POLITECNICO DI TORINO

Facoltà di Ingegneria dell'Informazione
Corso di Laurea in Ingegneria delle Telecomunicazioni

Codifica di immagini e video

Tesina sulle esperienze di laboratorio

Prof. Gabriella Olmo



Studenti:
Simone Franchi
Simone Geroso

Anno accademico 2001-2002

Indice

1 DWT, applicazione al segnale vocale e segnali audio personali **1**
1.1 Introduzione 1
1.2 Richiami Teorici 1
1.3 Prima fase sperimentale 9
1.4 Risposte all'impulso 20
1.5 Seconda fase sperimentale 24

2 Applicazione della DWT a immagini **33**
2.1 Introduzione 33
2.2 Prima fase sperimentale 35
2.3 Seconda fase sperimentale 65
2.3.1 Scomposizione in sottobande con 1, 2 e livelli 66
2.3.2 Valutazione contenuto energetico 71
2.3.3 Ricostruzione con azzeramento di sottobande 74
2.3.4 Valutazione del MSE e del PSNR 77
2.3.5 Scambio dei banchi e bordi con estensione simmetrica . 79

3 Compressione di immagini mediante la DWT **82**
3.1 Introduzione 82
3.2 DWT su immagini 83
3.3 Descrizione dell'esperienza e risultati ottenuti 85
3.3.1 Compressione mediante DWT con estensione simmetrica dei bordi 87
3.3.2 Compressione mediante DWT con estensione periodica dei bordi 97
3.3.3 Alcune prove di compressione 101
3.3.4 I difetti della DWT 104

4 Jpeg vs Jpeg2000 **111**
4.1 Introduzione 111

4.2 Jpeg2000 112
4.2.1 Schema generale 112
4.2.2 Il sistema di coordinate 114
4.2.3 La trasformata wavelets 115
4.2.4 La quantizzazione 116
4.2.5 La codifica entropica: EBCOT 116
4.2.6 Codifica a bit-plane dei blocchi 118
4.2.7 Codifica con regioni d'interesse 118
4.3 Descrizione dell'esperienza e risultati ottenuti 118
4.3.1 JPEG 120
4.3.2 JPEG2000 137
4.3.3 Jpeg e Jpeg2000 a confronto 152
4.3.4 Robustezza agli errori (Error Resilience) 154
4.3.5 Prove di compressione lossless 164
4.3.6 Scalabità in risoluzione e SNR 165
4.3.7 I Tiles 168
4.3.8 Regioni di interesse 169
4.3.9 Conclusioni 172
4.3.10 Note sul software 173

Capitolo 1

DWT, applicazione al segnale vocale e segnali audio personali

1.1 Introduzione

In questo capitolo saranno affrontate varie sperimentazioni volte a conoscere la DWT, la Trasformata Wavelet Discreta, applicata a segnali monodimensionali.

In una prima fase saranno usati segnali creati artificialmente mediante Matlab; questi segnali, manipolati attraverso la DWT, ci permetteranno di capire come funzione la metodologia di analisi delle wavelet e quindi allo stesso tempo individuare le caratteristiche dei segnali stessi.

Nascerà a questo punto la necessità di scoprire cosa sia la wavelet, o in italiano "ondina", e il concetto di mother wavelet, "ondina madre".

Seguirà un'applicazione analoga alla precedente mirata all'esame dell'influenza e alle potenzialità della DWT su segnali audio veri e propri.

1.2 Richiami Teorici

Uscendo dal dominio dei segnali stazionari, di tipo sinusoidale, consideriamo quelli che hanno caratteristiche variabili nel tempo. L'analisi in frequenza, nata dalla definizione della trasformata di Fourier, abbandonando completamente il dominio del tempo non è adatta a far risaltare le caratteristiche del segnale non stazionario. Come soluzione a questo problema, come sappiamo, può essere considerata l'analisi tempo-frequenza che utilizza la Trasformata di Fourier a breve termine (STFT), ovvero la trasformata di Fourier al segnale limitato da una finestra temporale. Iterando questo procedimento per

intervalli di tempo successivi è possibile quindi mantenere una caratterizzazione temporale dell'analisi del segnale. Di seguito riportiamo le formulazioni delle due metodologie di analisi enunciate evidenziando la loro interpretazione come prodotto scalare

$$X(f) = \int_{-\infty}^{+\infty} x(t)\, e^{-j2\pi ft} dt = \left\langle e^{-j2\pi ft}, x(t) \right\rangle \tag{1.1}$$

$$STFT_x(\tau, f) = \int_{-\infty}^{+\infty} x(t)\, g^*(t-\tau)\, e^{-j2\pi ft} dt = \left\langle g(t-\tau)\, e^{-j2\pi ft}, x(t) \right\rangle \tag{1.2}$$

in questo modo la traformata di Fourier può essere vista, più esplicitamente, come scomposizione del segnale $x(t)$ in sinusoidi, che costituiscono la base dello spazio di funzioni, rispetto al quale il nostro segnale viene scomposto (armoniche del segnale), ma ciò è particolarmente adatto a segnali stazionari in generale. Nel caso della STFT il segnale è assunto stazionario nell'intervallo di tempo limitato dalla finestra $g(t)$ e ciò sembrerebbe porti alla soluzione dei problemi, ma la finestra $g(t)$ di ampiezza Δt ha, in corrispondenza, una determinata larghezza di banda Δf costante che limita in range di frequenza del segnale cosicché ci ritroviamo con l'indeterminazione tempo-frequenza che limita di certo la possibilità di risoluzione della STFT.

A questo punto l'idea è quella di individuare delle basi dello spazio di funzioni diverse dalle sinusoidi, delle basi strutturate, ovvero che siano locali sia nel tempo che nella frequenza e semplici da ottenere (mediante traslazioni, modulazioni, scalamenti) in modo tale da esprimere il segnale, oggetto ad analisi, come una serie di coefficienti che indichino la somiglianza del segnale con la singola funzione della base.

La funzione che consideriamo è la seguente

$$\psi_{ab} = \frac{1}{\sqrt{|a|}} \psi\left(\frac{t-b}{a}\right) \tag{1.3}$$

Tale funzione, che chiameremo *MotherWavelet*, definirà, sostituita nelle espressioni precedenti quella che viene detta *Trasformata Wavelet Continua (CWT)*

$$CWT_x(a,b) = \int_{-\infty}^{+\infty} x(t)\, \psi\left(\frac{t-b}{a}\right) \tag{1.4}$$

le wavelet saranno le componenti della base che considereremo, ognuna con valori di a e b differenti, in modo che ognuna abbia occupazione temporale e/o frequenza diversa. Si può notare che il dominio della CWT non è più

né la frequenza né il tempo ma l'ampiezza della scala dell'ondina e quindi in corrispondenza dei valori di scala ci saranno le "componenti", o se vogliamo in analogia con l'analisi in frequenza le "armoniche", del segnale somiglianti alla wavelet con quella ampiezza di scala e traslata opportunatamente.

Il passaggio dalla CWT alla versione discretizzata non è molto difficile da concepire una volta affrontato lo stesso passaggio con la trasformata di Fourier; infatti, oltre a considerare il segnale $x[n]$ a tempo discreto, bisognerà discretizzare i valori di a e di b cosicché l'espressione generica della wavelet diventa

$$\psi_{a_m b_n} = \frac{1}{\sqrt{|a_m|}} \psi\left(\frac{n - b_n}{a_m}\right) \tag{1.5}$$

Discretizzando a e b possiamo trovarci in tre diverse situazioni:

- si ottiene una rappresentazione ridondante perché le wavelet considerate hanno dipendenze lineari reciproche, in altre parole a e b potrebbero essere campionati in modo più rado;
- si ottiene una rappresentazione non ridondante perché le wavelet sono linearmente indipendenti, e quindi sono tutte indispensabili per la rappresentazione;
- si ottiene una rappresentazione non completa perché a e b sono campionati in modo troppo rado;

Senza andare nei dettagli un ragionamento intuitivo ci porta a definire un criterio di discretizzazione che permette, in funzione della necessità, d'avere rappresentazioni ridondanti o meno. In particolare ci porta ad affermare che il passo di traslazione temporale b_m deve dipendere dal valore di scala a_n ed è ragionevole compiere una discretizzazione logaritmica del parametro di scala. Avremo quindi

$$\begin{array}{lll} a_m = a_0^m & m \in Z & a_0 > 1 \\ b_{m,n} = n \cdot b_0 \cdot a_0^m & n \in Z & b_0 > 0 \end{array} \tag{1.6}$$

Considerando $a_0 = 2$ si ottiene quella che viene chiamata *discretizzazione diadica*.

Uno sviluppo in serie di x appartenente ad uno spazio di funzioni S (a tempo continuo o discreto) significa trovare un insieme di funzioni $\{\phi_i\}_{i \in Z} \in S$ tali per cui sia possibile esprimere x come combinazione lineare di tali funzioni:

$$x = \sum_i \alpha_i \phi_i \tag{1.7}$$

Tale insieme di funzioni ha delle proprietà:

- $\{\phi_i\}$ si dice completo per S se vale la 1.7 $\forall x \in S$;
- Se $\{\phi_i\}_{i\in Z}$ è un insieme completo per S, allora esisterà anche un insieme $\left\{\widetilde{\phi}_i\right\}_{i\in Z} \in S$ detto *duale* rispetto a $\{\phi_i\}_{i\in Z}$ e non necessariamente distinto da esso, tale per cui

$$\alpha_i = \left\langle \widetilde{\phi}_i \, , x \right\rangle \tag{1.8}$$

  la natura dell'operazione di prodotto scalare dipende dall'insieme S;
- Se la famiglia $\{\phi_i\}_{i\in Z}$, oltre che rappresentare un insieme completo per S, è anche costituita da funzioni ortonormali, ovvero $\langle \phi_i, \phi_j \rangle = \delta\left[i-j\right]$, allora essa costituisce una base ortonormale per S; valgono le relazioni

$$\begin{array}{l} \left\{\widetilde{\phi}_i\right\}_{i\in Z} = \{\phi_i\}_{i\in Z} \\ x = \sum\limits_i \langle \phi_i, x \rangle \, \phi_i \end{array} \tag{1.9}$$

- Se $\{\phi_i\}_{i\in Z}$ è completo per S ed è formato da funzioni linearmente indipendenti, ma non ortonormali, allora esso rappresenta, insieme al suo duale $\left\{\widetilde{\phi}_i\right\}_{i\in Z}$, una base *biortogonale;*
- Se $\{\phi_i\}_{i\in Z}$ è completo per S, ma i suoi elementi non sono tra loro linearmente indipendenti, la rappresentazione di $x \in S$ in termini di questa famiglia di funzioni è ridondante.

A questo punto per effettuare l'analisi del segnale mediante le wavelet, serve scegliere la mother wavelet e da essa le wavelet secondo le indicazioni teoriche indicate.

Attraverso la cosiddetta base di Haar, che è ortonormale e completa per tutte le sequenze a quadrato sommabile, i sottocampionatori e i sovracampionatori, questi ultimi che sono elementi che variano la distanza temporale tra due campioni adiacenti e quindi come se cambiassero la frequenza di campionamento del segnale continuo (relazioni del *MSP, Multirate Signal Processing*), si è arrivati alla definizione di un banco di filtri di analisi che scompone il segnale di partenza in due sequenze che hanno frequenza di campionamento pari alla metà di quella della sequenza $x[n]$

E' possibile verificare che la sequenza $y0$ sono una versione filtrata passabasso e sottocampionata del segnale di partenza e quindi il suo contenuto è di componenti a bassa frequenza del segnale (versione approssimata del

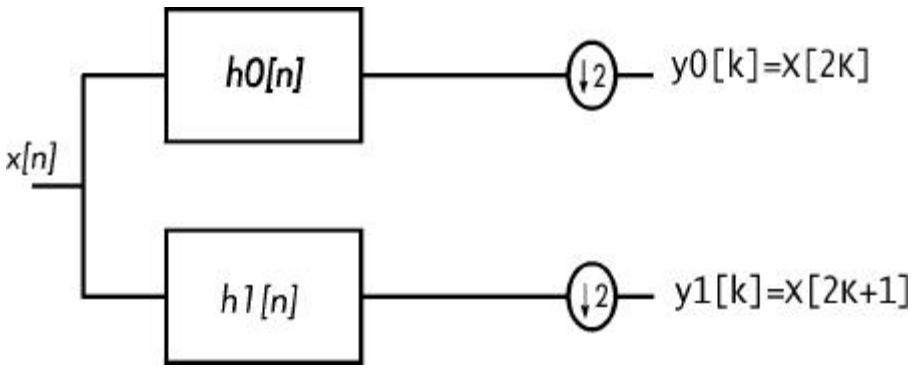


Figura 1.1. Banco di filtri di analisi

segnale); mentre la sequenza $y1$ porta informazioni sulle componenti di alta frequenza (informazioni di dettaglio).

Con un opportuno procedimento di ricostruzione è possibile definire il relativo banco di sintesi La definizione dei quattro filtri h0, h1, g0 e g1 presenti

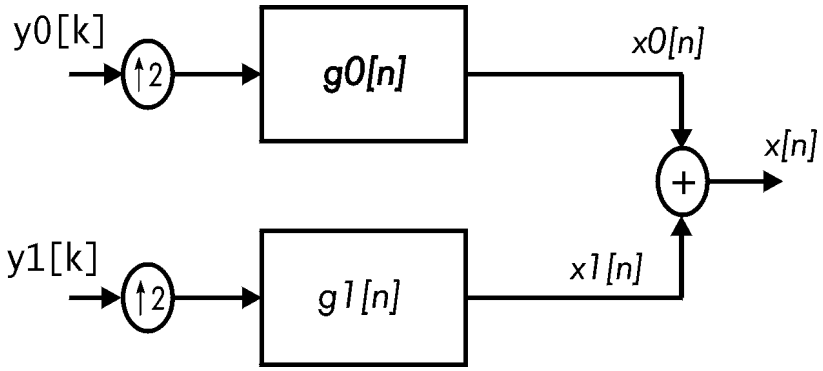


Figura 1.2. Banco di filtri di sintesi

nei due banchi porta alla definizione delle basi strutturate, di cui si parlava in precedenza, queste ultime possono essere ortonormali o biortogonali.

Per arrivare alla definizione di banchi a perfetta ricostruzione (PR), in modo tale da verificare

$$\hat{x}\,[n] = x\,[n] \tag{1.10}$$

si può effettuare un'analisi nel dominio della frequenza e tramite passaggi

matematici e sfruttando anche le relazioni del MRP si arriva alle condizioni

$$H_0(z)\,G_0(z) + H_1(z)\,G_1(z) = 2 \tag{1.11}$$

$$H_0(-z)\,G_0(z) + H_1(-z)\,G_1(z) = 0 \tag{1.12}$$

dove la 1.11 è una *condizione di assenza di distorsione*, mentre la 1.12 è una *condizione di assenza di aliasing*. L'aliasing è un tipo d'errore molto indesiderato e per questa ragione tale condizione, anche nella pratica, tende ad essere soddisfatta esattamente; l'altra condizione essendo invece assai restrittiva, nella pratica si accetta almeno la presenza di un ritardo tra $\hat{x}[n]$ e $x[n]$.

Nel progetto di banchi di filtri ortonormali dovremo aggiungere la condizione

$$g_i[n] = h_i[-n] \tag{1.13}$$

diversamente progetteremo banchi di filtri PR biortogonali.

A questo punto iterando N volte il banco di analisi sul ramo passa-basso otteniamo la cosiddetta codifica a sottobande Ogni stadio opera sulla se-

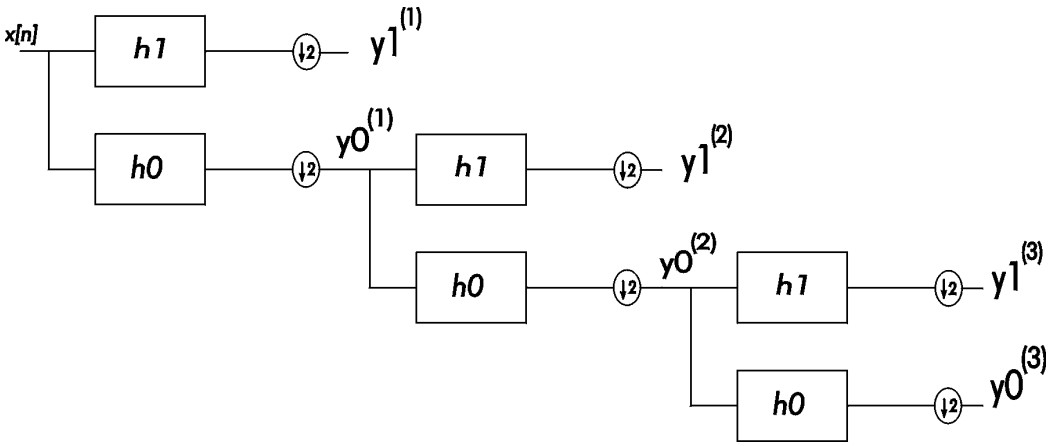


Figura 1.3. Iterazione del banco di filtri di analisi. Codifica a sottobande

quenza d'ingresso $y_0^{(i)}[k]$ la medesima scomposizione operata dal primo stadio su $x[n]$. La banda di $y_0^{(i)}[k]$ (corrispondente alla metà bassa della banda di $y_0^{(i-1)}[k]$) viene ulteriormente divisa in due parti, bassa e alta, e gli spettri delle due sequenze così ottenute, $yl^{(i+1)}[k]$e $yh^{(i+1)}[k]$, vengono riportati nel periodo fondamentale $[-0.5,0.5]$ dai sottocampionatori, che forniscono le sequenze $y_0^{(i+1)}[k]$ e $y_1^{(i+1)}[k]$ a risoluzione ulteriormente dimezzata e scala ulteriormente raddoppiata. Per tale ragione si parla di codifica a sottobande e

siccome le larghezze di banda delle sequenze sono successivamente ridotte di un fattore 2, la scomposizione viene anche detta a ottave.

Iterando N volte il banco di analisi sul ramo passa basso otteniamo l'insieme delle sequenze

$$\left\{y_1^{(1)}[k], y_1^{(2)}[k], \cdots, y_1^{(j)}[k], \cdots, y_1^{(N)}[k], y_0^{(N)}[k]\right\} \tag{1.14}$$

che rappresentano i coefficienti di uno sviluppo in serie di $x[n]$. Il pedice 1 indica uscita del ramo passa-alto, il pedice 0 l'uscita del ramo passa-basso mentre l'apice il livello di scomposizione. Questi coefficienti sono organizzati in sequenze ciascuna con risoluzione dimezzata e scala doppia della precedente e porta informazione su frazioni diverse di banda, localizzate a frequenze sempre più basse e di larghezza decrescente al crescere del numero dello stadio; infatti, si parla di scomposizione a banda relativa costante.

Quindi la codifica a sottobande compie una scomposizione in termini di funzioni a scale diverse, e i valori di scala sono diadici (del tipo 2^j, con j pari al generico livello della scomposizione) e per questo motivo lo schema di codifica a sottobande viene identificato con lo sviluppo in *serie wavelet diadico per segnali a tempo discreto (DT-DWS).*

Per la ricostruzione PR del segnale si usa un banco iterato di sintesi costruito analogamente a quello di analisi. Tutte le sequenze dei coefficienti 1.14 si possono ottenere sottoponendo $x[n]$ a un filtraggio con filtro H1 sovracampionato e a un opportuna cascata di sottocampionatori, ovvero un sottocampionatore di 2^j dove j è il livello. In modo analogo si può avere il

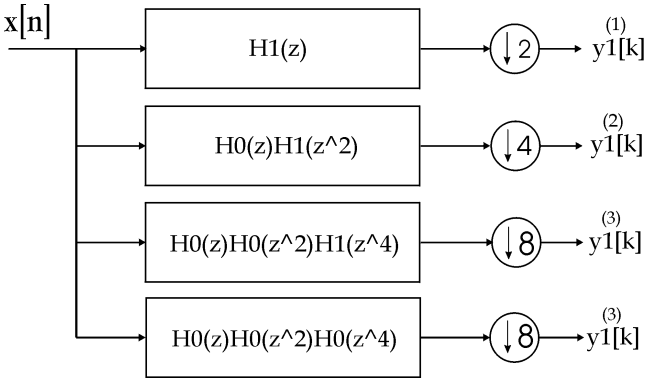


Figura 1.4. Schema equivalente al banco di analisi iterato

banco di sintesi.

Le sequenze ricostruite dal banco di sintesi

$$\left\{x_1^{(1)}[k], x_1^{(2)}[k], \cdots, x_1^{(N)}[k], x_0^{(N)}[k]\right\} \tag{1.15}$$

sono versioni del segnale a risoluzione differente ma alla stessa scala, e costituiscono quella che è chiamata *Analisi Multirisoluzione (MRA).*

L'analisi multirisoluzione scompone il segnale in *spazi approssimazione* concentrici

$$\cdots V_2 \subset V_1 \subset V_0 \subset V_{-1} \subset \cdots$$

Data una base $\{\phi(t-n)\}_{n\in Z}$ di uno di questi spazi V_i (detta *funzione di scalamento o father wavelet*) si individuano le basi degli altri spazi come basi strutturate $\left\{2^{\frac{-m}{2}}\phi\left(2^{-m}t-n\right)\right\}_{n\in Z}$ base di V_m.

Allo stesso tempo in una MRA si definiscono *spazi dettaglio* W_i ognuno dei quali costituisce il completamento ortogonale di V_i in V_{i-1}

$$V_{i-1} = V_i \oplus W_i \tag{1.16}$$

Quindi tutte le funzioni in V_i sono ortogonali a tutte le funzioni di W_i, e qualsiasi funzione in V_{i-1} si può costruire sommando le proiezioni in V_i e in W_i.

Iterando il procedimento della codifica a sottobande otteniamo

$$V_{i-1} = W_i \oplus V_i = W_i \oplus W_{i+1} \oplus V_{i+1} = W_i \oplus W_{i+1} \oplus W_{i+2} \oplus V_{i+2} = \cdots$$

$$V_{i-1} = \bigoplus_{j=1}^{\infty} W_j \tag{1.17}$$

Come fatto in precedenza, definita una base $\{\psi(t-n)\}$ per un sottospazio W_i (detta *mother wavelet*) è possibile definire le basi degli altri spazi come basi strutturate di quest'ultima ondina $\left\{2^{\frac{-m}{2}}\psi\left(2^{-m}t-n\right)\right\}_{n\in Z}$. E' possibile verificare la proprietà secondo la quale

$$\psi(t) = \sqrt{2}\sum_{n=-\infty}^{+\infty} g_1[n]\,\phi(2t-n) \tag{1.18}$$

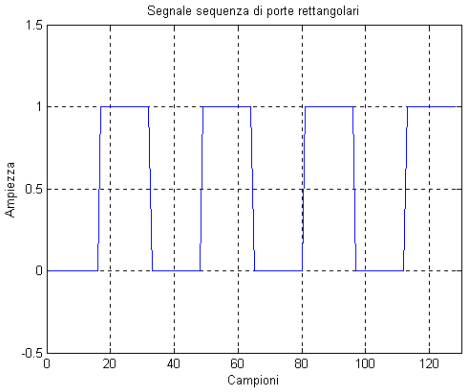
detta *equazione a due scale per l'ondina.* Se il segnale è ad energia finita la scomposizione porta ad avere un certo numero di sequenze di dettaglio del segnale (pari ai livelli di scomposizione) per concludersi con l'ultima versione approssimata che potrebbe avere contenuto energetico nullo.

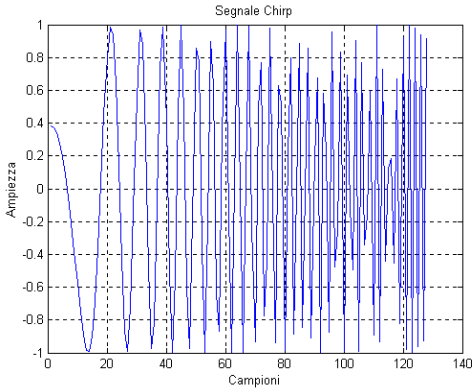
1.3 Prima fase sperimentale

In questi primi esperimenti abbiamo considerato due segnali particolari come una sequenza di porte e un chirp, costruiti in modo adeguato mediante Matlab. Con questi segnali abbiamo esaminato le uscite del banco di analisi

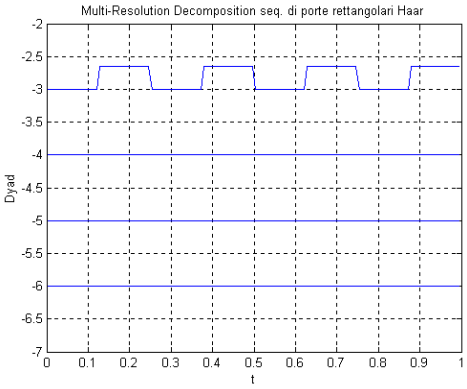
e del banco di sintesi, utilizzando sia filtri ortogonali, con ondina di Haar e Daubechies (10), sia filtri biortogonali con ondina CDF e Villasenor. In particolare, si è verificata anche la perfetta ricostruzione della procedura mantenendo tutta l'informazione contenuta nelle varie sottobande ad alta frequenza, e si è voluto vedere l'effetto dell'azzeramento di parte di tale informazione. Infine si è voluto fare una verifica del contenuto energetico delle varie sottobande, calcolando l'energia di ognuna di esse e facendo vari confronti.

