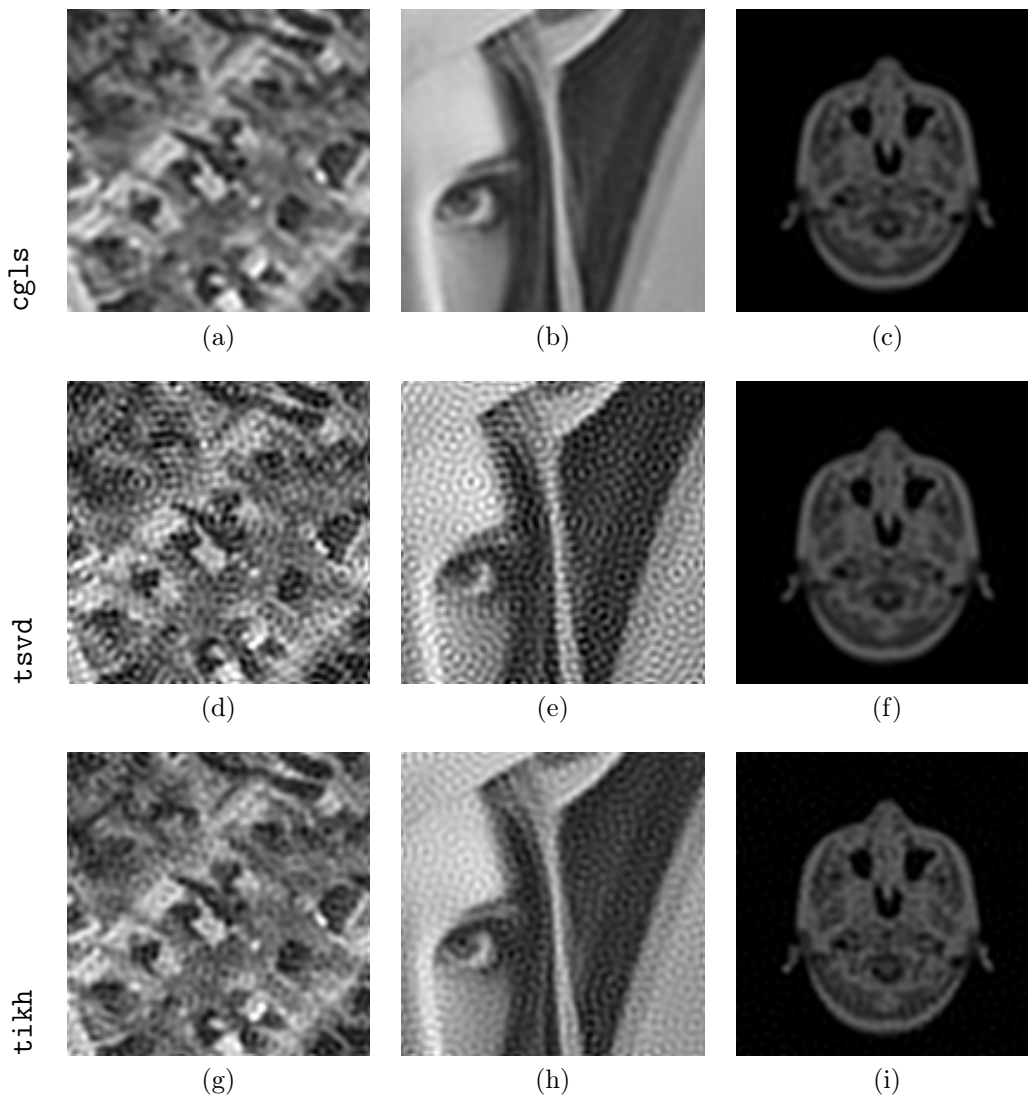


Figura 3.14: Confronto immagini originali con i risultati delle sperimentazioni

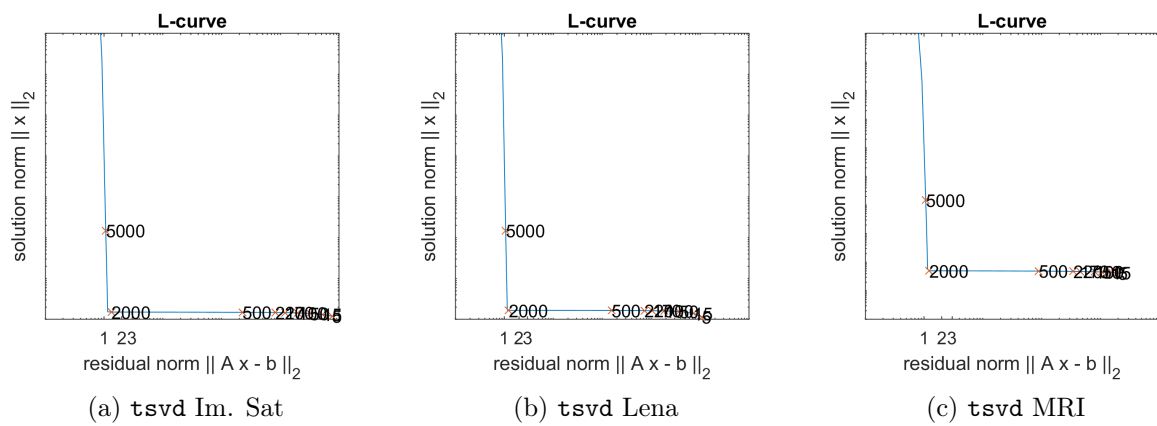


Figura 3.15: L-curve tsvd

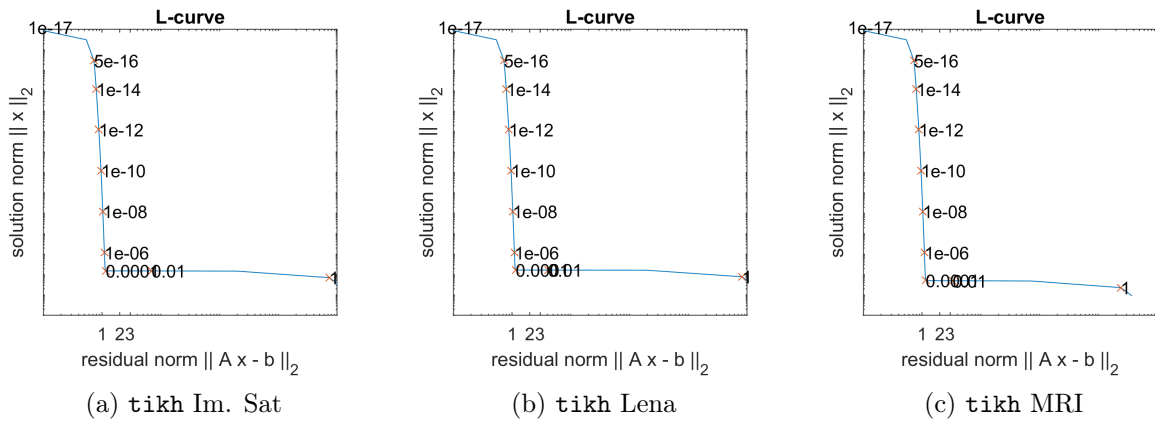


Figura 3.16: L-curve tikh

3.3.2 $L = D_1$

Per l'utilizzo dell'operatore di derivata prima come matrice di regolarizzazione L sono state utilizzate le routine `pcg`, `cgl`, `tikh`. Anche in questo caso per poter utilizzare la matrice di regolarizzazione nei metodi iterativi si è ricorso alla routine `std_form`, operazione che impiega circa 120 secondi in tutti i casi delle immagini. Per il metodo `tikh` si è invece ricorsi alla routine `cgsvd` per ottenere la SVD generalizzata in versione compatta e le funzioni `plot_lc` e `el_corner` per disegnare la L-curve e ricavare il parametro di regolarizzazione ottimale. I risultati sono interessanti e mostrano delle bande verticali, caratteristica dovuta al tipo di matrice di regolarizzazione utilizzata. La derivata infatti agisce in una sola direzione sull'immagine che ricordiamo viene ridotta a una striscia verticale e cioè un vettore colonna. Sono vettori colonna sia \mathbf{b} che \mathbf{x} quindi è inevitabile l'influenza nell'unica direzione verticale della matrice D_1 . Nonostante l'effetto a righe i risultati sono comunque sufficientemente definiti da rendere le figure delle immagini individuabili, in particolare nei risultati del metodo `tikh`.

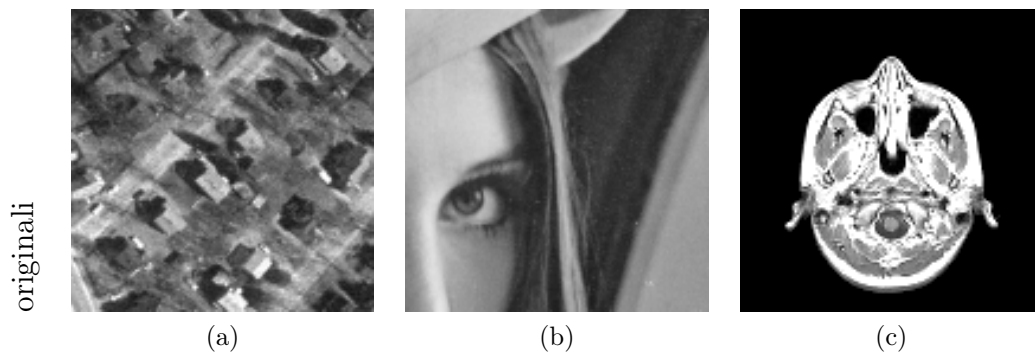


Figura 3.17: Confronto immagini originali con i risultati delle sperimentazioni

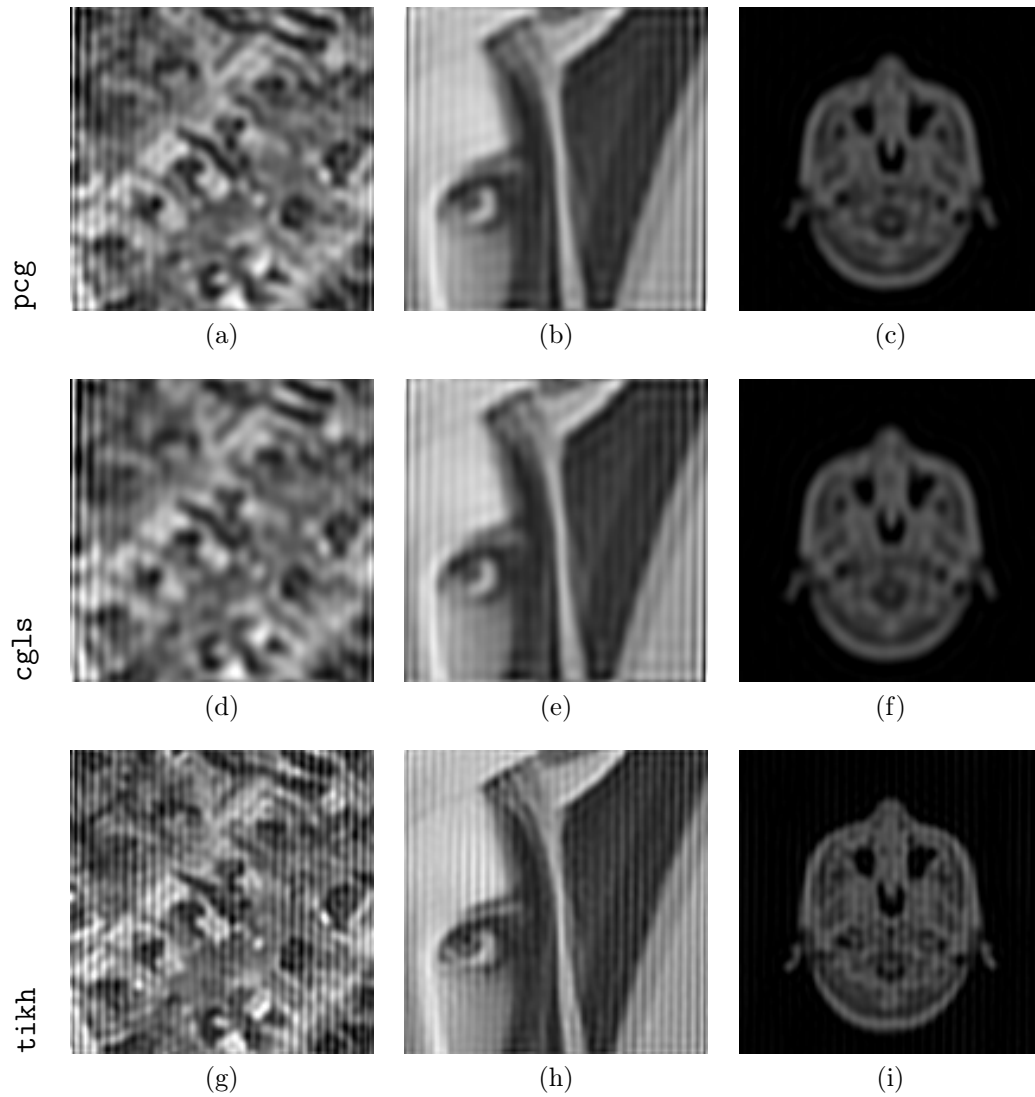


Figura 3.18: Confronto immagini originali con i risultati delle sperimentazioni

	pcg		
	Im. satell.	Lena	MRI
tempo op. (sec)	581.32	274.70	356.81
residuo rel.	5.2493e-06	7.5679e-07	9.4075e-07
iterazioni	4918	2280	3136

Tabella 3.6: Tabella riassuntiva pcg

	cgls		
	Im. satell.	Lena	MRI
tempo op. (sec)	263.05	350.54	291.12
residuo rel.	0.0048	0.002	0.0052
iterazioni	1165	1474	1308

Tabella 3.7: Tabella riassuntiva cgls

	tsvd		
	Im. satell.	Lena	MRI
tempo op. (sec)	7.3	6.56	7.08
residuo rel.	8.4271e-05	7.63e-05	2.6125e-04
reg. corner	5.3088e-04	8.9145e-04	0.0014
rho	1.1929	1.1943	1.2041
eta	2.0499e+03	1.2167e+03	766.4074

Tabella 3.8: Tabella riassuntiva tikh

3.3.3 L = R

Per ovviare al problema riscontrato utilizzando $L = D_1$ la scelta della matrice di regolarizzazione ricade su un operatore che rappresenti il gradiente. L'uso di una matrice rappresentate il gradiente come matrice di regolarizzazione è conosciuto in letteratura come "Total variation", per implementarlo sono stati utilizzate le routine `qr`, che restituisce la fattorizzazione QR della matrice data in ingresso, `kron` che effettua il prodotto di Kronecker.