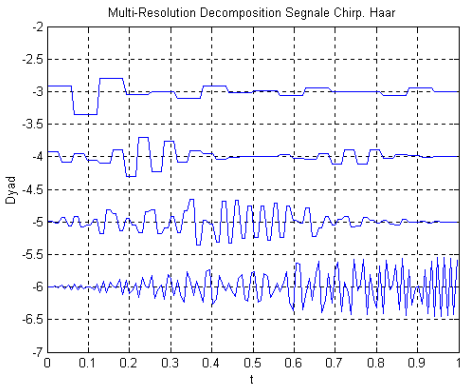
Consideriamo inizialmente filtri ortogonali e i due segnali suddetti costruiti su un numero di campioni $NFIN = 128$.



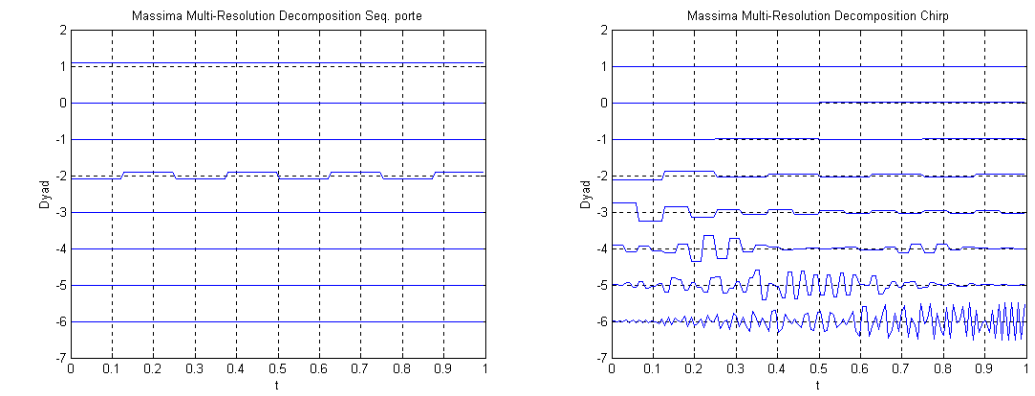


Mediante il comando Matlab FWT_PO (Traformata Wavelet Diretta Periodicizzata Ortogonale) abbiamo realizzato un banco di analisi inizialmente con un numero di livelli $k = 3$; tale funzione deve avere come parametri d'ingresso oltre al segnale $x[n]$, il numero di livelli $L = \log_2(n) - k$, e il vettore coefficienti del filtro scelto qmf (quadrature mirror filter), che è una possibile soluzione che soddisfa 1.11 e 1.12, precedentemente calcolata mediante col comando MakeOnFilter, che richiede due parametri: il tipo di filtro ed un parametro che è correlato con i momenti nulli della wavelet. Di seguito vi sono le uscite del banco di analisi costruito con ondina di Haar.



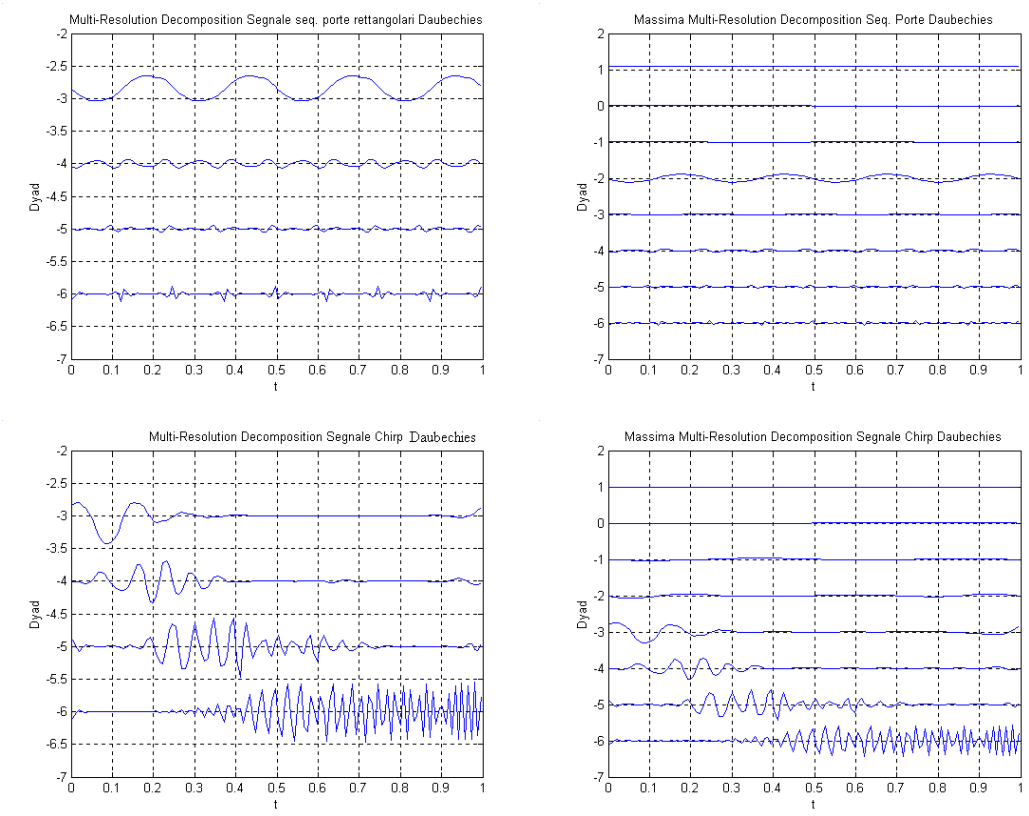


In questi grafici in ordinata ritroviamo il numero d'ordine del livello di scomposizione, nella riga superiore il segnale approssimato dalla funzione di scalamento, nelle righe inferiori le varie componenti di dettaglio di contenuto in frequenza sempre più elevato con l'aumentare dei livelli di scomposizione. E' da notare che il numero di livello riportato in ordinata fa riferimento alla massima scomposizione possibile,nel nostro caso con $NFIN = 128$ $k = 7$ di cui riportiamo di seguito l'uscita del relativo banco di analisi.



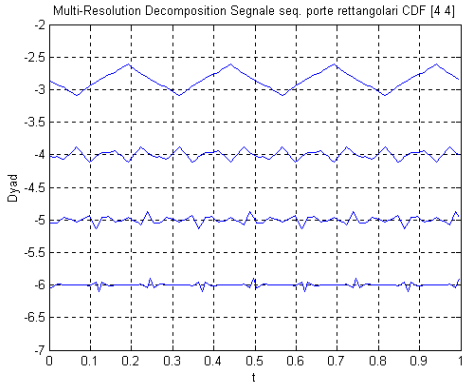
Nel conteggio dei livelli di scomposizione solitamente non si parte da 0 e per tale ragione non ritroviamo 7 in corrispondenza del livello di contenuto in frequenza più elevato (H7, quello più in basso), mentre ritroviamo corrispondenza nella notazione del numero d'ordine che associamo al livello di approssimazione della segnale (L1, componente superiore).

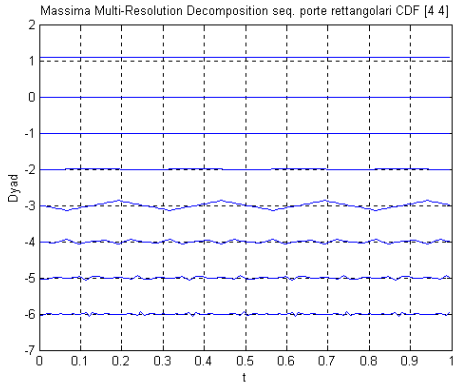
Dal punto di vista delle uscite relativamente al segnale che abbiamo considerato, la progressione del contenuto in frequenza dei vari livelli di dettaglio è ben visibile con il segnale chirp; tuttavia l'ondina utilizzata (Haar) non è per niente adatta al tipo di segnale perché con pochi livelli di scomposizione l'approssimazione del segnale d'ingresso è pessima. Ciò è molto evidente poiché abbiamo, inoltre, posto vicino le uscite del banco con un segnale d'ingresso di sequenze di porte rettangolari per il quale si adatta benissimo l'ondina di Haar; la maggior parte dei livelli di dettaglio non hanno alcun contenuto, nel primo caso $k = 3$ il livello superiore L1 contiene il segnale per intero, mentre nel secondo caso $k = 7$ la scomposizione riesce a distinguere la componente costante del segnale ponendola al livello approssimazione L1 e la componente a onda quadra con valor medio nullo con una certa frequenza al livello di dettaglio K3. Considerando sempre filtri ortogonali utilizziamo un'altra ondina: *Daibechies(10).*

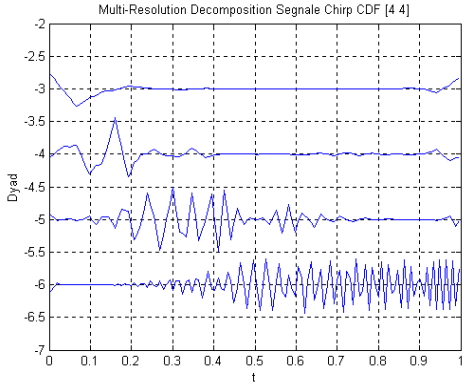


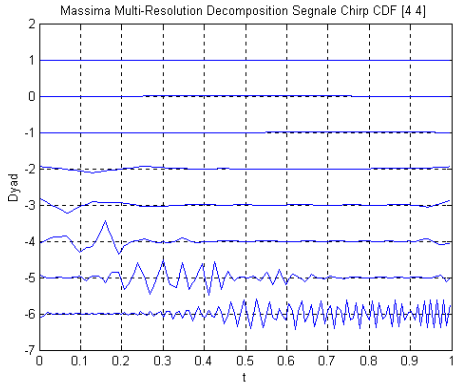
In questi grafici ritroviamo quasi una situazione opposta alla precedente: con tale scomposizione in bande il contenuto in frequenza del segnale sequenza di porte viene "spalmato" nei vari livelli, mentre il segnale chirp ciene approssimato molto meglio in questo caso anche se l'ulteriore scomposizione per $k = 7$ non permette in entrambi i casi di distribuire il contenuto in frequenza in modo migliore (è concentrato nei primi 4 o 5 livelli di dettaglio).

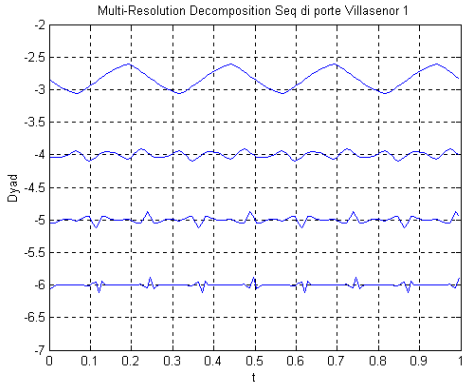
Ripetiamo l'esperienza questa volta con filtri biortogonali che richiedono un progetto più complesso ma sono comunque utilizzati spesso in quanto offrono più gradi di libertà, permettendo di distinguere le caratteristiche dei filtri di analisi da quelli di sintesi. Inoltre permettono una perfetta ricostruzione con fase lineare, cosa molto difficile utilizzando filtri ortonormali fatta eccezione del banco di Haar. Di seguito riportiamo l'uscita del relativo banco di analisi con ondina CDF(4 4), Villasenor 1 e 4 con entrambi i segnali, realizzato con il comando FWT_PBS (Trasformata Wavelet Diretta Periodicizzata Biortogonale Simmetrica) che richiede come parametri oltre al segnale $x[n]$, il parametro L definito prima, qmf e $dqmf$ (quest'ultimo è il vettore dei coefficienti dei filtri di sintesi), calcolati mediante il comando MakeBSFilter.

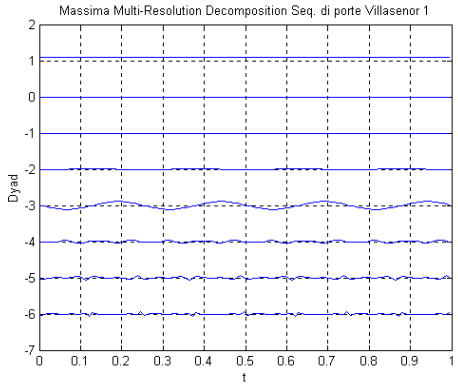

Multi-Resolution Decomposition Segnale seq. porte rettangolari CDF [4 4]
Dyad
t

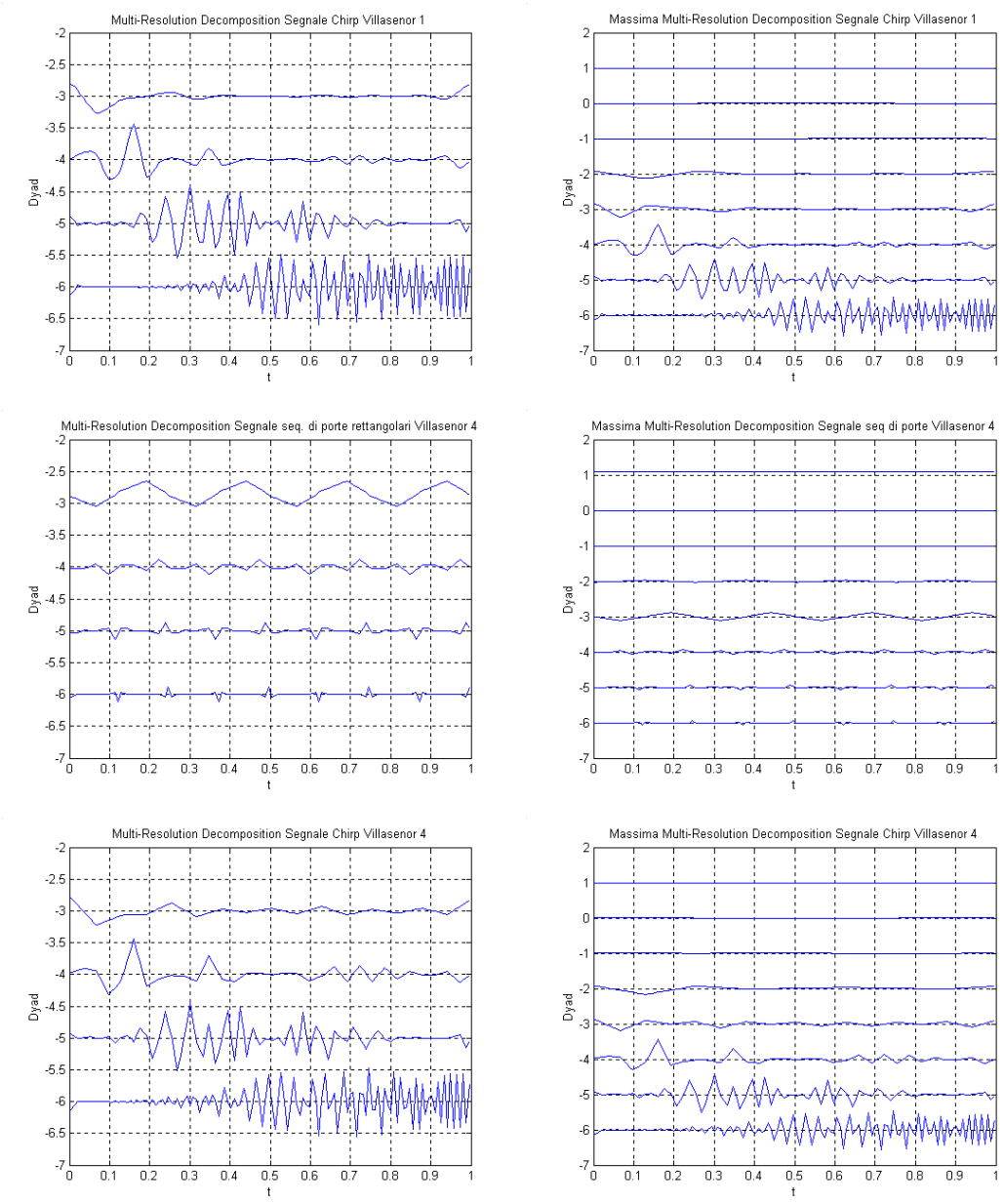

Massima Multi-Resolution Decomposition seq. porte rettangolari CDF [4 4]
Dyad
t


Multi-Resolution Decomposition Segnale Chirp CDF [4 4]
Dyad
t


Massima Multi-Resolution Decomposition Segnale Chirp CDF [4 4]
Dyad
t


Multi-Resolution Decomposition Seq di porte Villasenor 1
Dyad
t

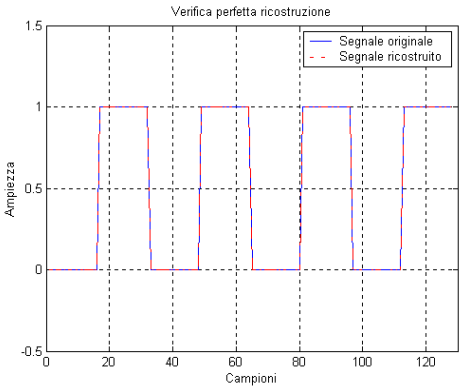

Massima Multi-Resolution Decomposition Seq. di porte Villasenor 1
Dyad
t



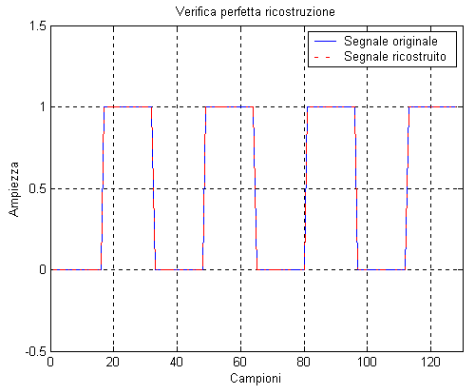
I commenti che si possono fare a riguardo non sono molto differenti da quelli fatti a seguito dei grafici ottenuti con banco di Daibechies (10) se non che i contributi di frequenza sembrano più distribuiti nel caso di Daubechies piuttosto che con questi filtri biortogonali, ma verificheremo queste differenze nella valutazione del contenuto energetico.

Per quanto riguarda la perfetta ricostruzione viene realizzata implementando il banco di sintesi mediante i comandi IWT_PO e IWT_SBS a seconda del tipo di filtro; i risultati di ricostruzione utilizzando l'intera informazione

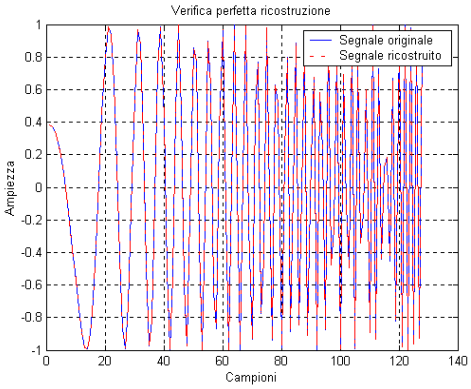
prodotta dal banco di analisi sono verificati indipendentemente dal segnale d'ingresso e dal tipo di filtri utilizzati. Ne riportiamo alcuni.



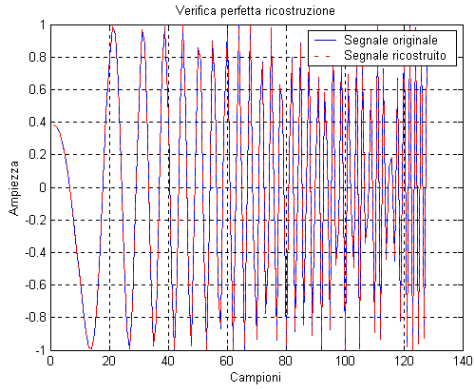
PR con banco CDF [4 4]



PR con banco Daubechies (10)

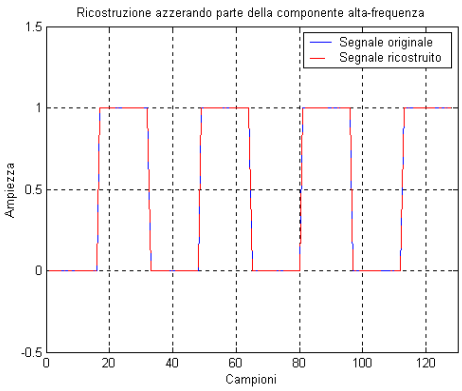


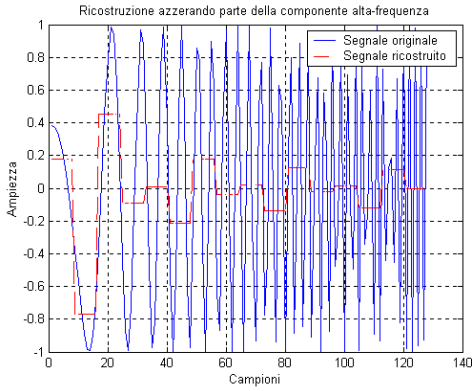
PR con banco di Haar



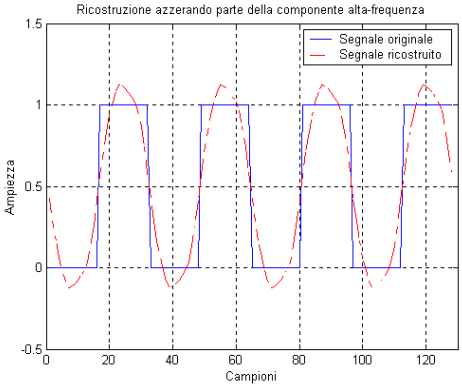
PR con banco Villasenor 4

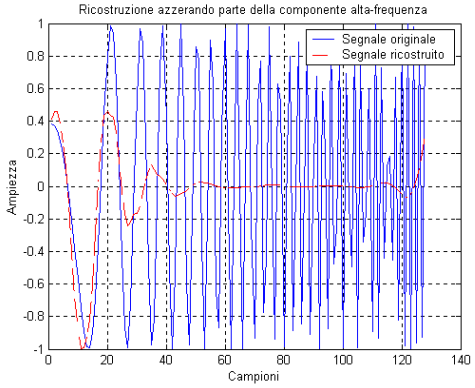
Successivamente abbiamo provato ad eseguire la ricostruzione del segnale di partenza azzerando l'informazione ad alta frequenza, precisamente i livelli H1, H2 e H3 partendo da una scomposizione a tre livelli. Il risultato dipende dal tipo di segnale in ingresso e dal tipo di banco utilizzato; in generale si può dire che se il banco è adatto al segnale d'ingresso l'approssimazione è migliore (vedi ricostruzione con banco di Haar del segnale sequenza di porte), ma anche che il segnale chirp ben difficilmente può essere ricostruito visto che ha molte componenti ad alta frequenza, infatti solo la parte iniziale di questo segnale può essere ricostruita fedelmente.



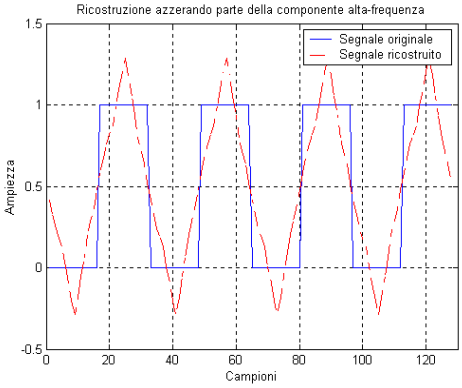


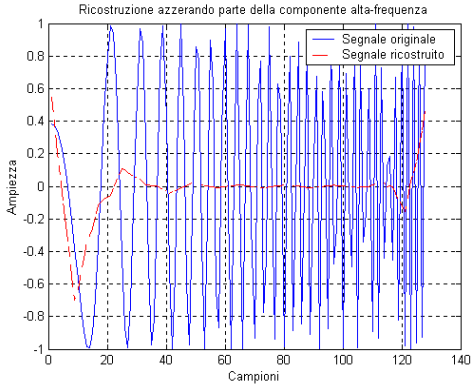
Ricostruzioni con banco di Haar



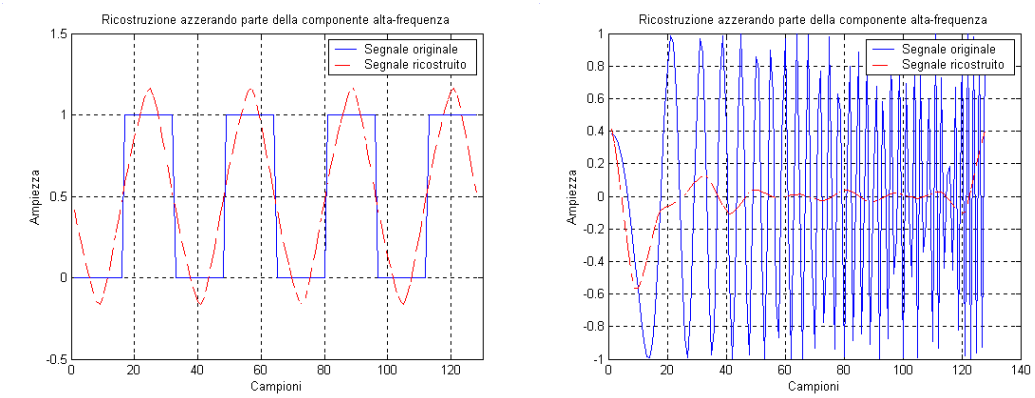


Ricostruzioni con banco di Daubechies (10)

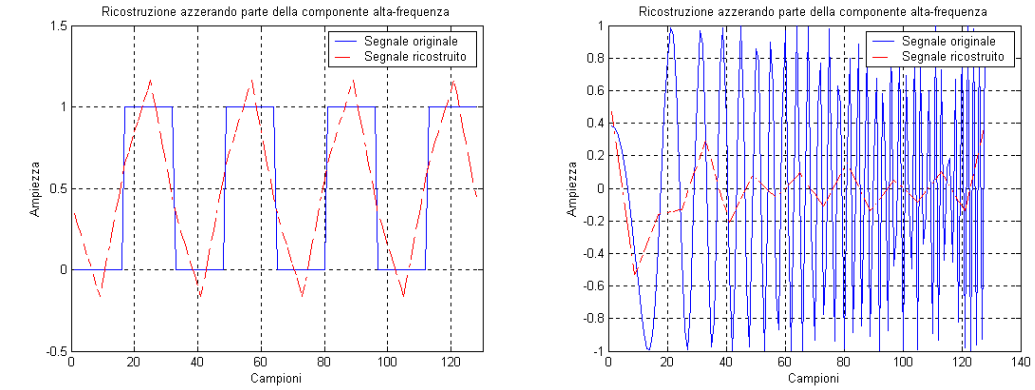




Ricostruzioni con banco di CDF [4 4]



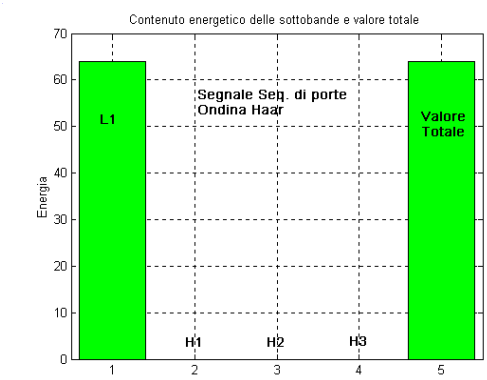
Ricostruzioni con banco di Villasenor 1

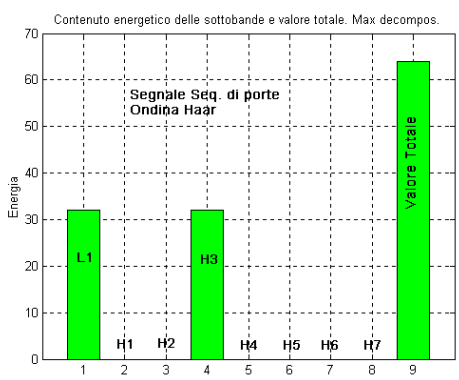


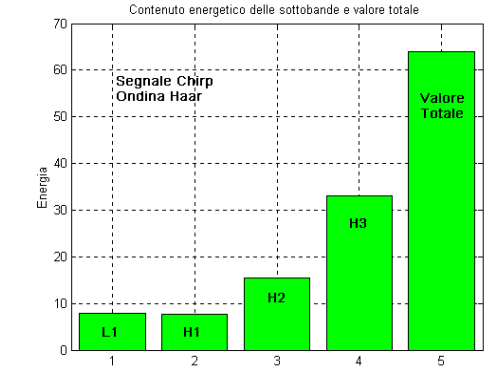
Ricostruzioni con banco di Villasenor 4

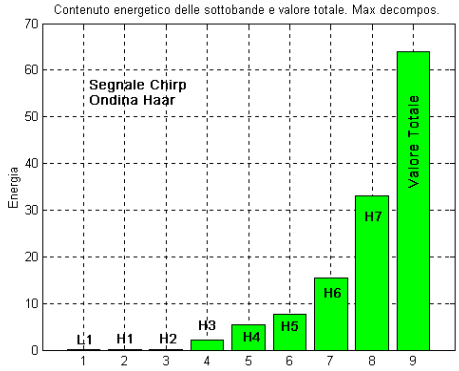
A termine di questa fase di sperimentazione è stato fatto il calcolo del contenuto energetico delle singole sottobande; questo ha portato alla verifica di ciò che si è detto a seguito delle precedenti esperienze. Ciò che può risaltare dai successivi grafici è che l'andamento del contenuto energetico del segnale chirp nelle relative sottobande è che aumenta all'aumentare del livello di dettaglio fino ad azzerarsi al livello di approssimazione contrariamente al segnale sequenze di porte dove la distribuzione energetica varia molto in funzione del banco considerato. Questa variazione non è così marcata per il segnale chirp.

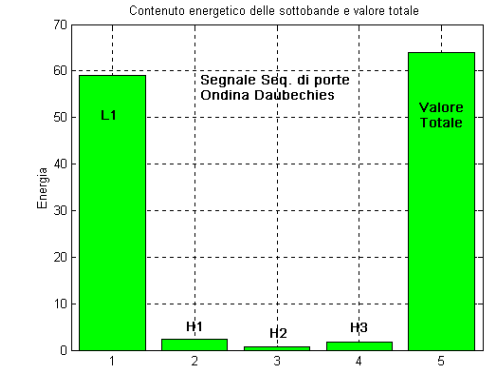
Di seguito riportiamo degli istogrammi del contenuto energetico delle varie sottobande calcolato per ogni tipo di banco trattato, confrontandoli direttamente col contenuto energetico totale del segnale, sia per $k = 3$ che per $k = 7$ (in tal caso si evidenzia l'andamento del contenuto energetico di cui si parlava nel primo paragrafo).

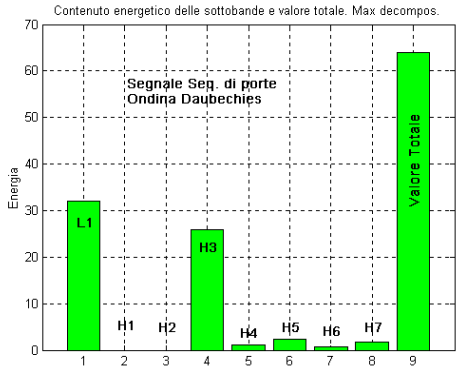

Contenuto energetico delle sottobande e valore totale
Segnale Seq. di porte
Ondina Haar
L1
H1
H2
H3
Valore
Totale
Energia

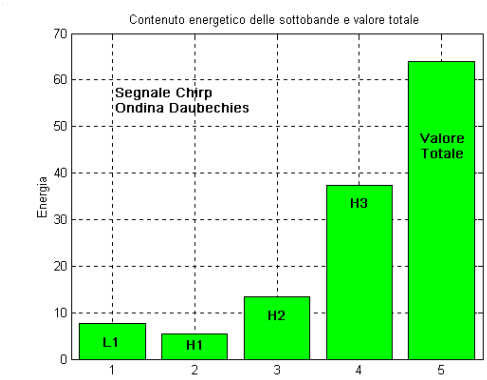

Contenuto energetico delle sottobande e valore totale. Max decompos.
Segnale Seq. di porte
Ondina Haar
L1
H1
H2
H3
H4
H5
H6
H7
Valore Totale
Energia

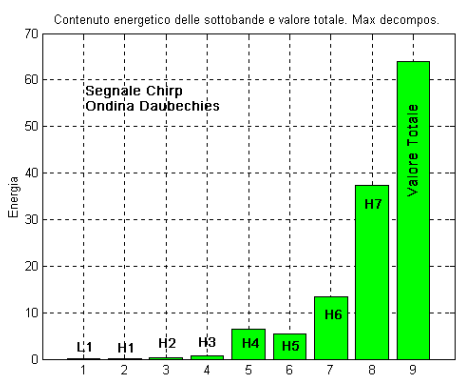

Contenuto energetico delle sottobande e valore totale
Segnale Chirp
Ondina Haar
L1
H1
H2
H3
Valore
Totale
Energia

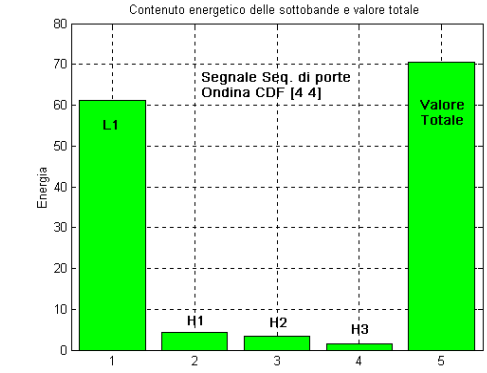

Contenuto energetico delle sottobande e valore totale. Max decompos.
Segnale Chirp
Ondina Haar
L1
H1
H2
H3
H4
H5
H6
H7
Valore Totale
Energia

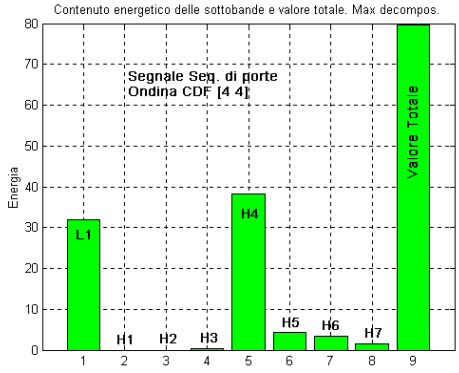

Contenuto energetico delle sottobande e valore totale
Segnale Seq. di porte
Ondina Daubechies
L1
H1
H2
H3
Valore
Totale
Energia

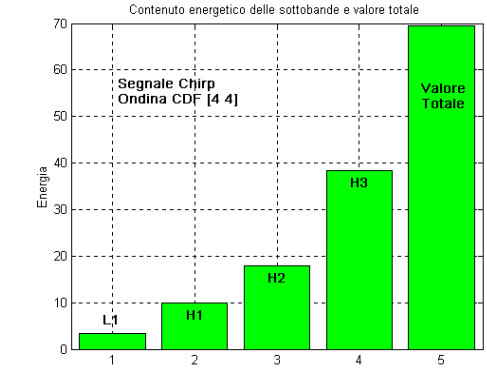

Contenuto energetico delle sottobande e valore totale. Max decompos.
Segnale Seq. di porte
Ondina Daubechies
L1
H1
H2
H3
H4
H5
H6
H7
Valore Totale
Energia

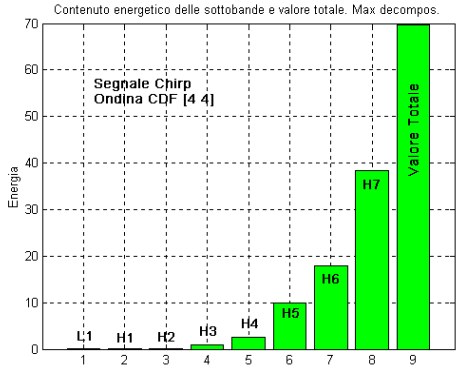

Contenuto energetico delle sottobande e valore totale
Segnale Chirp
Ondina Daubechies
Energia
L1
H1
H2
H3
Valore Totale

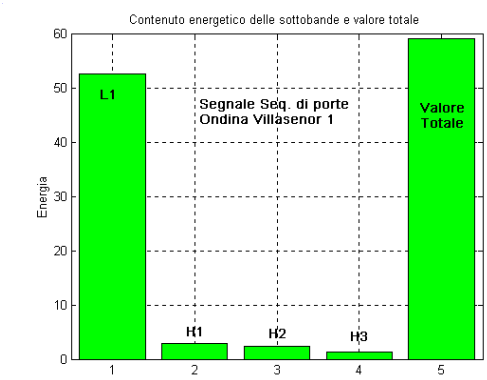

Contenuto energetico delle sottobande e valore totale. Max decompos.
Segnale Chirp
Ondina Daubechies
Energia
L1
H1
H2
H3
H4
H5
H6
H7
Valore Totale

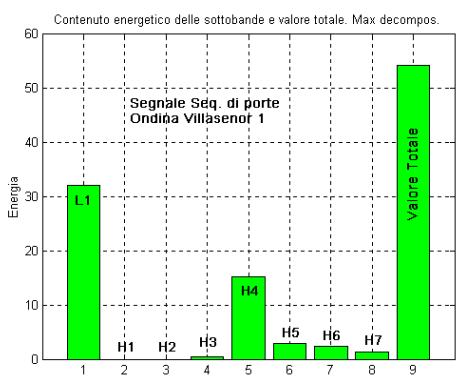

Contenuto energetico delle sottobande e valore totale
Segnale Seq. di porte
Ondina CDF [4 4]
Energia
L1
H1
H2
H3
Valore Totale

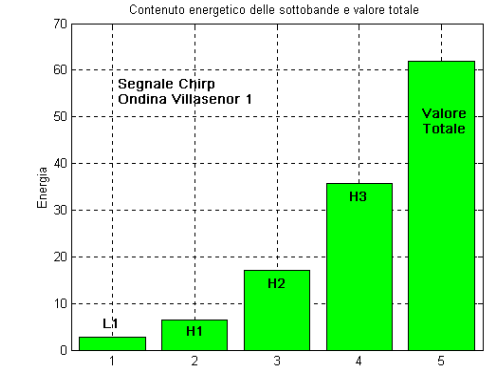

Contenuto energetico delle sottobande e valore totale. Max decompos.
Segnale Seq. di porte
Ondina CDF [4 4]
Energia
L1
H1
H2
H3
H4
H5
H6
H7
Valore Totale

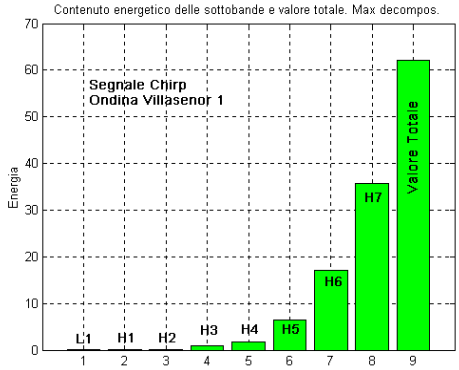

Contenuto energetico delle sottobande e valore totale
Segnale Chirp
Ondina CDF [4 4]
Energia
L1
H1
H2
H3
Valore Totale

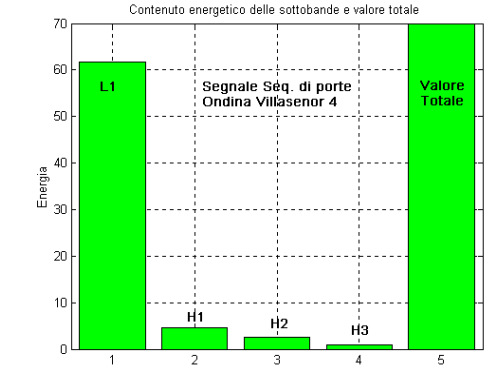

Contenuto energetico delle sottobande e valore totale. Max decompos.
Segnale Chirp
Ondina CDF [4 4]
Energia
L1
H1
H2
H3
H4
H5
H6
H7
Valore Totale

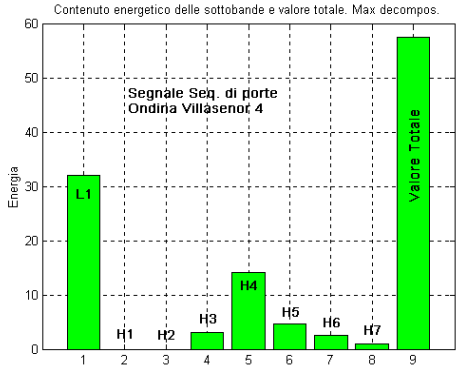

Contenuto energetico delle sottobande e valore totale
Segnale Seq. di porte
Ondina Villasenor 1
L1
H1
H2
H3
Valore
Totale
Energia

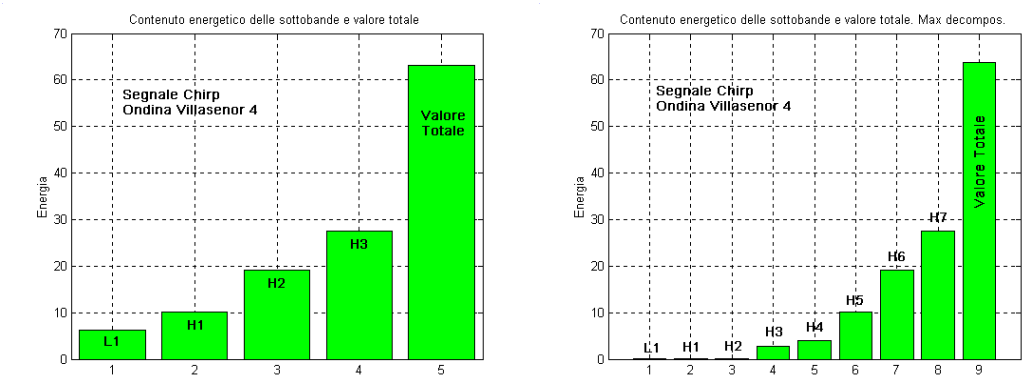

Contenuto energetico delle sottobande e valore totale. Max decompos.
Segnale Seq. di porte
Ondina Villasenor 1
L1
H1
H2
H3
H4
H5
H6
H7
Valore Totale
Energia


Contenuto energetico delle sottobande e valore totale
Segnale Chirp
Ondina Villasenor 1
L1
H1
H2
H3
Valore
Totale
Energia


Contenuto energetico delle sottobande e valore totale. Max decompos.
Segnale Chirp
Ondina Villasenor 1
L1
H1
H2
H3
H4
H5
H6
H7
Valore Totale
Energia


Contenuto energetico delle sottobande e valore totale
Segnale Seq. di porte
Ondina Villasenor 4
L1
H1
H2
H3
Valore
Totale
Energia

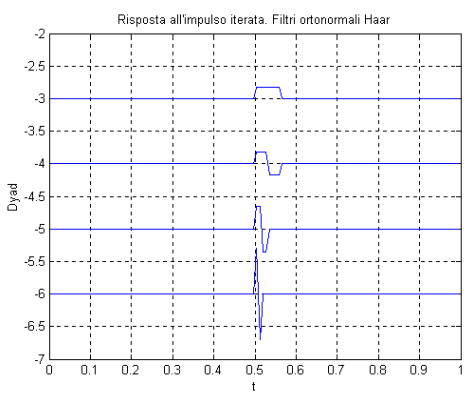

Contenuto energetico delle sottobande e valore totale. Max decompos.
Segnale Seq. di porte
Ondina Villasenor 4
L1
H1
H2
H3
H4
H5
H6
H7
Valore Totale
Energia

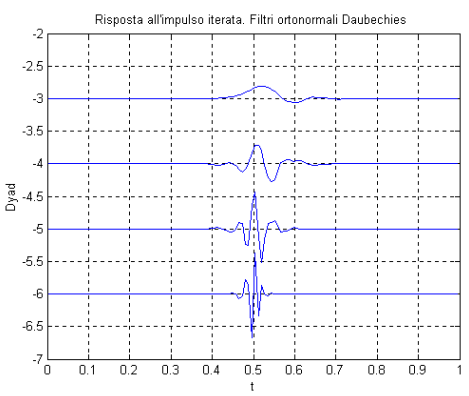


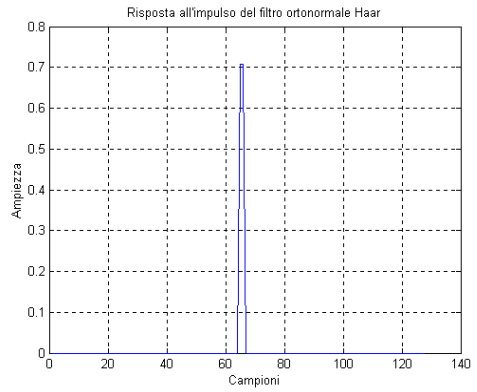
Nel caso di segnale sequenza di porte ritroviamo, con qualsiasi banco, metà del contenuto energetico dovuto alla componente continua del segnale e quindi al livello approssimazione L1; il rimanente contenuto lo si ritrova dal livello di dettaglio H3 in poi (facendo riferimento ovviamente ai grafici ottenuti con $k = 7$). Inoltre, a parte il banco di Haar, caso particolare già discusso ed evidente, per il segnale sequenza di porte si verifica una progressiva ininfluenza del contenuto alle frequenze più alte. Per il segnale chirp, come già detto, vale il vicecersa.

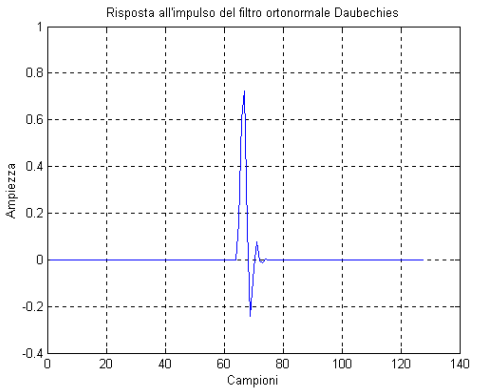
1.4 Risposte all'impulso

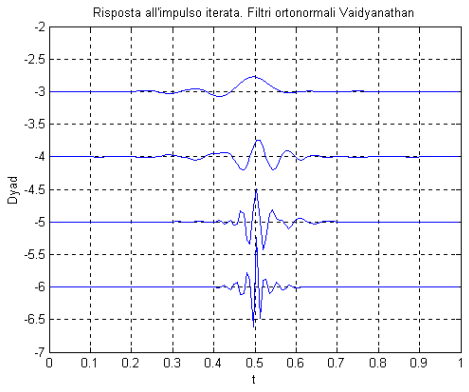
Vista la struttura del banco di analisi e di sintesi costituita da filtri, si è pensato di valutare la loro risposta all'impulso. Evidenziando l'iterazione si vede il perché si filtra componenti in frequenza sempre più alte in quanto l'andamento della risposta è più nervoso quindi con componenti ad alta frequenza. Tale risposta è la base dello spazio di funzioni appartenenti a quel livello. Nei primi due livelli della scomposizione ritroviamo l'ondina padre e l'ondina madre utilizzati, evidenziati, perdipiù, dalla risposta all'impulso di un filtro a cui si pone come coefficienti quelli trovati come qmf e $dqmf$, rispettivamente per il filtro di analisi e sintesi.