Definizione 3.1 *Siano $A \in \mathbb{R}^{m \times n}$ e $B \in \mathbb{R}^{p \times q}$, è definito $A \otimes B$ come prodotto di Kronecker il cui risultato è una matrice definita in $\mathbb{R}^{mp \times nq}$*

$$A \otimes B = \begin{bmatrix} a_{11} B & \cdot & \cdot & \cdot & a_{1n} B \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ a_{m1} B & \cdot & \cdot & \cdot & a_{mn} B \end{bmatrix}$$

Per rappresentare il gradiente viene generata una matrice composta da due blocchi, che rappresentano rispettivamente le derivate lungo i due assi.

$$L = \begin{bmatrix} I_N \otimes D_1 \\ D_1 \otimes I_N \end{bmatrix} \in \mathbb{R}^{(2(n-1)N) \times (nN)}$$

e ne risulta

$$L = \begin{bmatrix} D_z \\ D_x \end{bmatrix}$$

che nel nostro caso ha dimensioni 32512×16384 . Le dimensioni sono evidentemente incompatibili con quelle del vettore \mathbf{x} (16384×1) per poterlo utilizzare nei metodi usati precedentemente. Si è quindi ricorsi alla fattorizzazione QR compatta, che restituisce solo le righe non nulle della matrice ortogonale Q e della triangolare superiore R. La triangolare superiore R risulta avere dimensioni 16384×16384 ed è quindi possibile utilizzarla per premoltiplicare il vettore \mathbf{x} nel metodo di Tikhonov. Dati i numerosi passaggi e approssimazioni eseguite per poter utilizzare il gradiente come matrice di regolarizzazione non è stato possibile ricondurre il sistema alla forma standard con conseguente impossibilità di utilizzo delle routine `pcg` e `cglsl`. I risultati del metodo `tikh` sono rappresentati in Figura 3.19, in tabella 3.9.

	tikh		
	Im. satell.	Lena	MRI
tempo op. (sec)	4.3426	3.9912	4.1617
residuo rel.	8.4643e-05	7.6570e-05	2.6191e-04
reg. corner	4.0764e-04	7.2179e-04	9.2303e-04
rho	1.1981	1.1985	1.2071
eta	2.5738e+03	1.5147e+03	1.1824e+03