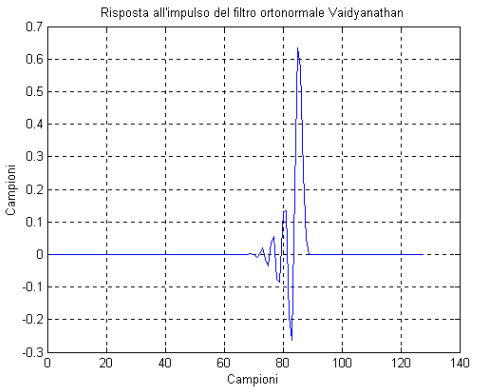

Risposta all'impulso iterata. Filtri ortonormali Haar
Dyad
t

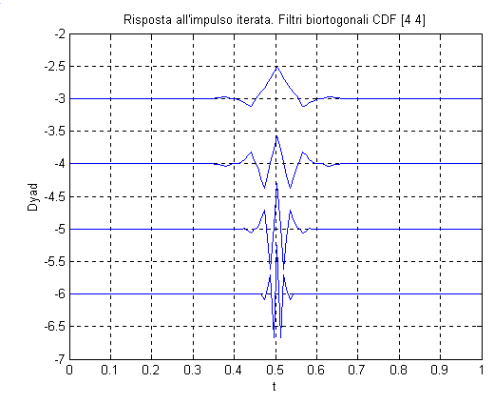

Risposta all'impulso iterata. Filtri ortonormali Daubechies
Dyad
t

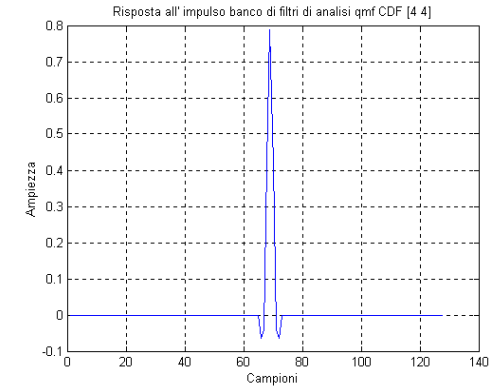

Risposta all'impulso del filtro ortonormale Haar
Ampiezza
Campioni

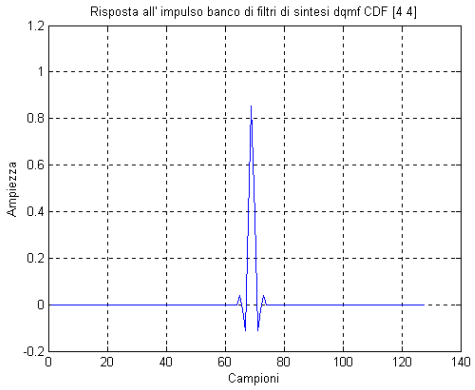

Risposta all'impulso del filtro ortonormale Daubechies
Ampiezza
Campioni

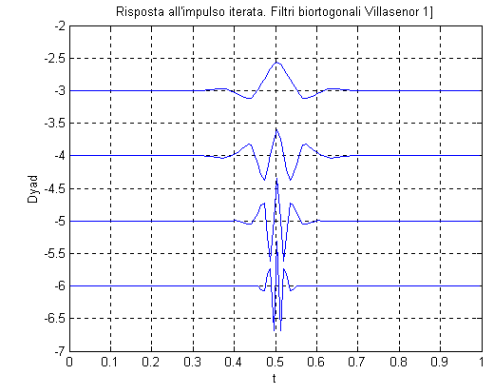

Risposta all'impulso iterata. Filtri ortonormali Vaidyanathan
Dyad
t

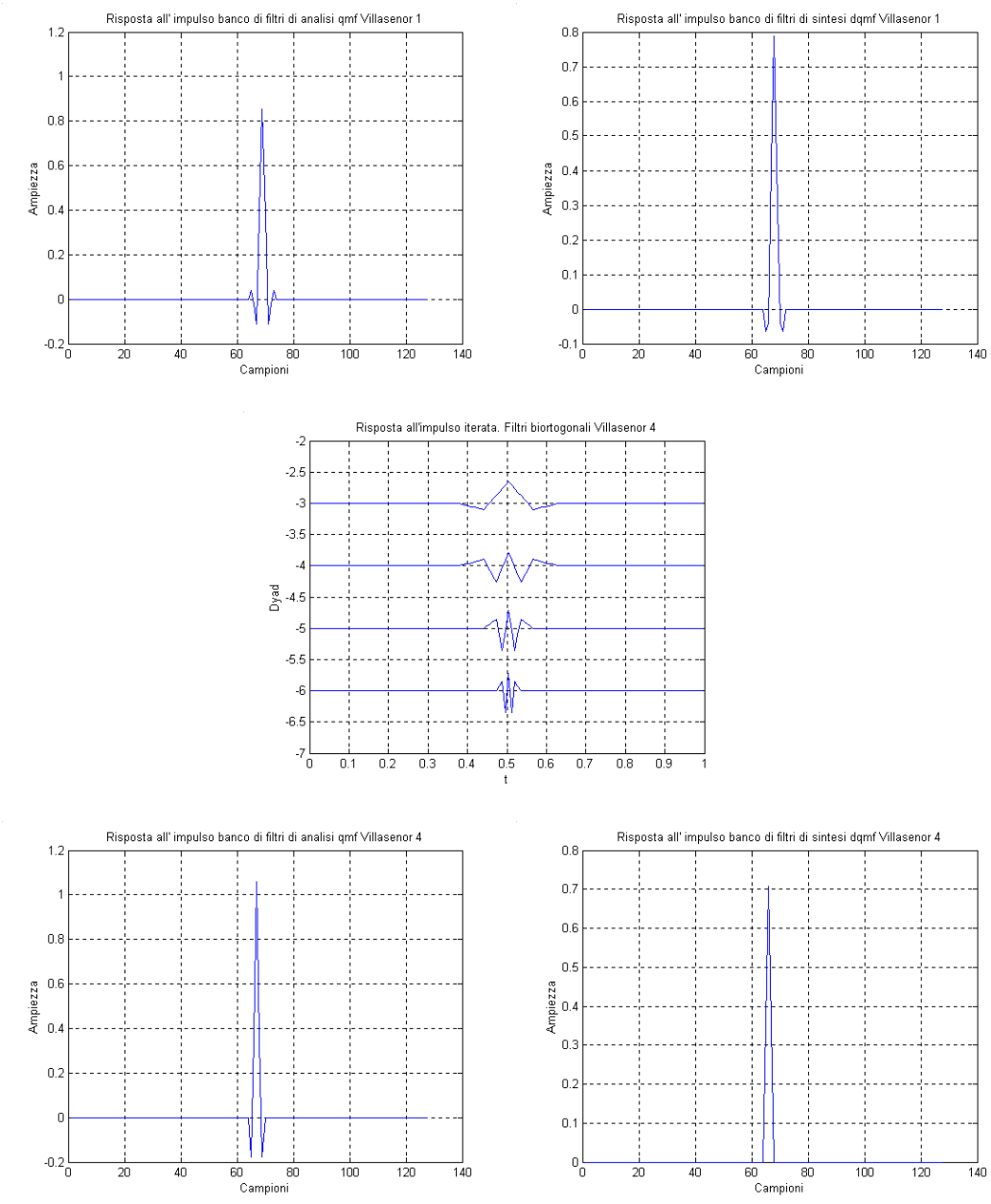

Risposta all'impulso del filtro ortonormale Vaidyanathan
Campioni
Campioni


Risposta all'impulso iterata. Filtri biortogonali CDF [4 4]
Dyad
t


Risposta all' impulso banco di filtri di analisi qmf CDF [4 4]
Ampiezza
Campioni


Risposta all' impulso banco di filtri di sintesi dqmf CDF [4 4]
Ampiezza
Campioni


Risposta all'impulso iterata. Filtri biortogonali Villasenor 1]
Dyad
t



Attraverso questi grafici notiamo che l'ondina relativa ai tre filtri biortogonali non presentano una differenza basilare, questo dimostra il motivo della somiglianza dei risultati precedenti applicando il banco corrispondente ai due segnali d'ingresso.

Si è introdotto l'esame anche di un altro banco di filtri ortogonale denominato Vaidyanathan che è particolarmente adatto a trattare il segnale audio perché estremamente selettivo, indìfatti nella sezione successione ne faremo uso.

1.5 Seconda fase sperimentale

In questa seconda fase abbiamo provato ad utilizzare la procedura descritta precedentemente per segnali audio reali:

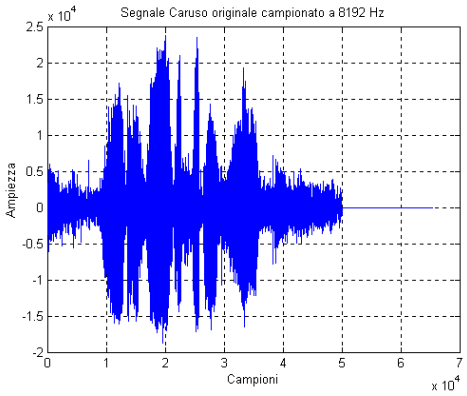
- il primo contenuto in formato ASCII nel toolbbox di Wavelab è una porzione della "Tosca" di G. Puccini cantata da Enrico Caruso; tale registrazione è molto disturbata da rumore di fondo, è campionata a 8000Hz e vine caricata da Matlab in un vettore mediante il comando ReadSignal sempre contenuto nello stesso toolbox;

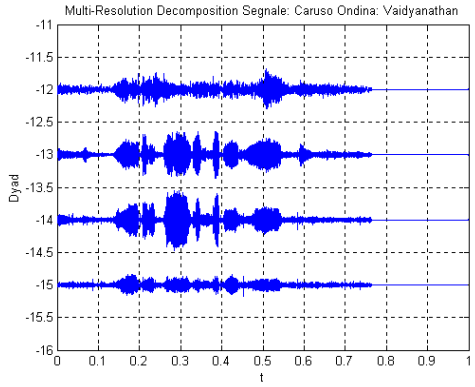
- il secondo segnale è l'incipit della "Rapsodia in Blue" di G. Gershwin eseguita al clarinetto, registrato presso l'Istituto Elettrotecnico Nazionale "G. Ferraris" di Torino in camera di misura della prova di calpestio. La registrazione è stata fatta con la massima qualità possibile (48000 Hz, 16 bit) ponendo attenzione alla saturazione dello strumento di misura utilizzato (un DAT), con l'assistenza dell' Ing. Andrea Pavoni Belli; per la manipolazione al computer è stato modificato il formato della registrazione mediante il programma Wavestudio (11000 Hz, 16 bit).

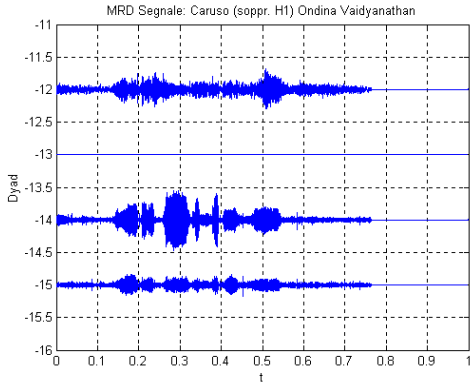
- il terzo segnale non è altro che il precedente limitato alla fase del glissato (rapida escursione da una nota grave dello strumento ad una acuta), eliminando così il motivo successivo (sempre al clarinetto) d'introduzione nella Rapsodia dell'orchestra.

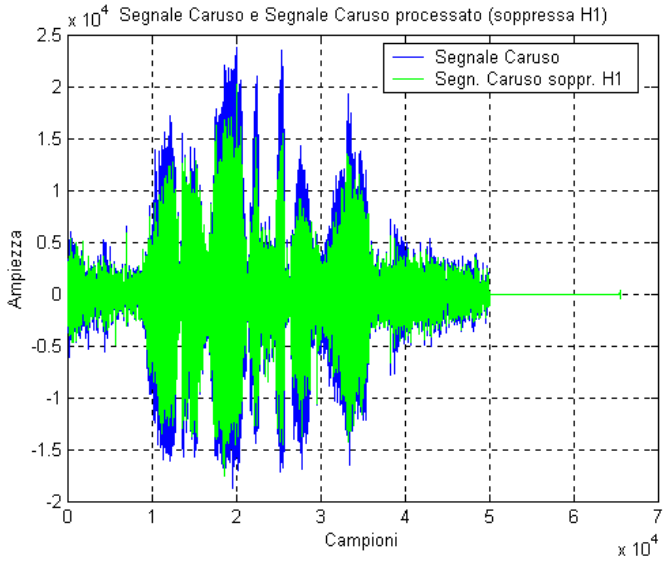
Una volta caricati in un vettore (per il secondo e il terzo si usa il comando Wavread di Matlab), la lunghezza di tale vettore deve essere resa diadica (potenza di due) per applicare la DWT aggiungendo degli zeri (riscontrabili nei grafici che riporteremo).

Per quanto riguarda il primo segnale ne abbiamo eseguito la trasformata wavelet con banco di Vaidyanathan, abbiamo soppresso la sottobanda H1 (in particolare abbiamo usato una scomposizione su tre livelli), ne abbiamo fatto l'antitrasformata e abbiamo provato ad ascoltare l'effetto di tale soppressione. L'effetto risultante all'ascolto non ha ribaltato molto quello dell'originale in quanto, ribadiamo, la registrazione era affetta da un rumore molto elevato; tuttavia si può notare un'affievolimento della brillantezza della voce (e del rumore anche se non sufficiente), tipico della riduzione di componenti ad alta frequenza. Di seguito riportiamo i grafici relativi all'esperienza.



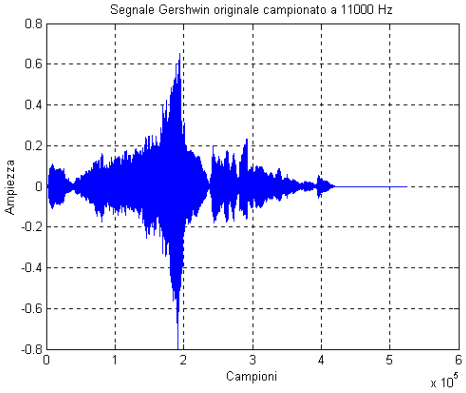


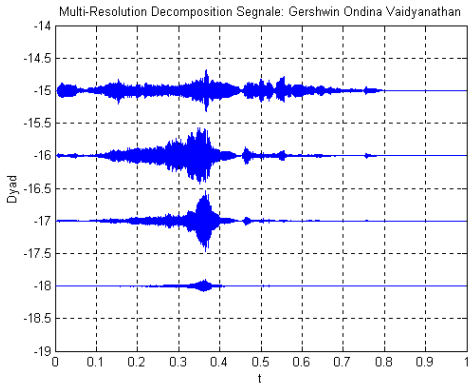


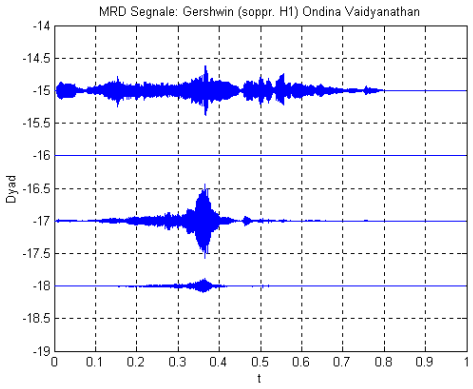


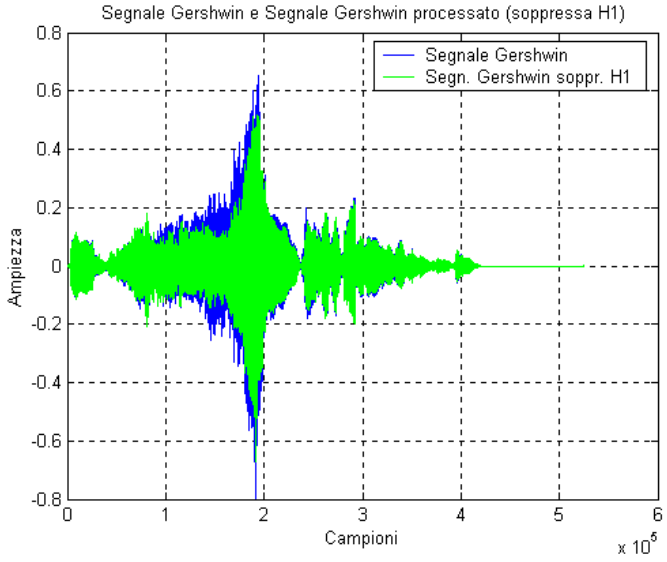
Come si può notare l'effetto della soppressione di H1 è più evidente nel grafico dei due segnali a confronto che all'ascolto.

Per il secondo segnale abbiamo eseguito l'identica procedura descritta prima col segnale di Caruso e ne riportiamo i grafici.



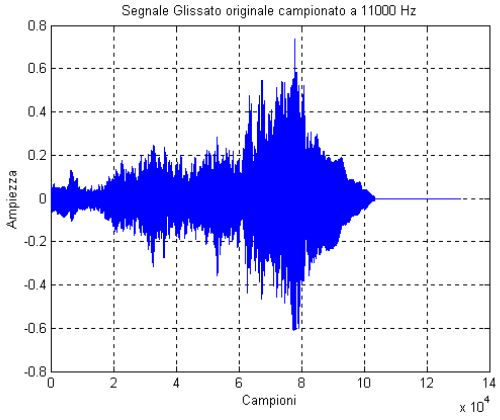




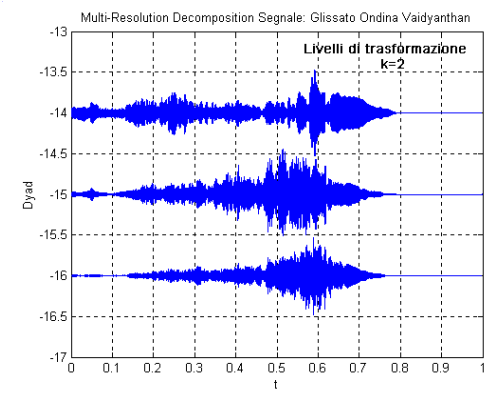


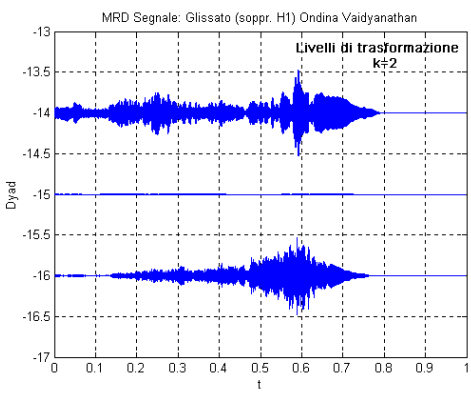
Il grafico di Multi-Risoluzione permette di verificare che le componenti presenti sono prevalentemente a bassa frequenza e solo in corrispondenza della fase terminale del glissato vi sono componenti ad alta frequenza. Questo concorda con le caratteristiche acustiche dello strumento che presenta un naturale filtro acustico con frequenza di taglio intorno ai 1500 Hz e solo suonando particolarmente forte determinate note si riesce a superare l'attenuazione di questo filtro naturale (è proprio il caso della fase terminale del glissato). Soppressa la sottobanda H1 la ricostruzione, come nel caso precedente, evidenzia una certa differenza col segnale originale ma a differenza del caso precedente l'effetto all'ascolto è abbastanza forte: l'effetto che si ha è analogo all'effetto di una sordina posta ad una tromba (quasi un ronzio), inoltre, nella prima fase del glissato, si ascolta in maniera sufficientemente distinta, l'effetto di un glissato con andamento contrario (decrescente) che prima non c'era forse perché nascosto e che può essere originato dal sottocampionamento, infine si nota che il motivo introduttivo all'orchestra (seconda sezione del segnale) non presenta rilevanti modifiche e questo lo si nota anche dal grafico.

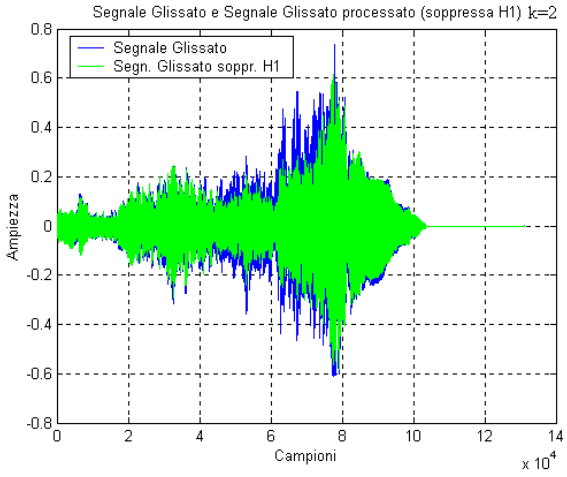
A questo punto è nata la curiosità di esaminare l'effetto della soppressione di H1 sul segnale precedente limitandoci alla fase del glissato e a visualizzare in tempo-frequenza il risultato ottenuto confrontandolo con quello trovato in occasione della tesina del corso di "Analisi Statistica del Segnale". Perdipiù si è voluto fare questa verifica per diversi valori del livello di scomposizione k. Riportiamo di seguito i grafici trovati.

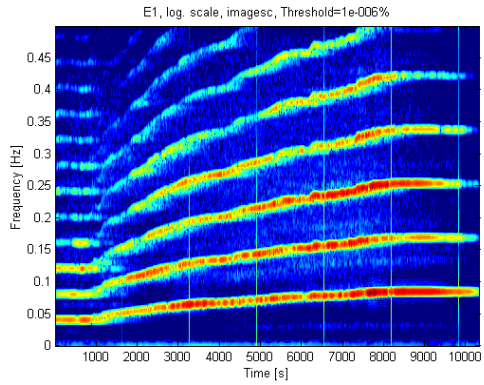


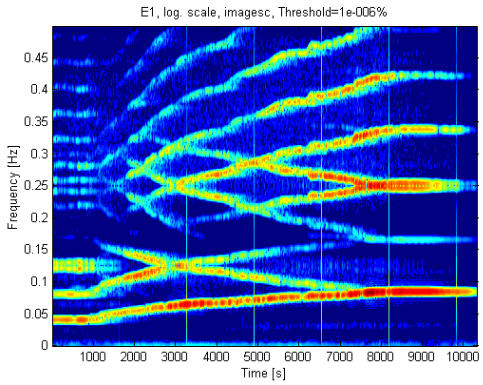
k=2;









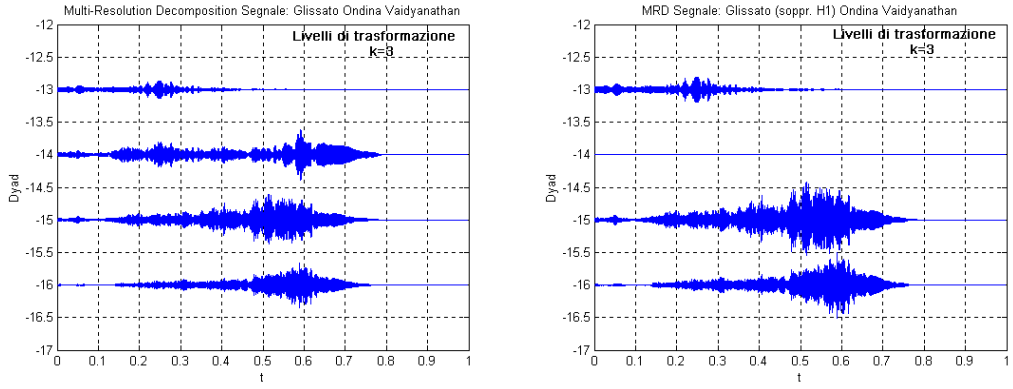


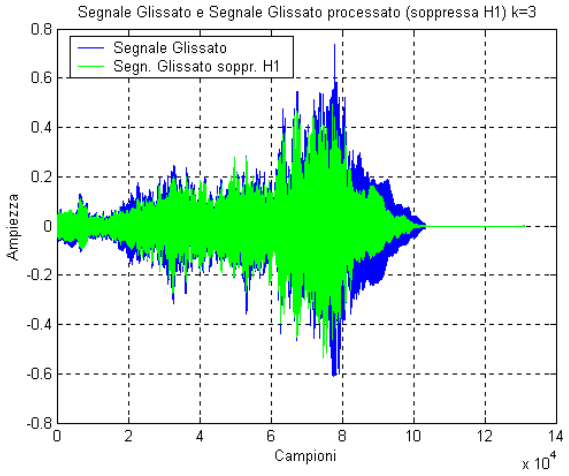
Glissato originale Glissato con H1 soppressa e k=2

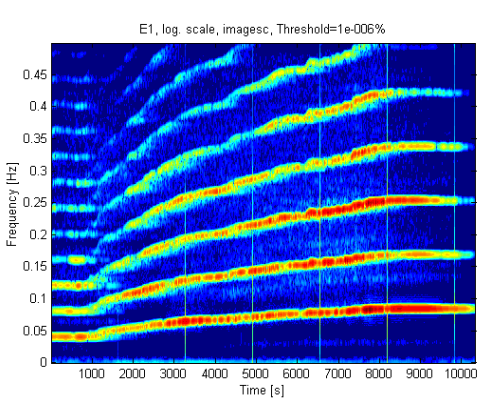
Con una scomposizione così povera l'effetto è disastroso in quanto il contenuto eliminato è consistente: dai grafici tempo-frequenza si può notare che

viene in particolar modo eliminata la 5^a armonica (si ricorda che un caratteristica dello strumento è di avere esclusivamente gli armonici di ordine dispari)nella prima fase del glissato e parzialmente della 3^a nella seconda fase; l'effetto che si diceva prima di un contemporaneo glissato con andamento decrescente è presente anche in questo caso ed è anche ben visibile nel grafico $t - f$; presente anche l'effeto "sordina".

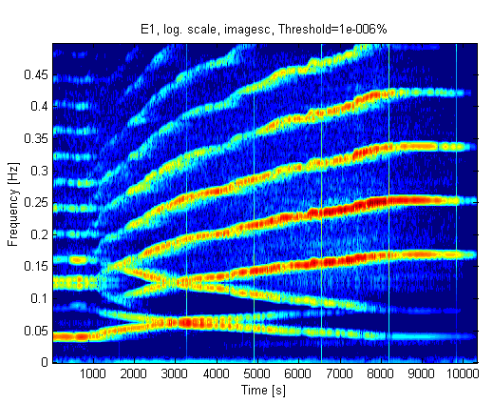
k=3;







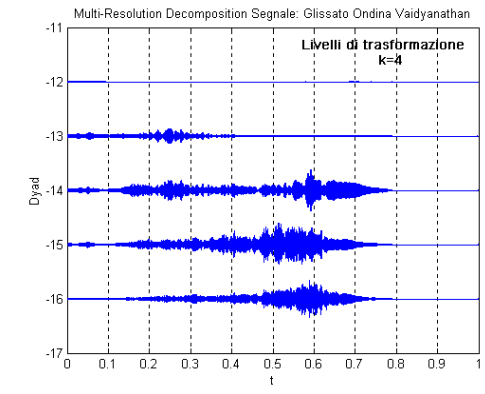
Glissato originale

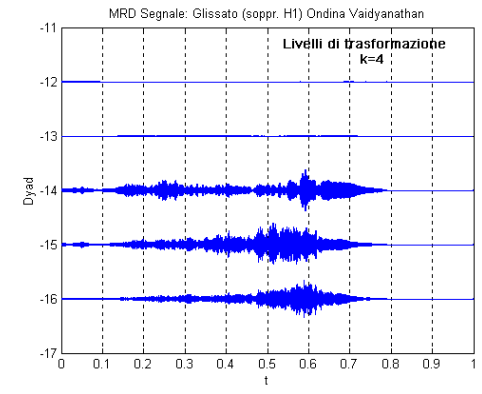


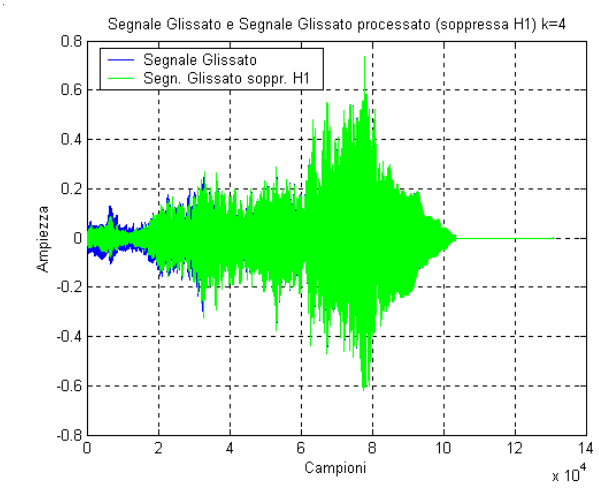
Glissato con H1 soppresso k=3

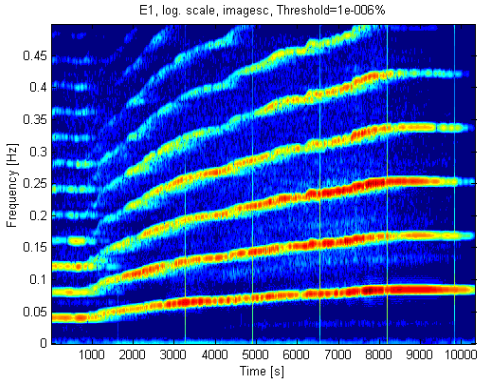
Dai grafici di Multi-Risoluzione si rileva che sopprimendo H1 si va ad eliminare la componente che possiamo definire sempre presente, infatti le altre hanno delle zone quasi azzerate, e ciò lo si può notare anche nel grafico di confronto dei due segnali dove non c'è una zona esentata da soppressione di componenti. L'effetto all'ascolto è stato già descritto precedentemente col secondo segnale analizzato, di cui faceva parte, con un forte (maggiore del caso con $k = 2$) effetto "sordina" e anche di glissato "opposto". Si può notare dal grafico $t - f$ che questa soppressione coincide con la soppressione della 3^a armonica nella prima fase del glissato e della $fondamentale$ nella seconda fase.

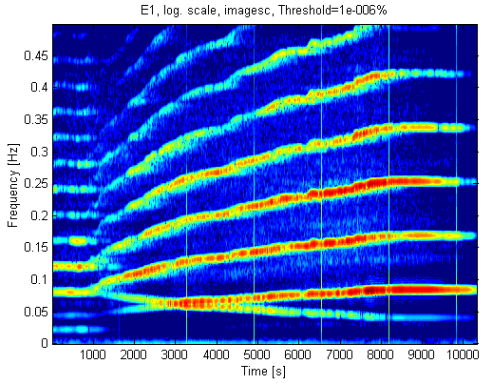
k=4;











Glissato originale Glissato sopprimendo H1 e con k=4

La prima cosa da notare con questa soppressione è la componente di approssimazione del segnale nel grafico Multi-Risoluzione (ricordiamo che è la prima in alto) che è praticamente nulla; si è potuto verificare che procedendo nella scomposizione non si ricava alcun vantaggio, le componenti del segnale rimangono sostanzialmente limitale ai primi 4 livelli di dettaglio ottenuti che abbiamo trattato fino ad ora. Per tale ragione questo è l'ultimo caso considerato. Anche dal grafico del confronto dei due segnali si rileva che la componente che si va ad eliminare è quella a bassa frequenza relativamente alle caratteristiche del segnale. I grafici $t - f$ confermano questa sensazione, cioè che viene eliminata la componente *fondamentale* della prima fase del segnale e poco altro (perché a bassa frequenza). L'effetto all'ascolto è di un glissato fatto come da un altro strumento più acuto o partendo da una nota differente; se non era per la presenza, anche se decisamente minore rispetto ai casi precedenti, di un glissato "opposto" non si parlerebbe di alterazione di un segnale.

Un ultima annotazione riguarda l'effetto "sordina" che è dovuto al fatto che le alte frequenze che andiamo ad eliminare sono enfatizzate in questo brano dal volume quindi è lecito che eliminandole il volume del segnale diminuisca; non per altro l'effetto "sordina" non è presente nell'ultimo caso.

Capitolo 2

Applicazione della DWT a immagini

2.1 Introduzione

In questo secondo capitolo affronteremo l'applicazione della DWT a segnali d'ingresso che rappresentano immagini. Infatti il segnale d'ingresso sarà una matrice di coefficienti che rappresentano la luminanza di un'immagine in bianco e nero. La scelta di operare unicamente con immagini in b/n è dovuta al fatto che l'occhio umano è particolarmente sensibile ai contrasti di luce e in misura minore alla crominanza dell'immagine stessa, quindi la difficoltà di operare su segnali che costituiscono immagini, è quellla di mantenere un buon contrasto, dei contorni degli oggetti dell'immagine ben definiti. L'informazione di crominanza, infatti, anche nella codifica del segnale televisivo viene manipolata con una cura minore ripetto alla luminanza.

I metodi utilizzabili per applicare la DWT sono principalmente due:

- mediante filtro bidimensionale applicato all'immagine; questa soluzione presenta difficoltà elevate e non viene da noi considerata;
- una volta caricata l'immagine come matrice, la considero come una successione di righe o colonne in modo alternato; visto che questo sarà il modo utilizzato verrà spiegata meglio la procedura.

Data la matrice dell'immagine se ne prende la prima riga e si calcola la trasformata wavelet (a questo punto è come se fosse un segnale monodimensionale trattati nel capitolo precedente) così avremo che la riga sarà scomposta in due sezioni: quella a sinistra conterrà il contenuto a bassa frequenza mentre la sezione di destra conterrà il contenuto ad alta frequenza.



Figura 2.1.

Ripetendo questo per ogni riga della matrice arriveremo ad avere una matrice scomposta in due sezioni verticali, quella di sinistra con contenuto a bassa frequenza mentre quella a destra ad alta frequenza.

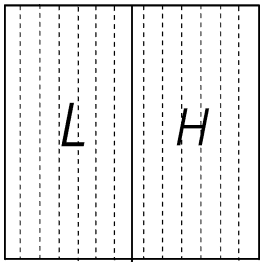


Figura 2.2.

A questo punto si applica nuovamente la trasformata wavelet considerando tutte le colonne della matrice appena ottenuta come nella figura successiva.

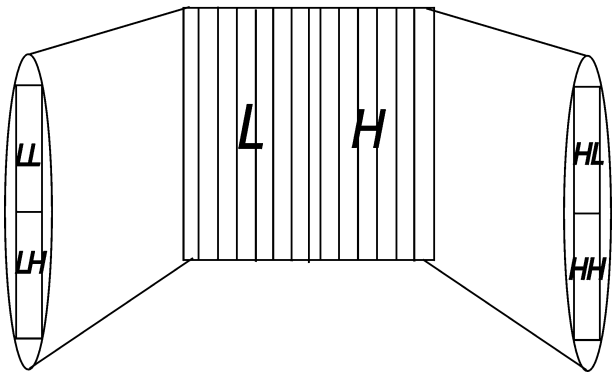


Figura 2.3.

Alla fine dopo aver effettuato la trasformata wavelet a tutte le colonne avremo

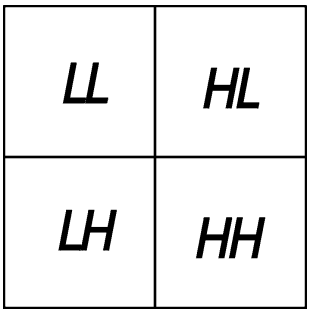


Figura 2.4.

Iterando la procedura come abbiamo fatto per il segnale monodimensionale sul livello a contenuto in frequenza minore (LL) otteniamo

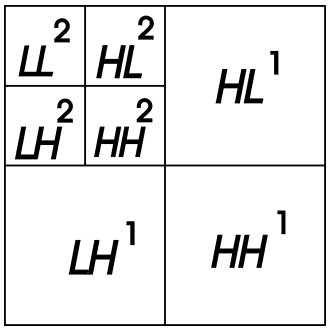


Figura 2.5.

In tal modo si può procedere idealmente all'infinito, nella realtà fino ad arrivare ad una matrice LL costituita da un unico coefficiente, analogamente al caso monodimensionale in cui concludevamo con un vettore di un unico coefficiente (valore continuo del segnale).

2.2 Prima fase sperimentale

La procedura schematizzata nell'introduzione sarà esplicitamente svolta nel seguito. Nel toolbox di Wavelab vi sono alcune immagini tra cui quella scelta

da noi, denominata “Lenna” rappresentante un primo piano di una ragazza e comprendente anche parte di uno sfondo. I comandi Matlab utilizzati per effettuare la DWT sono del tutto analoghi a quelli utilizzati nel dominio monodimensionale, anche nel nome: la trasformata wavelet diretta bidimensionale periodicizzata e ortogonale mediante FWT2_PO, e la relativa inversa IWT2_PO; la trasformata wavelet diretta bidimensionale periodicizzata biortogonale mediante FWT2_PBS, e la relativa inversa IWT2_PBS; i parametri d’ingressso a tali funzioni sono del tutto analoghi.



Figura 2.6.

I banchi di filtri che prenderemo in considerazione sono: Haar, Daubechies (10), Daubechies (20) (ortogonali), Villasenor 1, Villasenor 4 (biortogonali). Sarà considerata una scomposizione in sottobande su più livelli (precisamente 1, 2 e 3). Riportiamo di seguito l’uscita del corrispondente banco di filtri di analisi nei vari casi.

Banco di Haar;

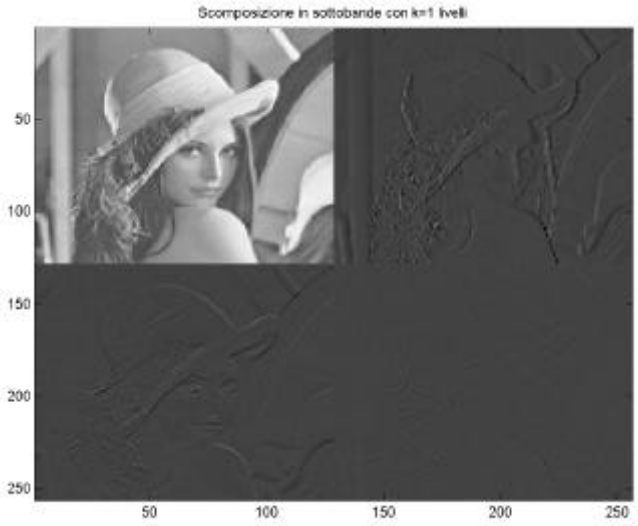


Figura 2.7.

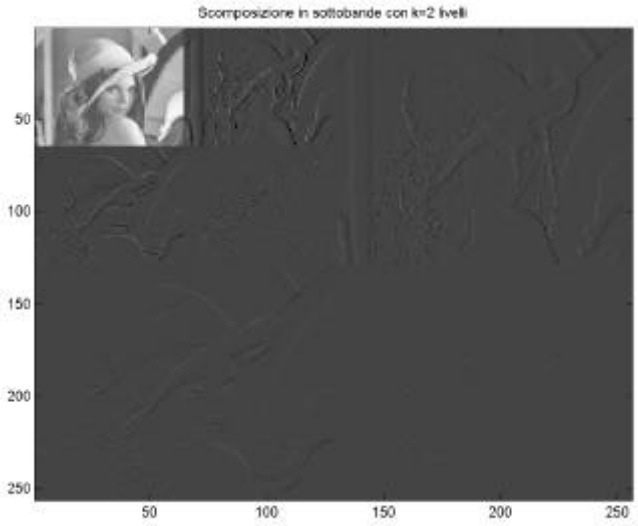


Figura 2.8.

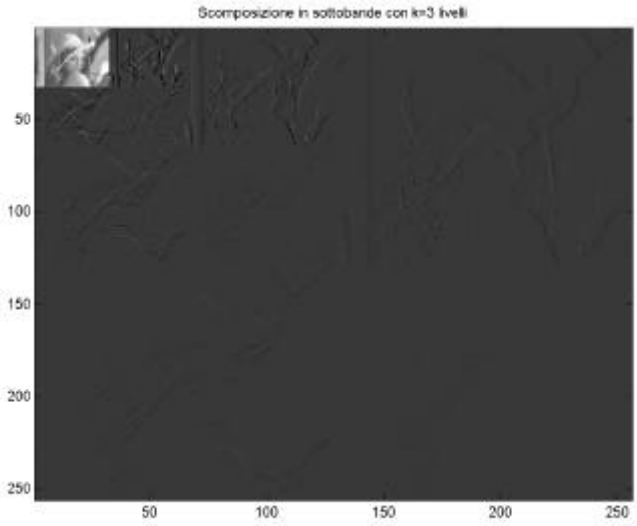


Figura 2.9.

Banco di Daubechies (10);

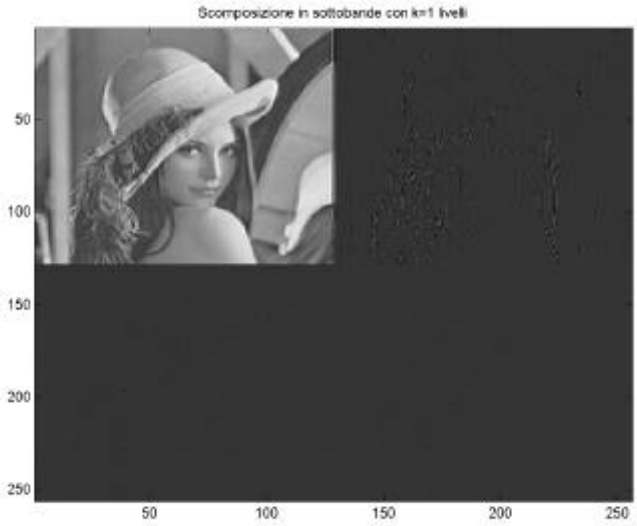


Figura 2.10.

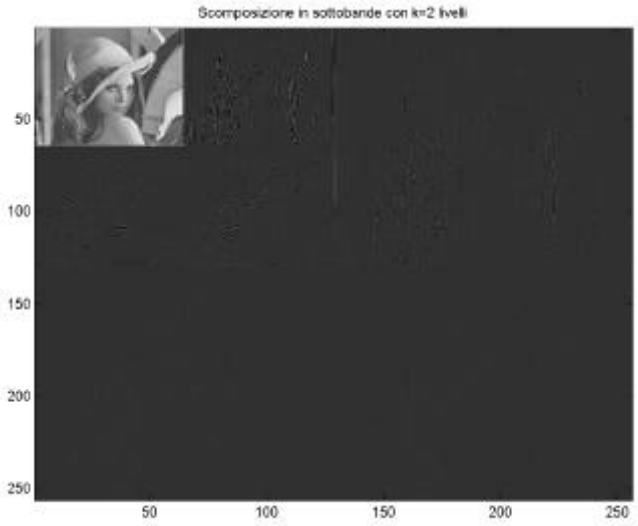


Figura 2.11.

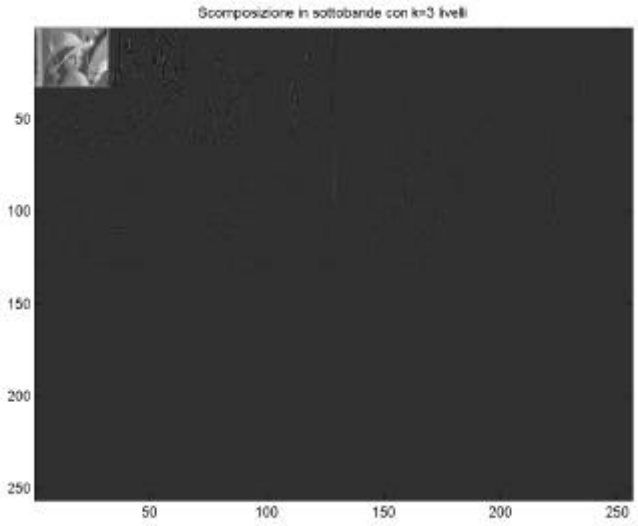


Figura 2.12.

Banco di Daubechies (20);

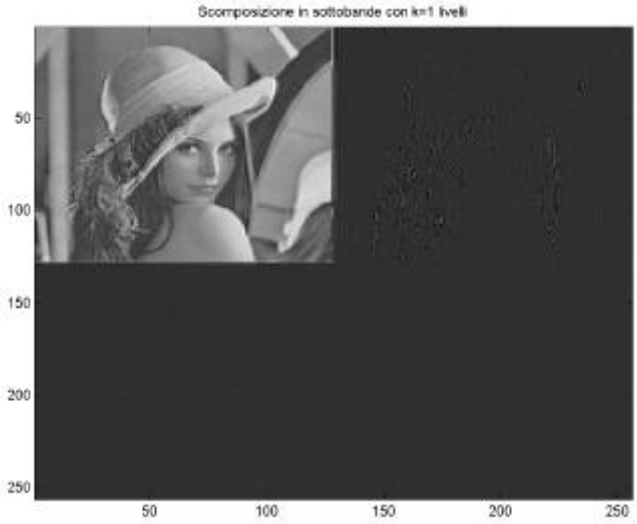


Figura 2.13.

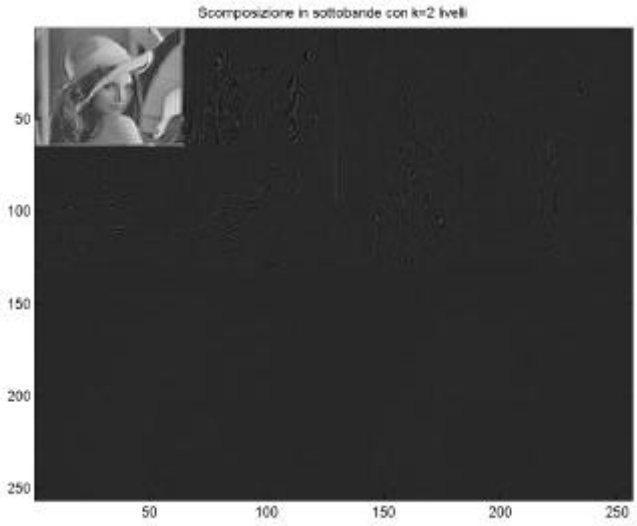


Figura 2.14.

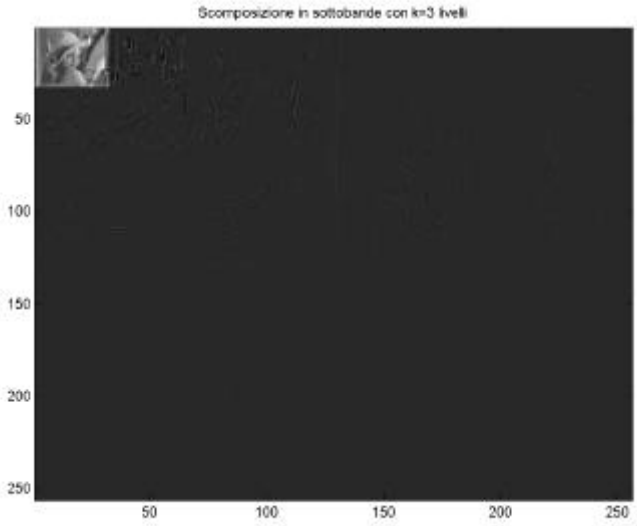


Figura 2.15.

Banco di Villasenor 1;

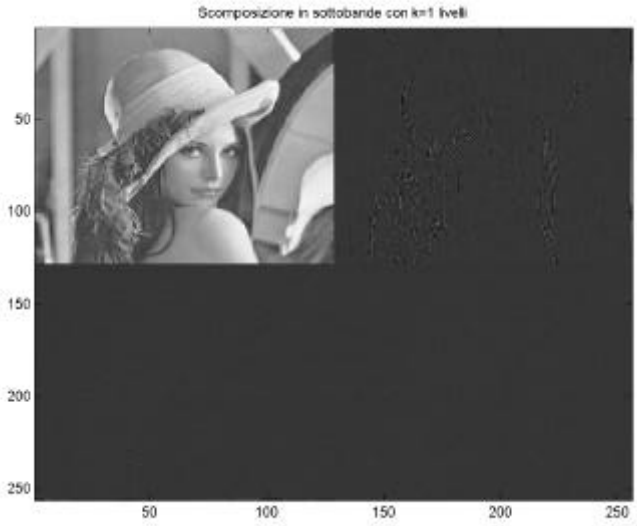


Figura 2.16.

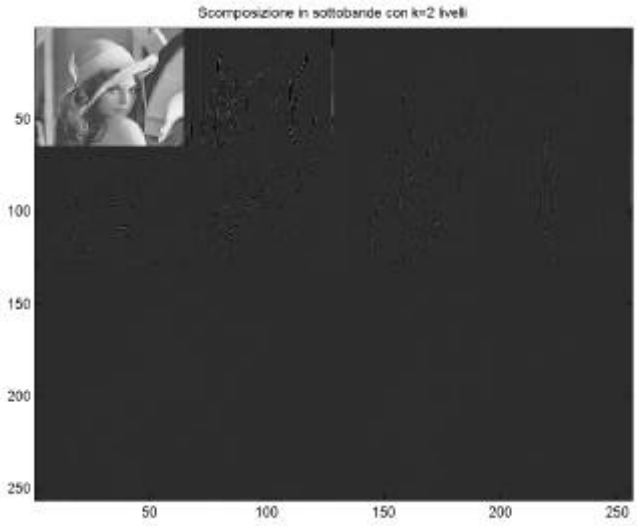


Figura 2.17.

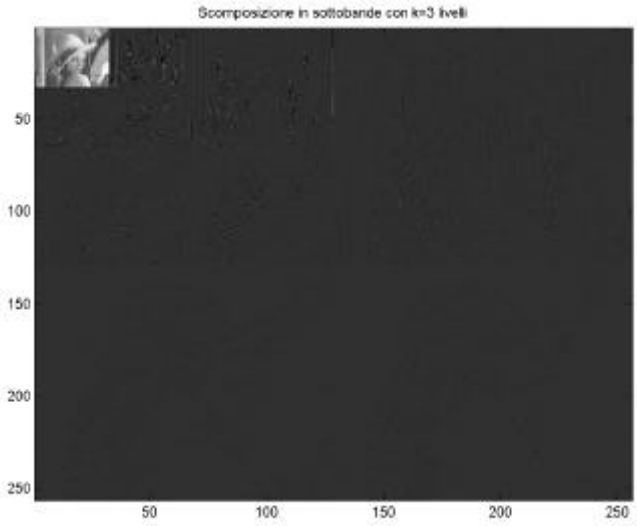


Figura 2.18.

Banco di Villasenor 4;



Figura 2.19.

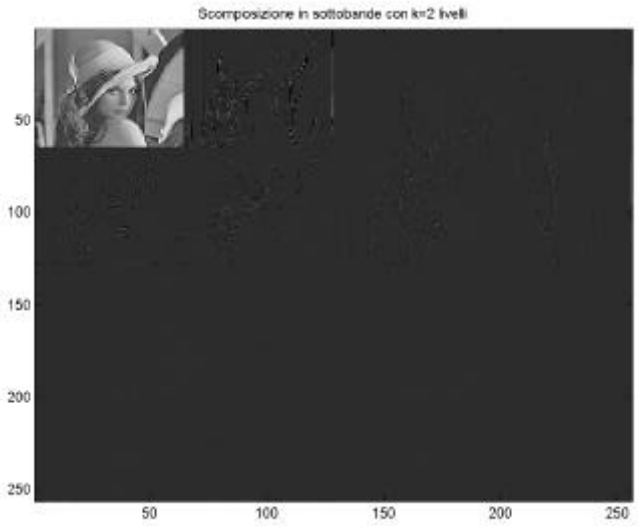


Figura 2.20.