Tabella 3.9: Tabella riassuntiva `tikh`

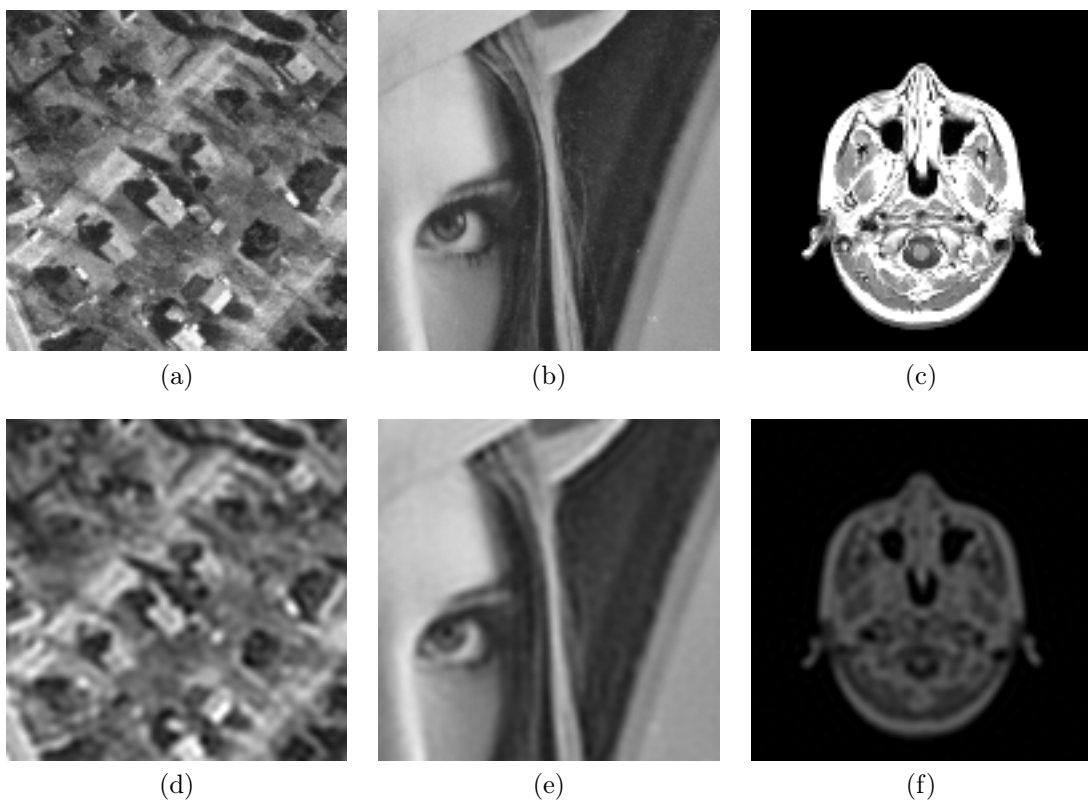


Figura 3.19: Confronto immagini originali con risultati tikh

Capitolo 4

Conclusioni

I metodi utilizzati per affrontare i problemi monodimensionale e bidimensionale si sono rivelati tutti validi, in circostanze differenti. Nel caso monodimensionale i risultati ottenuti con i metodi iterativi sono quelli più precisi nel caso di $L = I$, nel caso di utilizzo di matrici di regolarizzazione i metodi Tikhonov e TSVD si rivelano all'altezza, se non migliori in alcuni casi, con l'ausilio della L-curve per la scelta del parametro di regolarizzazione. Nel caso bidimensionale i metodi iterativi si rivelano i più rapidi a fornire una soluzione con pochi passaggi intermedi mentre nei metodi di Tikhonov e TSVD il problema maggiore è dato dal tempo di calcolo delle decomposizioni a valori singolari. Tuttavia supponendo di conoscere la PSF e di avere quindi la decomposizione SVD di essa già pronta, l'approccio al problema si rivela decisamente più rapido utilizzando questi due metodi. Un lato negativo, o meglio una necessità, da rimarcare è la quantità di memoria necessaria per elaborare immagini anche di dimensioni relativamente piccole come quelle utilizzate in questa tesi. Tuttavia considerando la presenza di valori molto prossimi a zero, al di sotto della soglia di sensibilità del sistema, le matrici ottenute possono essere considerate sparse in e aprendo la possibilità di una gestione delle matrici sparse ottimizzate anche in routine come `svd` si abbatterebbero notevolmente i requisiti di memoria.

Bibliografia

- [1] G. Rodriguez - S. Seatzu, *Introduzione alla Matematica Applicata e Computazione*, Pitagora Editrice, Bologna, 2010.
- [2] A. N. Tikhonov & V. Y. Arsenin, *Solutions of Ill-Posed Problems*, Winston & Sons, Washington, D.C., 1977.
- [3] G. Rodriguez, *Algoritmi Numerici*, Pitagora Editrice, Bologna, 2008.
- [4] P. C. Hansen, *Regularization, GSVD and truncated GSVD*, BIT 29 (1989), 491–504.
- [5] P. C. Hansen, *Computation of the singular value expansion*, Computing 40 1988, 185–199
- [6] K. Miller, *Least squares methods for ill-posed problems with a prescribed bound*, SIAM J. Math. Anal. 1 (1970), 52–74.
- [7] A. Bjorck, *A bidiagonalization algorithm for solving large and sparse ill-posed systems of linear equations*, BIT 28, (1988), 659–670.
- [8] M. Hanke, *Conjugate Gradient Methods for Ill-Posed Problems*, Longman Scientific and Technical, Essex, 1995.
- [9] P. C. Hansen, *Regularization Tools Version 4.0 for Matlab 7.3*, Numerical Algorithms, 46 (2007), pp. 189-194. Regularization Tools Version 4.1 (for MATLAB Version 7.3) <http://www.imm.dtu.dk/~pcha/Regutools/>