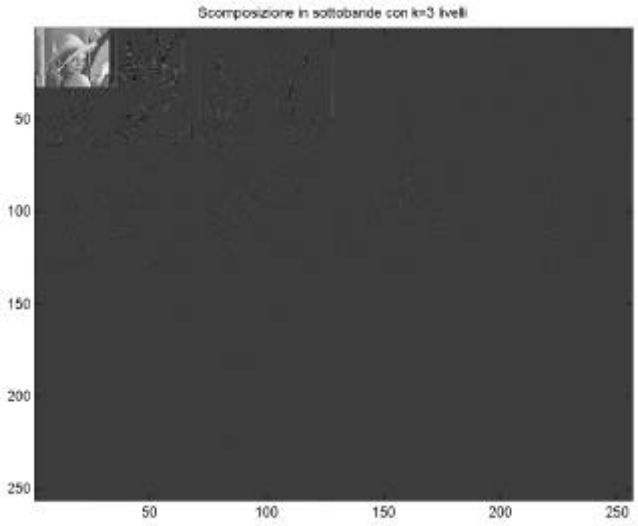
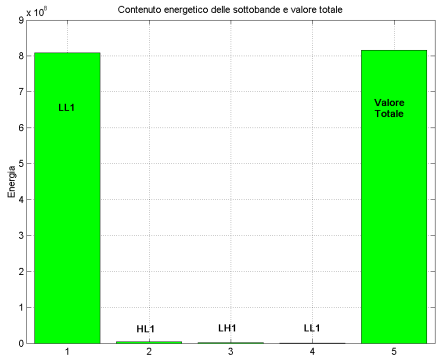


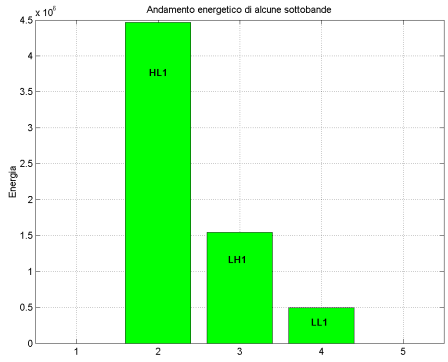
Figura 2.21.

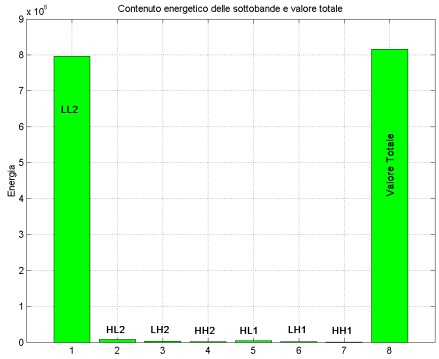
Ciò che si può dire riguardo questi grafici è che rende visibile la procedura di scomposizione in sottobande con l'immagine approssimazione sempre più piccola è meno dettagliata; infatti, per quanto si può, si nota che l'immagine approssimazione all'aumentare del livello di scomposizione ha i contorni sempre meno dettagliati e se poniamo attenzione tali informazioni sono contenute nei vari livelli di dettaglio sottoforma di variazioni di nero. Questo è lecito in quanto in corrispondenza dei contorni, quindi dei contrasti di luminanza abbiamo il contenuto in frequenza più alto (analogia con i segnali con ripide variazioni).

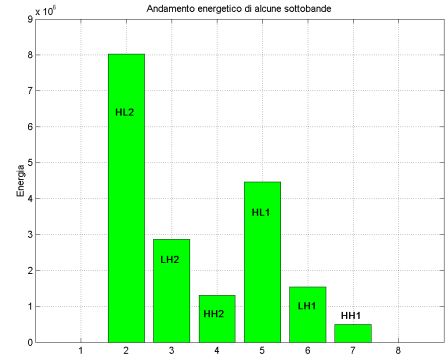
Come per il segnale monodimensionale abbiamo provato a calcolare l'energia contenuta nelle varie sottobande e a farne degli istogrammi che aiutano facilmente a intuire qual'è la potenzialità del banco di filtri a scomporre il segnale in sottobande.

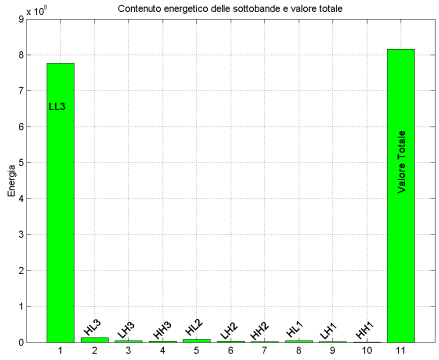
Banco di Haar;

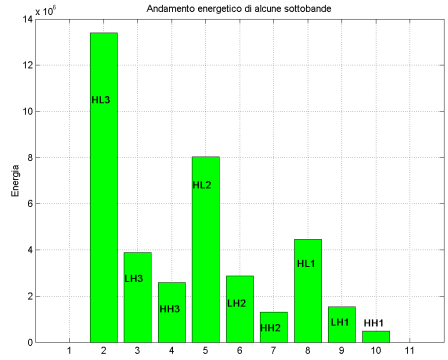




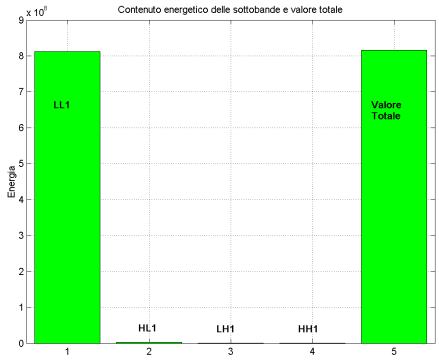


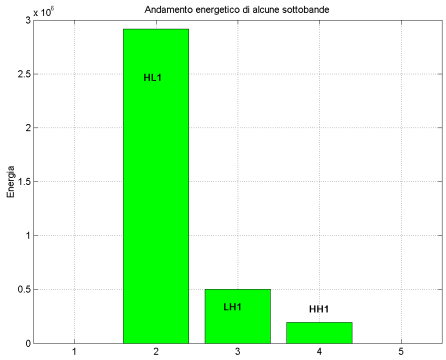


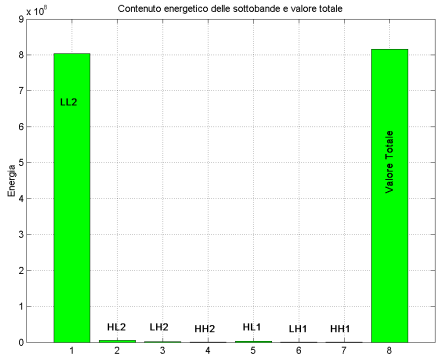


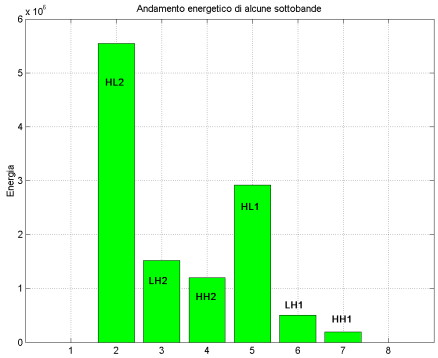


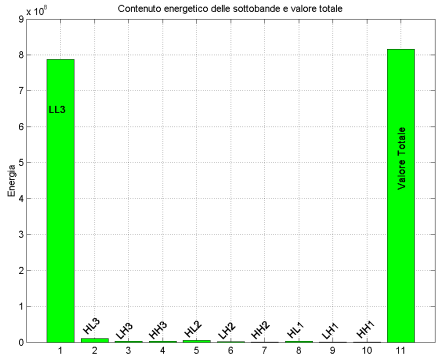
Banco di Daubechies (10);

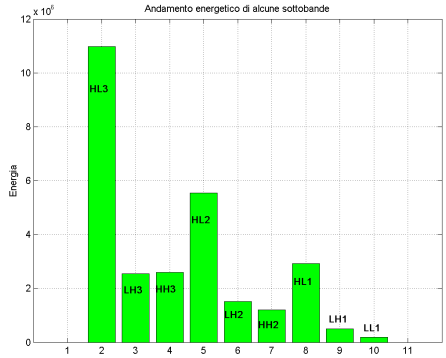




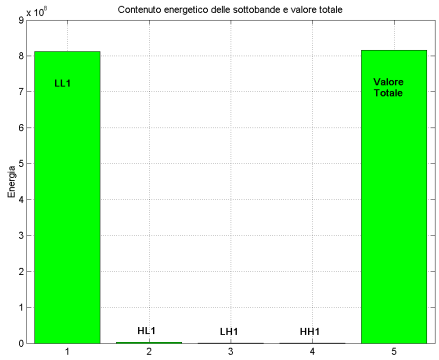


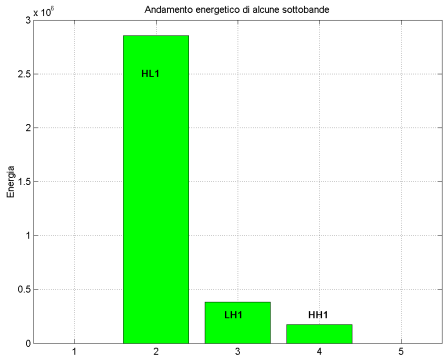


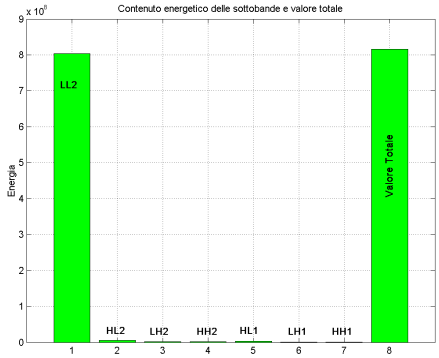


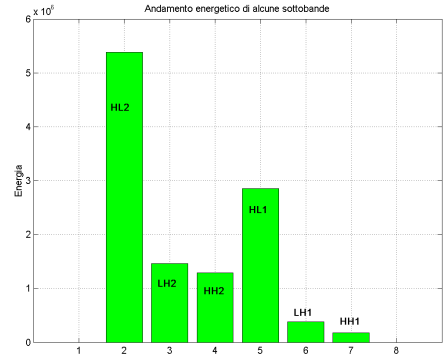


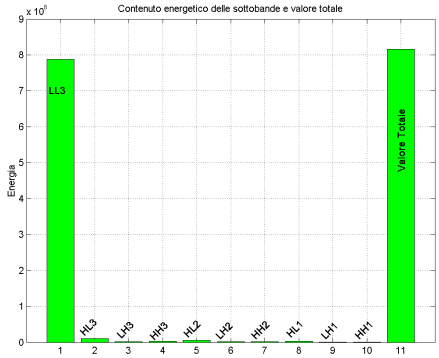
Banco di Daubechies (20);

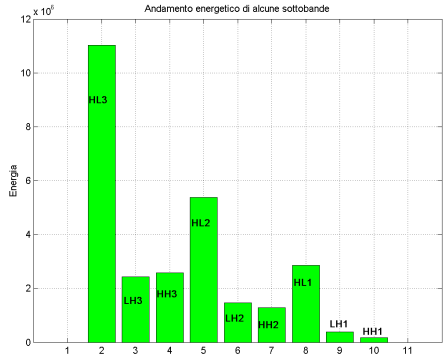




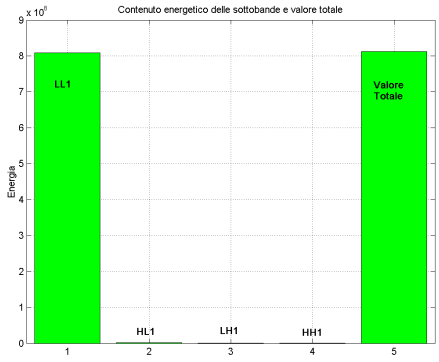


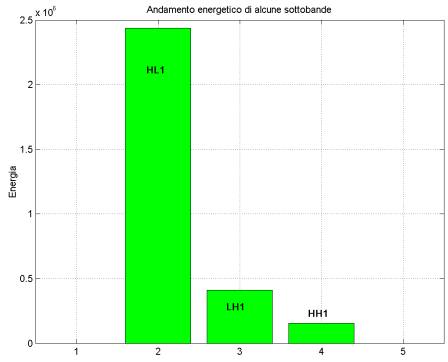


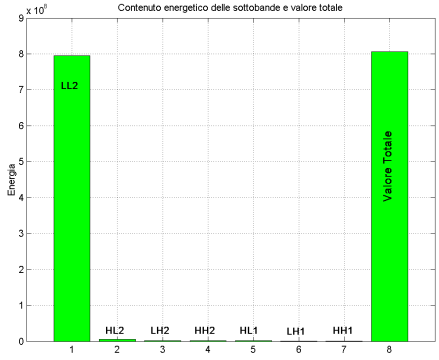


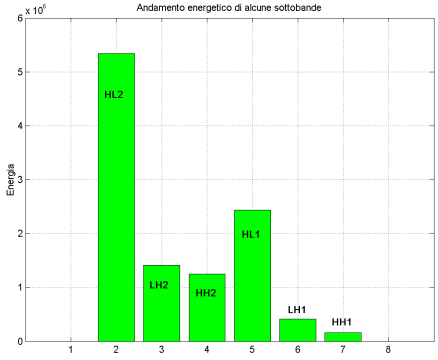


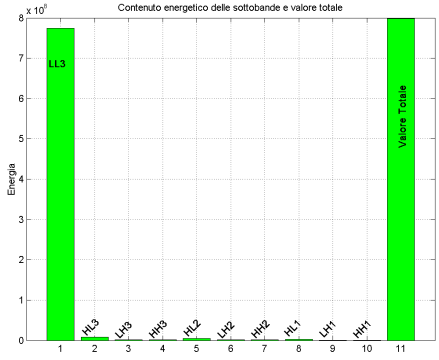
Banco di Villasenor 1;

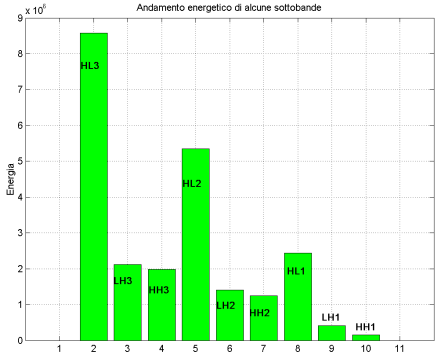




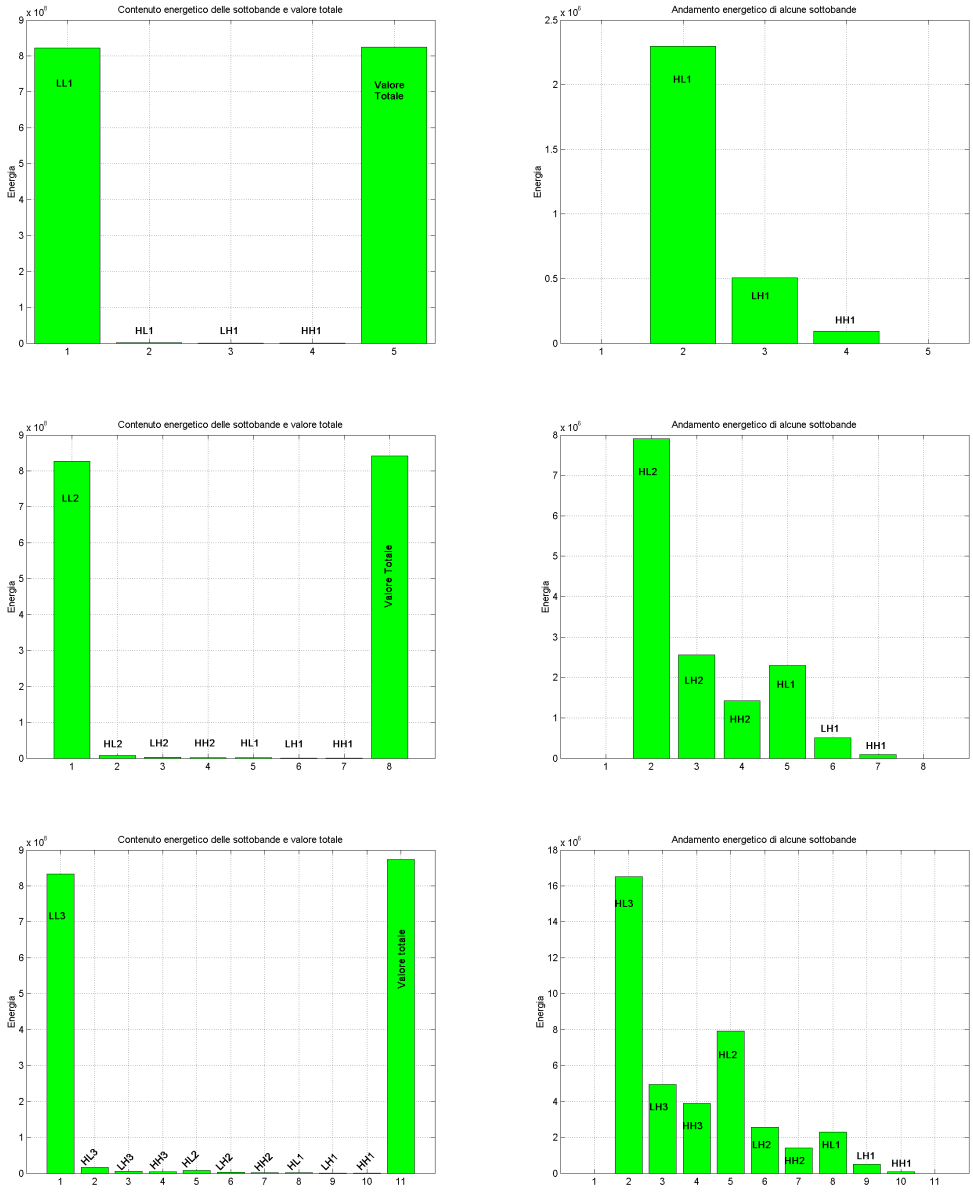








Banco di Villasenor 4;



Questi grafici ci permettono di verificare che il contenuto energetico delle sottobande di dettaglio è molto ridotto rispetto al contenuto della sottobanda di approssimazione, particolarmente evidente nei grafici (posti alla sinistra) in cui vi è il confronto diretto con il valore energetico dell'intera immagine e lo sbalzo di ordine di grandezza con le sottobande. Con i grafici (posti alla destra) che rappresentano esclusivamente il contenuto energetico delle sottobande di dettaglio è possibile evidenziare la differenza che si ha con i vari

banchi di filtri considerati; la scelta del banco da utilizzare potrà ricadere su uno piuttosto che un'altro in base alle necessità di elaborazione e alla qualità preferita, che rimane in ogni caso caratteristica soggettiva. Diciamo che fino a che non si effettuano alterazioni delle sottobande prima della ricostruzione non si hanno effetti che portino a preferire un banco rispetto ad un'altro.

Proprio a questo scopo abbiamo provato a valutare l'effetto che si ottiene sulla ricostruzione dell'immagine azzerando il contributo delle sottobande. In corrispondenza abbiamo valutato l'Errore Quadratico Medio (MSE) e il Rapporto Segnale/Rumore di Picco in dB ($PSNR_{dB}$), verificando l'aumento del primo man mano che aumentiamo il livello di scomposizione (e quindi le informazioni azzerate) e il diminuire del secondo in corrispondenza. Questi valori, di cui riportiamo anche in questo caso degli istogrammi che presentano visivamente l'andamento nei vari casi, possono essere sicuramente utili nella valutazione e scelta di un banco di filtri piuttosto che un altro, ma sulla qualità dell'immagine ottenuta niente può fare l'impressione che ognuno di noi ha e sceglire in funzione di questo.

Banco di Haar;



Figura 2.22.



Figura 2.23.

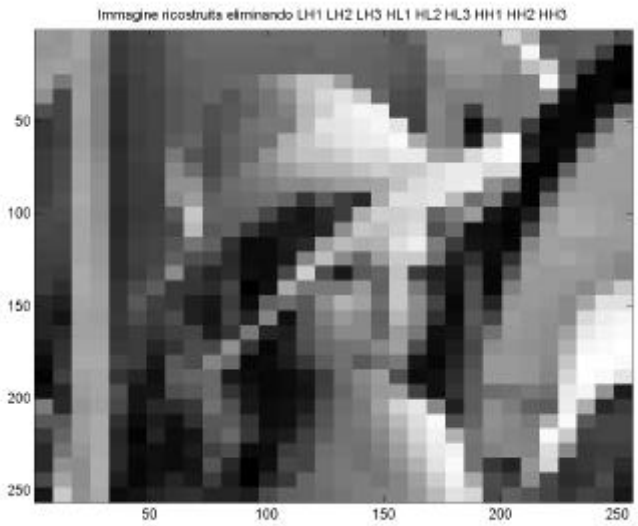
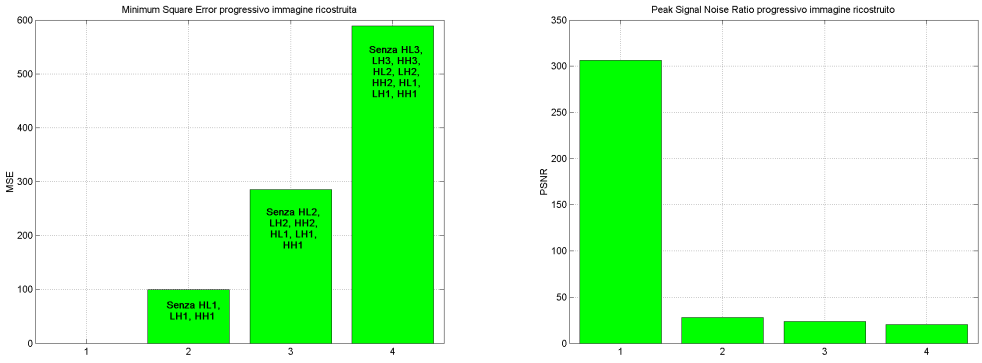


Figura 2.24.



Banco di Daubechies (10);



Figura 2.25.



Figura 2.26.

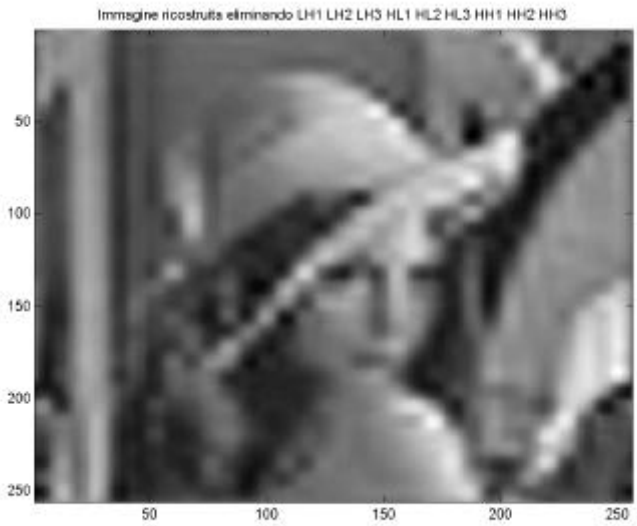
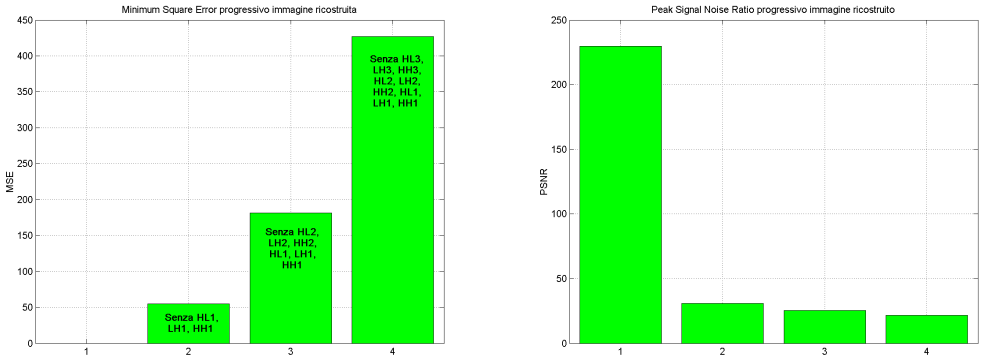


Figura 2.27.



Banco di Daubechies (20);



Figura 2.28.



Figura 2.29.

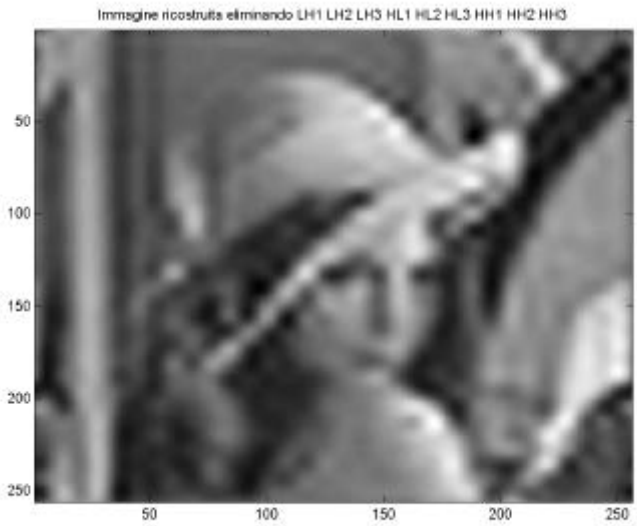
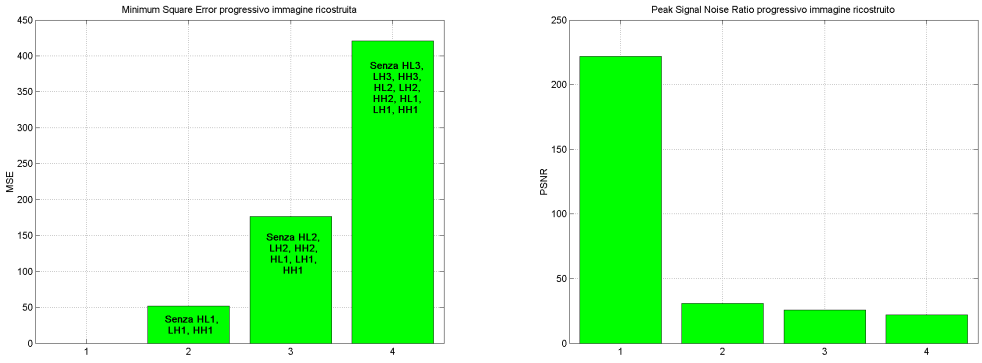


Figura 2.30.



Banco di Villasenor 1;



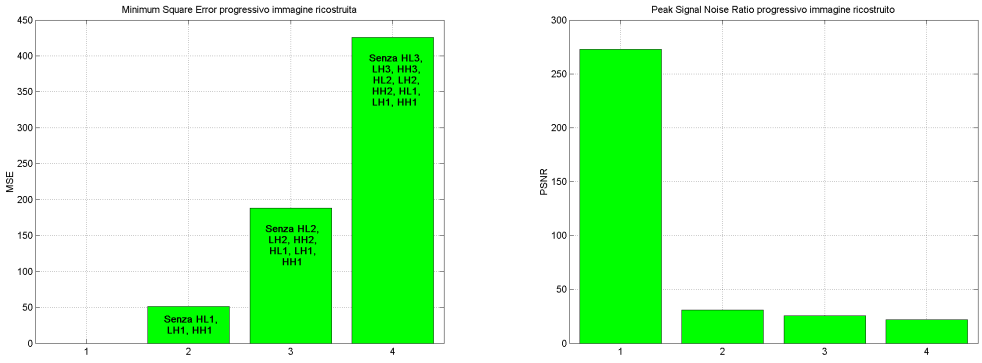
Figura 2.31.



Figura 2.32.



Figura 2.33.



Banco di Villasenor 4;



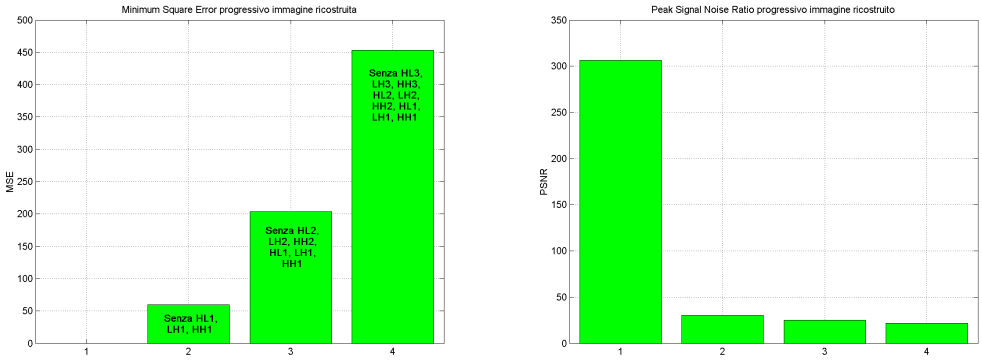
Figura 2.34.



Figura 2.35.



Figura 2.36.



Oltre ad un ovvio peggioramento dell'immagine ricostruita man mano che si eliminano contributi ad alta frequenza comune per tutti i banchi utilizzati, possiamo notare che nel caso del banco di Haar la degradazione dell'immagine è maggiore che negli altri casi dove comunque si riesce ad intuire l'immagine originaria, ed è verificabile anche dal grafico del MSE.

Per quanto riguarda il trattamento con filtri biortogonali può essere interessante vedere l'influenza che si ha sulla ricostruzione dell'immagine invertendo i due banchi di analisi e sintesi. Riportiamo i risultati ottenuti:



Figura 2.37.



L'immagine è ancora molto nitida ma presenta delle differenze, non tanto nei contrasti ma nella luce che caratterizza l'intera immagine; invertendo i banchi si ottiene un'immagine più opaca, in tal modo la distinzione tra le varie sezioni dell'immagine è leggermente compromessa ma non come prima sottraendo componenti ad alta frequenza. Tutto ciò è dovuto alla differenza della risposta all'impulso dei due banchi con un dominio temporale differente e quindi anche in frequenza e ciò porta a un trattamento del segnale che si ripercuote sulla ricostruzione. Si vedrà che il comportamento differente dei due banchi è particolarmente sensibile nel momento in cui si effettua una manipolazione dell'immagine (compressione d'immagine).

All'inizio di questo paragrafo abbiamo enunciato i comandi Matlab utilizzati per effettuare la trasformata wavelet e la relativa inversa. Una delle caratteristiche della trasformazione che fino ad ora abbiamo utilizzato era il trattamento del segnale in ingresso mediante periodicizzazione del segnale stesso, in tal modo i coefficienti dei bordi dell'immagine vengono trattati avendo pochissima correlazione con con i vicini; questi ultimi sono quelli del bordo opposto periodicizzato e così tra i due segnali uguali vi è una discontinuità. Attraverso il comando FWT2_SBS, letteralmente Trasformata Wavelet Diretta bidimensionale con Estesione Simmetrica Biortogonale, si manipola in modo differente proprio i bordi dell'immagine, non periodicizzando il segnale ma riportando lo stesso segnale speculare al bordo stesso (simmetrico per l'appunto); in questo modo tra i due segnali non vi è una discontinuità ma un punto di flesso e i coefficienti che descrivono il segnale nel bordo sono tra loro correlati.

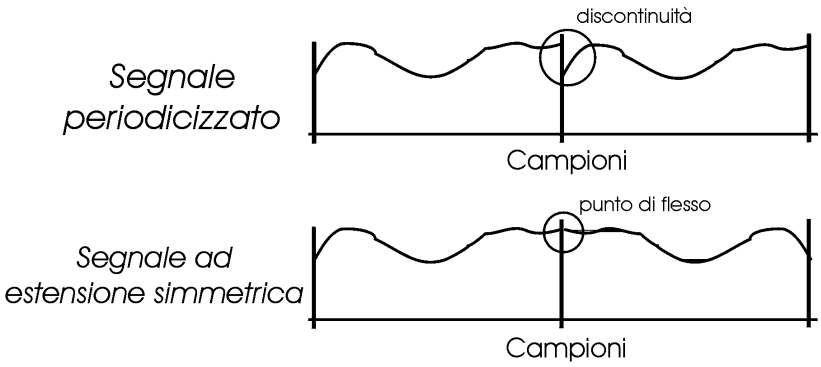


Figura 2.38.

Riportiamo di seguito i risultati ottenuti con trattamento ad estensione simmetrica che si possono confrontare con quelli ottenuti tramite periodicizzazione.



Figura 2.39.

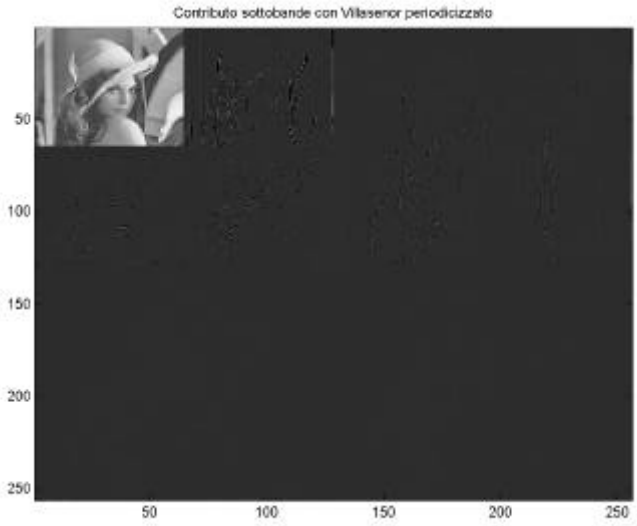


Figura 2.40.

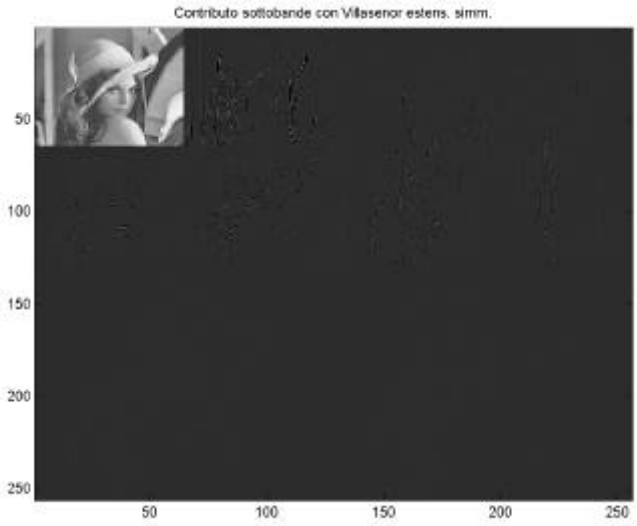


Figura 2.41.

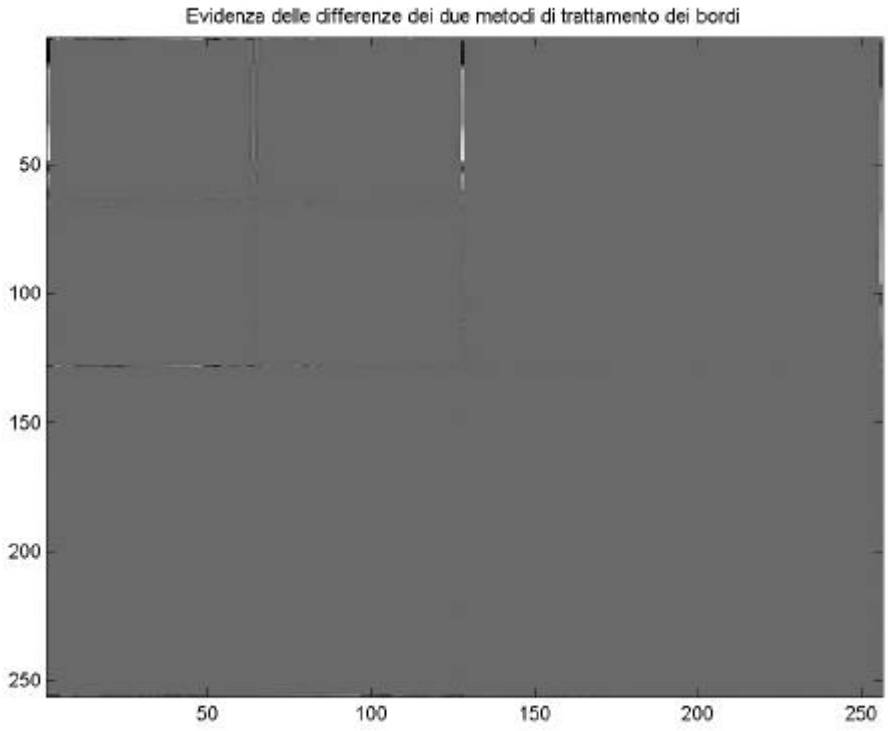


Figura 2.42.

Dai due grafici precedenti e soprattutto da quest'ultimo notiamo che la differenza, come si voleva dimostrare, si limita ai bordi nelle varie sottobande; con il metodo dell'estensione simmetrica il contenuto ad alta frequenza del bordo viene mantenuto dal livello approssimazione e quindi non lo ritroviamo nelle sottobande di dettaglio. L'effetto sulla ricostruzione è di un'immagine identica rispetto all'originale, d'altronde stiamo adoperando banchi a perfetta ricostruzione, ma il maggior contenuto ad alta frequenza dovute alle discontinuità indotte dall'estensione periodica può presentare dei problemi quendo l'immagine può essere oggetto di manipolazione (ad esempio compressione)..

2.3 Seconda fase sperimentale

Come per il primo capitolo abbiamo cercato di produrre un'immagine caratteristica, al fine di applicare le procedure viste nel paragrafo precedente e approfondire così le considerazione fino ad ora effettuate. L'immagine considerata è stata prodotta artificialmente mediante comandi Matlab come matrice valori numerici; la successione particolare di tali coefficienti abbiamo fatto in modo che ci produca una scacchiera in bianco e nero che riportiamo di seguito.

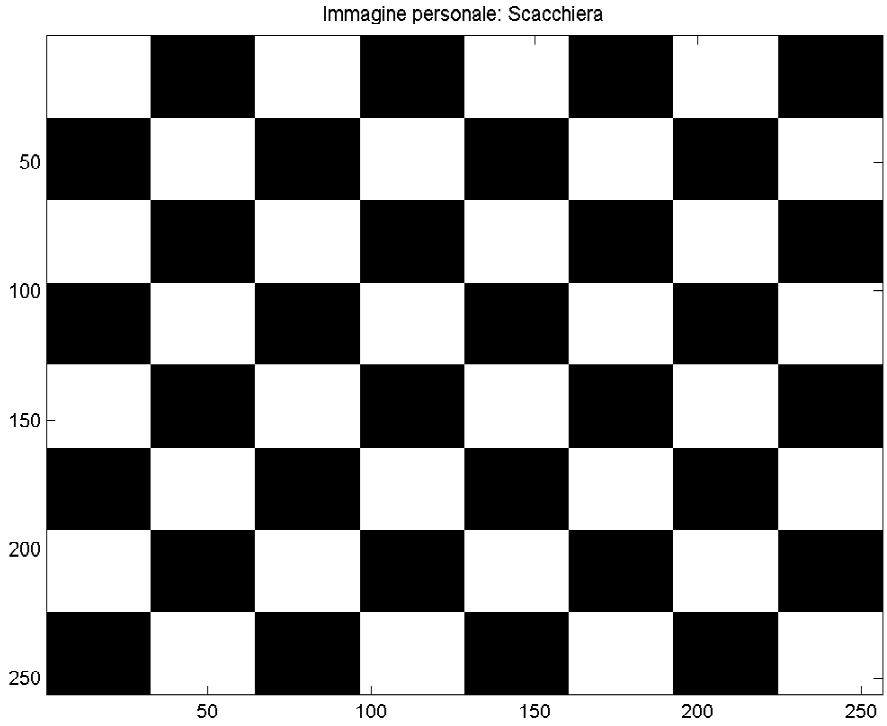


Figura 2.43.

Tutte le prove effettuate prima con l'immagine "Lenna" verranno applicate a questa scacchiera, commentando i risultati quando considerati particolarmente interessanti. Verranno usati esclusivamente i banchi di Haar, Daubechies (10) e Villasenor 1.

2.3.1 Scomposizione in sottobande con 1, 2 e livelli

Banco di Haar;

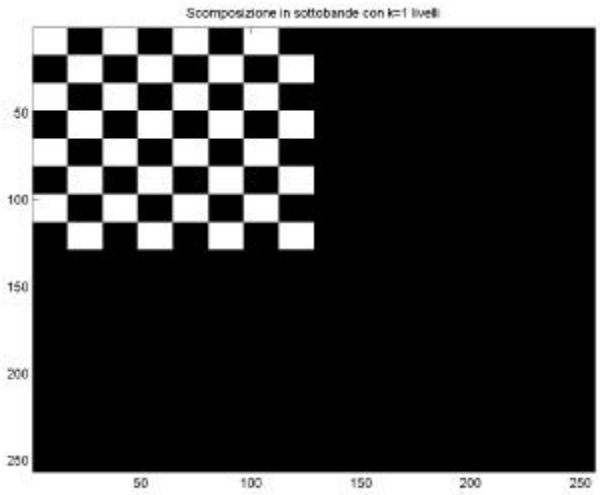


Figura 2.44.

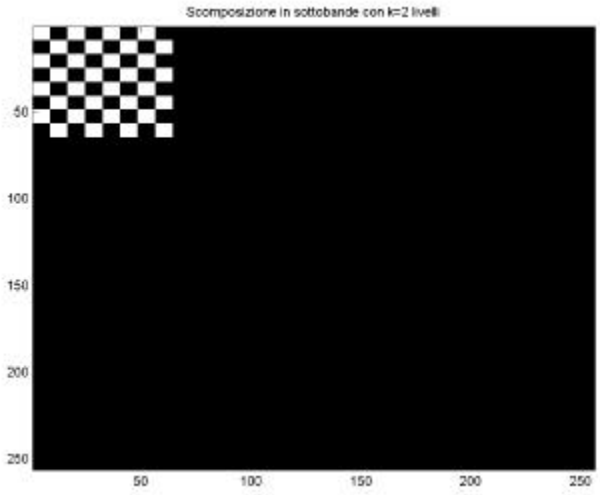


Figura 2.45.

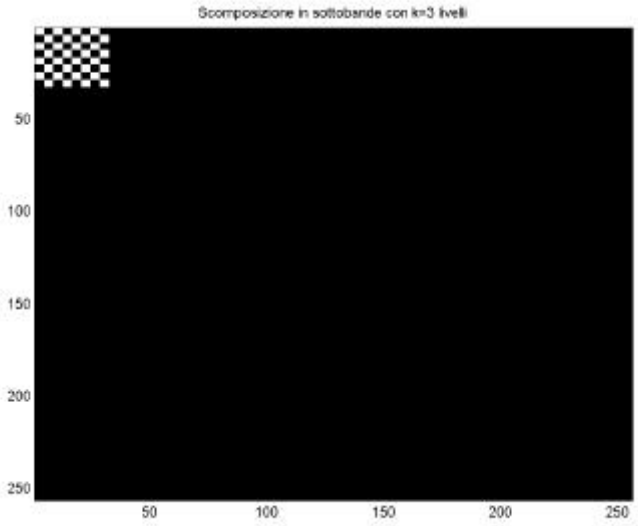


Figura 2.46.

Banco di Daubechies (10);

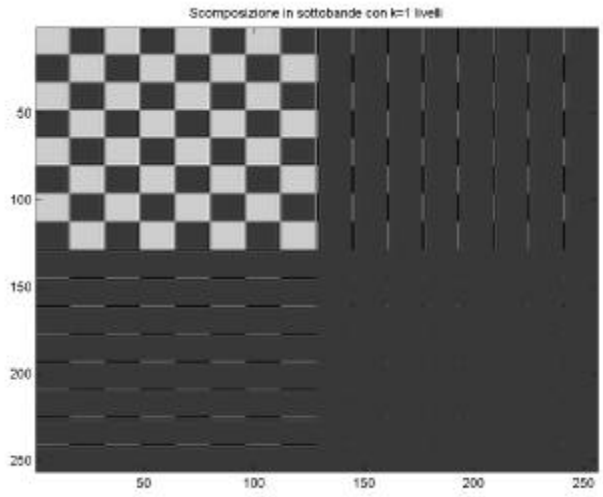


Figura 2.47.

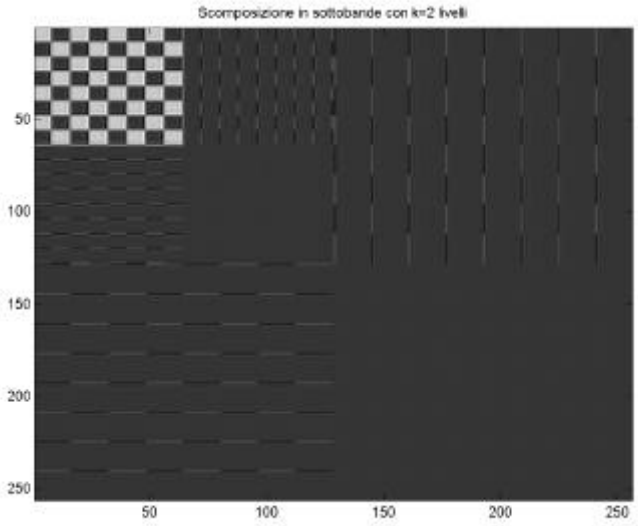


Figura 2.48.

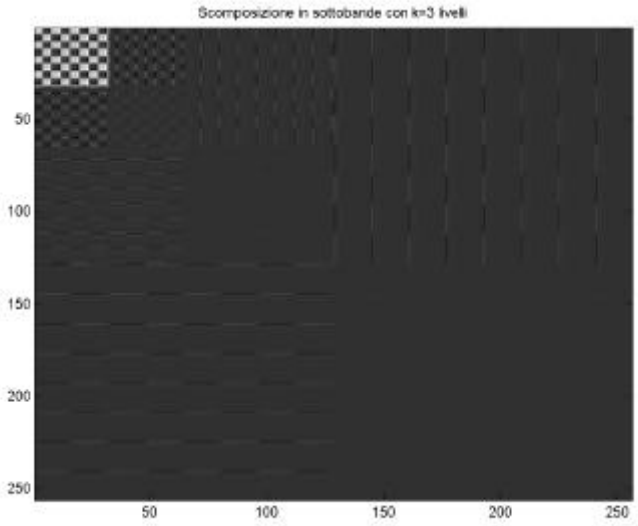


Figura 2.49.

Banco di Villasenor 1;

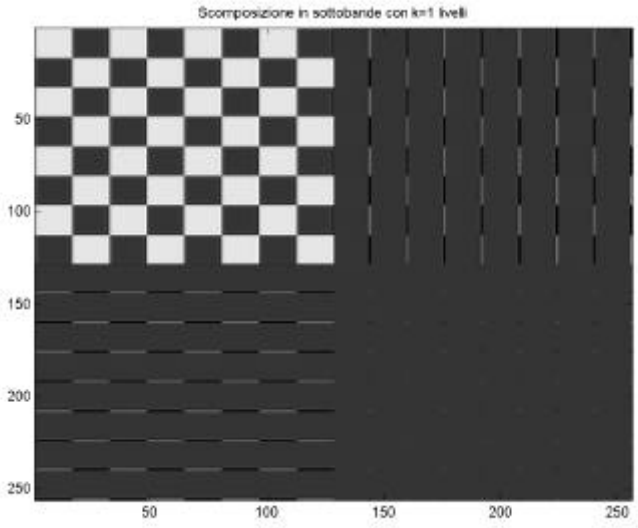


Figura 2.50.

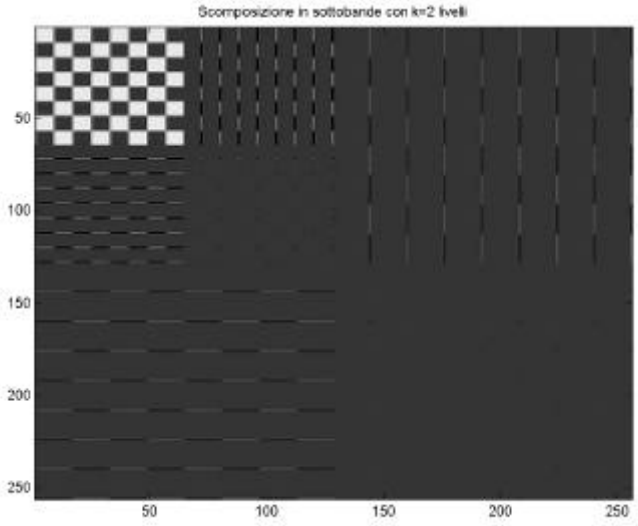


Figura 2.51.

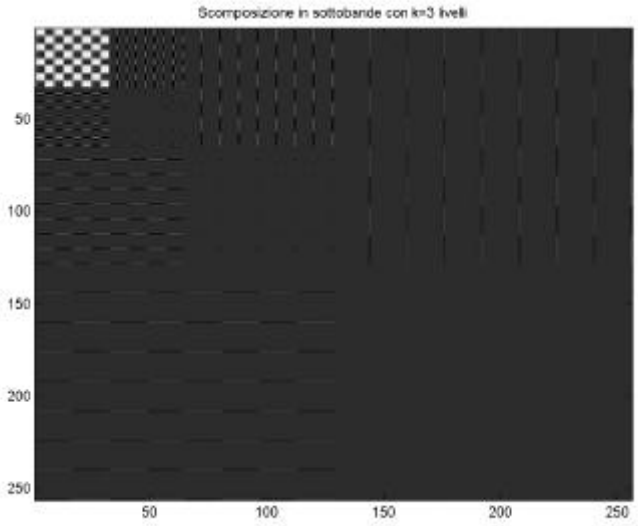
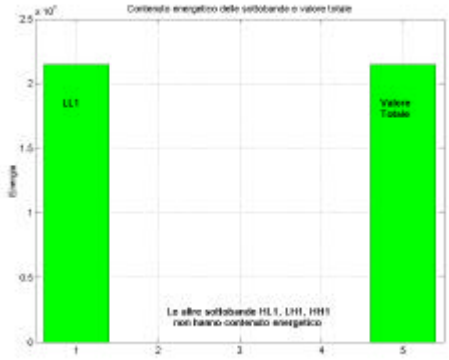


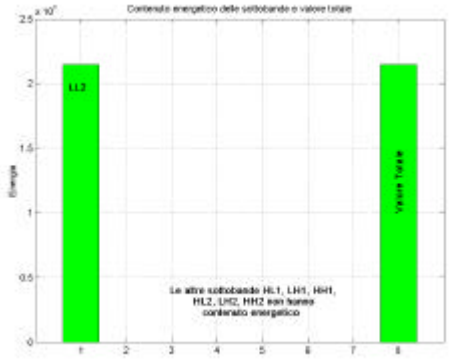
Figura 2.52.

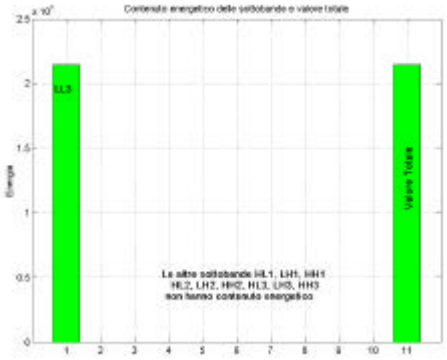
Da questi grafici si può notare che con il banco di Haar i livelli di dettaglio sono nulli; ritroviamo in formato bidimensionale la situazione che avevamo con il segnale monodimensionale di sequenze di porte per cui si addiceva particolarmente il banco di Haar. Con gli altri banchi, in maniera più o meno evidente, notiamo che con i livelli di dettaglio riusciamo ad evidenziare e catturare le discontinuità verticali con i livelli del tipo HL, mentre le discontinuità orizzontali con i livelli del tipo LH; per deduzione le discontinuità diagonali saranno catturate nei livelli di tipo HH (qui praticamente assenti).

2.3.2 Valutazione contenuto energetico

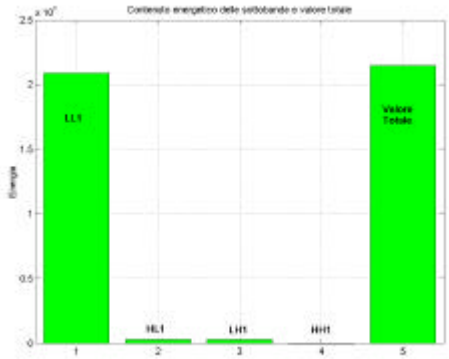
Banco di Haar;

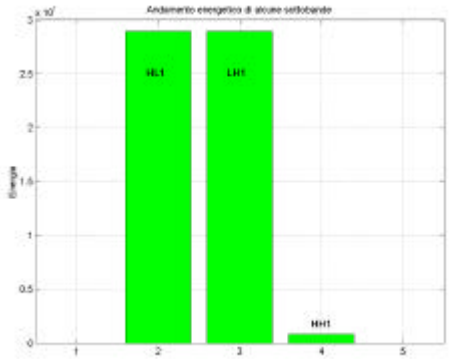


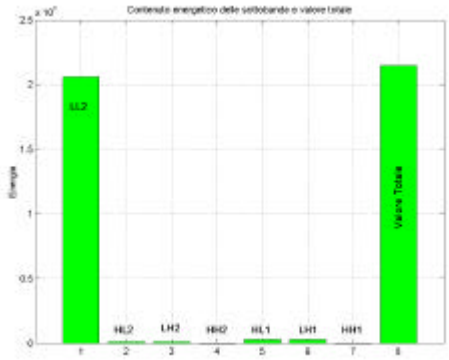


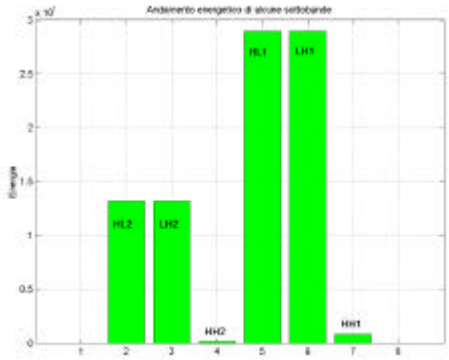


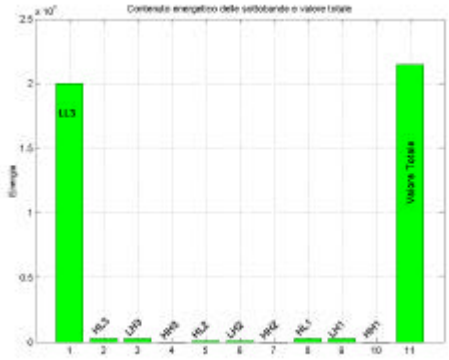
Banco di Daubechies (10);

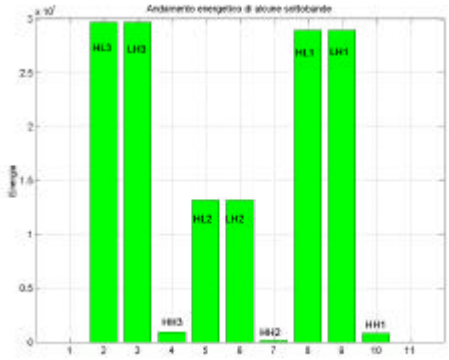




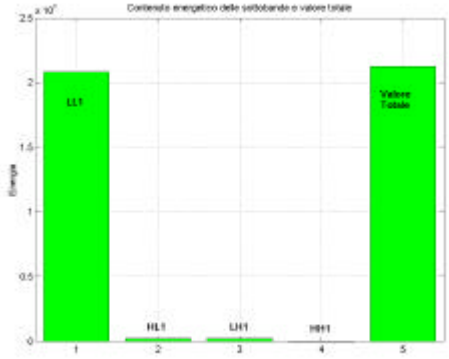


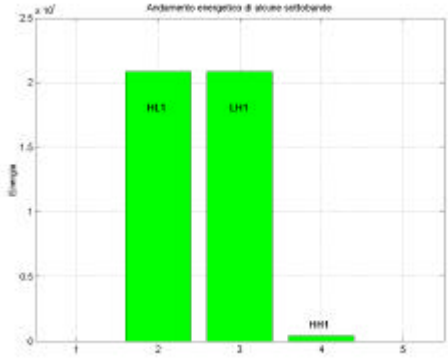


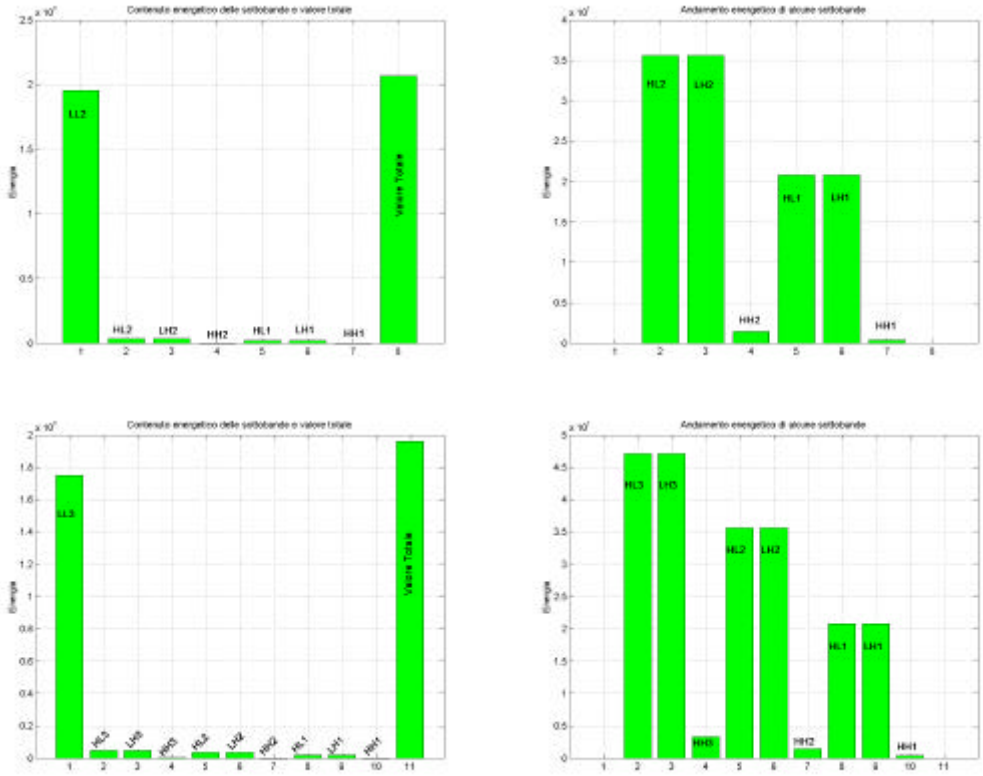




Banco di Villasenor 1;



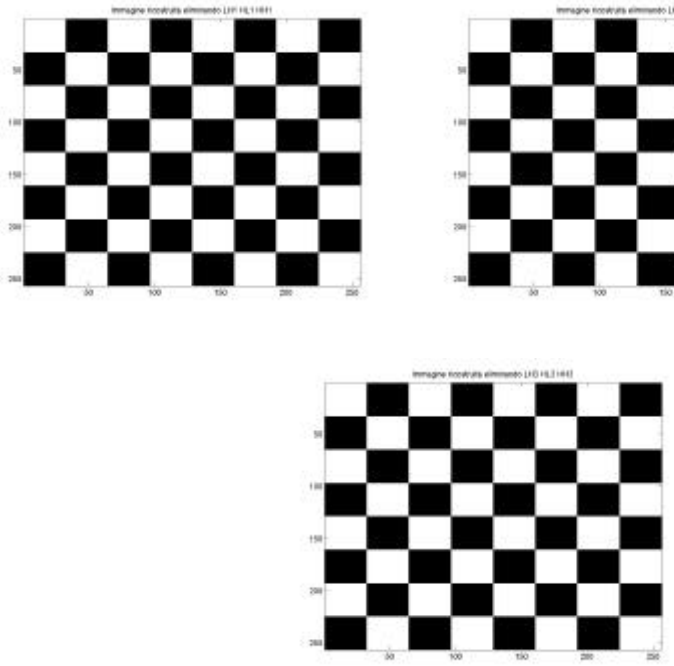




In questi grafici vengono, una volta in più, confermate le deduzioni precedenti relative al contenuto d'informazione, infatti le sottobande del tipo HL e LH hanno un contenuto consistente e paritario, visto tra l'altro il tipo di segnale, mentre il contenuto delle sottobande del tipo HH è irrilevante anche se presente. Tuttociò ovviamente ottenuto dai banchi di Daubechies e Villasenor perché come nel primo capitolo il banco di Haar è particolarmente adatto a questo segnale e con questi grafici verifichiamo la completa assenza di alcun contenuto nelle sottobande di dettaglio. Un ulteriore scomposizione avrebbe probabilmente permesso di trovare metà del contenuto energetico nel livello di approssimazione e l'altra metà in un livello di dettaglio (in particolare un livello HL e LH). Si noti anche la progressione decrescente del contenuto energetico dei livelli di dettaglio per il banco di Villasenor, cosa che non si verifica col banco di Daibechies.

2.3.3 Ricostruzione con azzeramento di sottobande

Banco di Haar;



Banco di Daubechies (10);

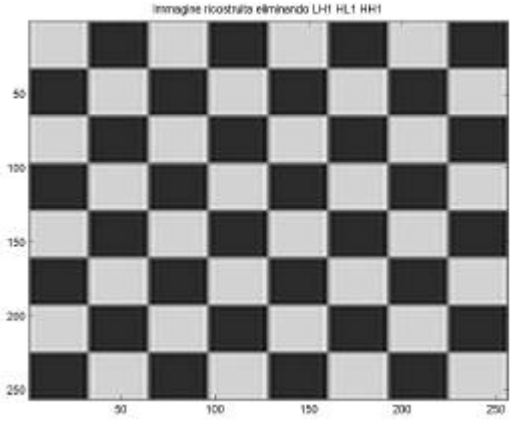


Figura 2.53.

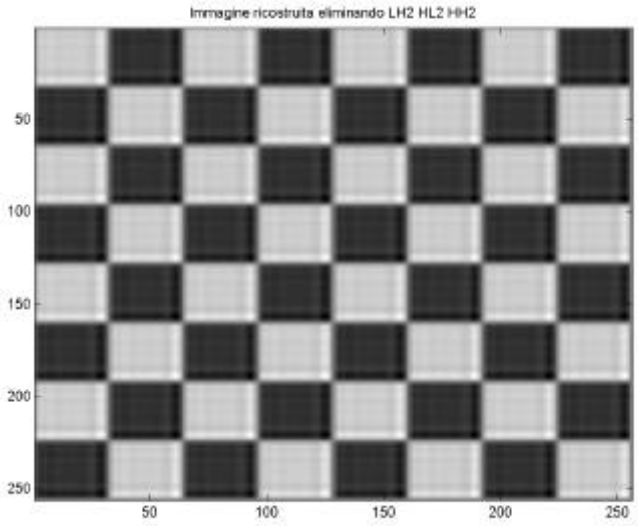


Figura 2.54.

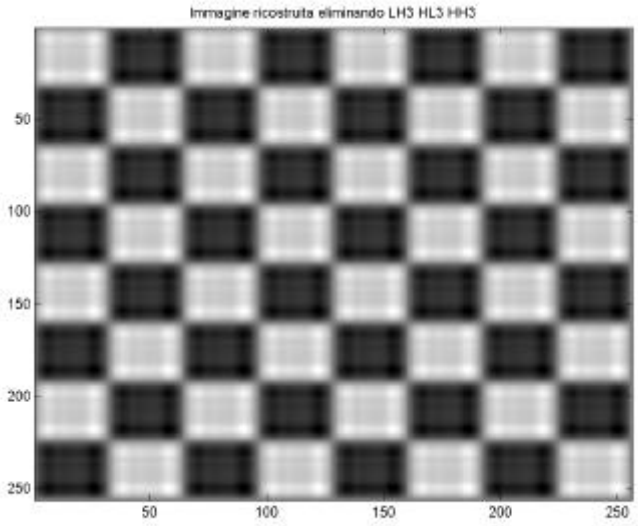


Figura 2.55.

Banco di Villasenor;

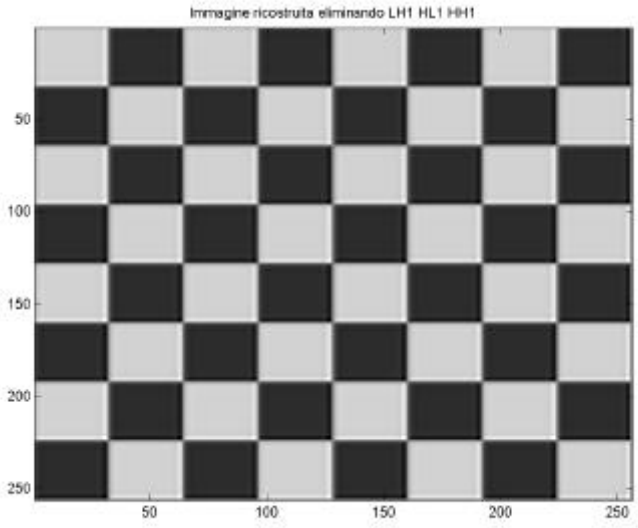


Figura 2.56.

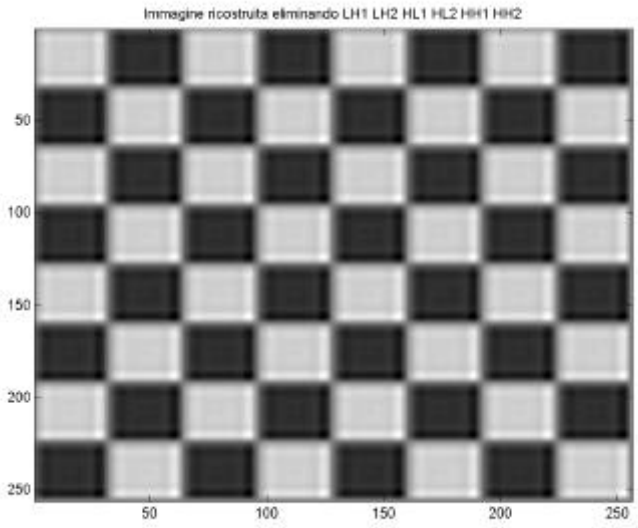


Figura 2.57.

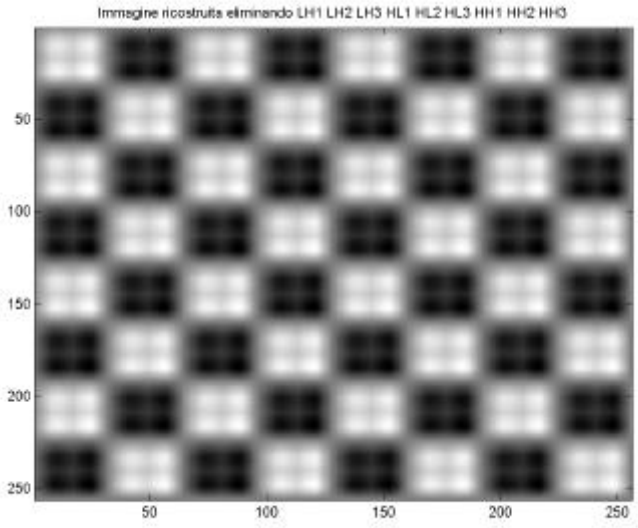
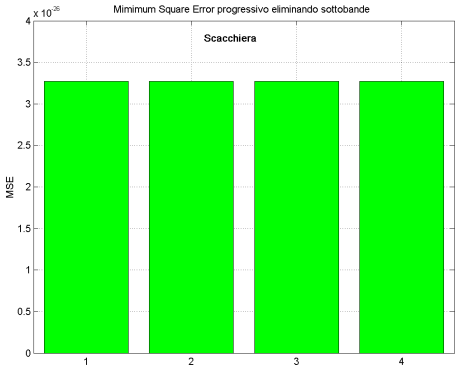


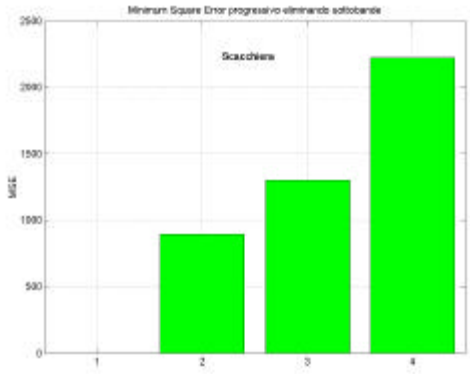
Figura 2.58.

Nel caso del banco di Haar la ricostruzione, nonostante l'azzeramento delle sottobande di dettaglio progressivo, è perfetta; negli altri casi, come ci si poteva aspettare, la ricostruzione è sempre più affetta da errore.

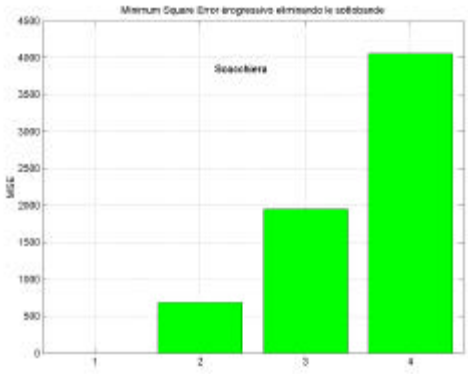
2.3.4 Valutazione del MSE e del PSNR



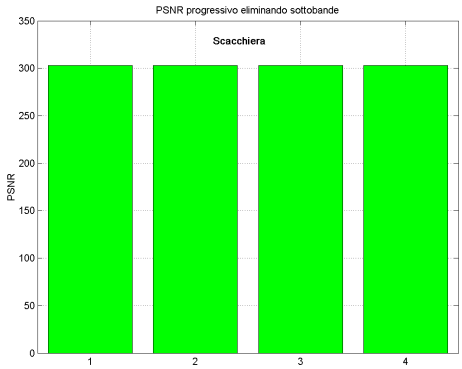
MSE progressivo con banco di Haar



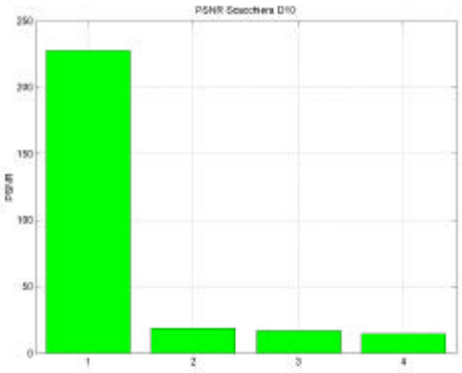
MSE progressivo con banco Daubechies



MSE progressivo con banco Villasenor



PSNR progressivo con banco di Haar



PSNR progressivo con banco di Daubechies



PSNR progressivo con banco di Villasenor

Da questi grafici notiamo la già ribadita caratteristica del banco di Haar su quest'immagine (notare l'ordine di grandezza del grafico MSE relativo),

ma in più notiamo che per tale immagine il banco di Villasenor con sottobande soppresse presenta un MSE minore nei vari casi, anche se poi la qualità dell'immagine può comunque non essere soddisfacente.

2.3.5 Scambio dei banchi e bordi con estensione simmetrica

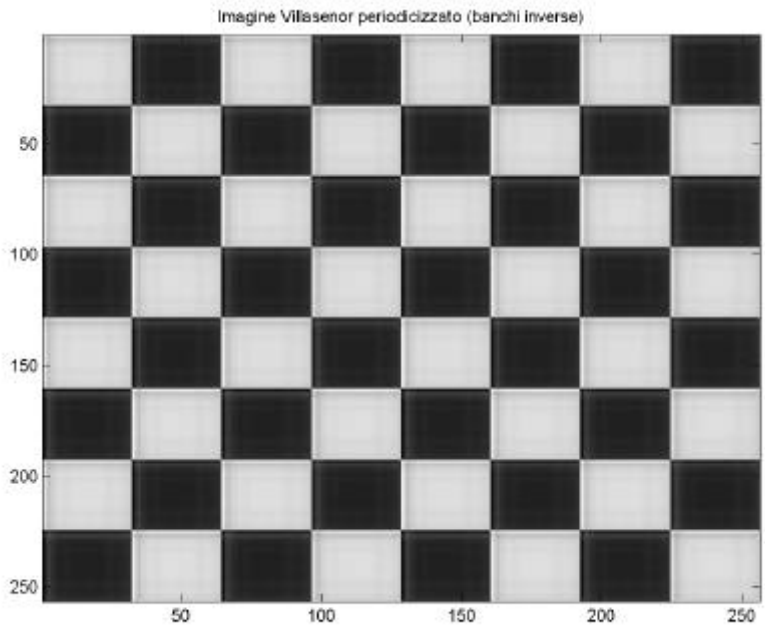


Figura 2.59.

Commenti analoghi, se non più evidenti, a quelli fatti per Lenna.

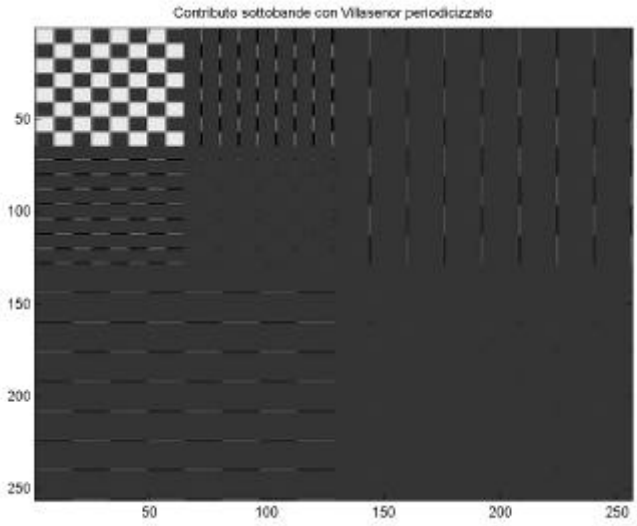


Figura 2.60.

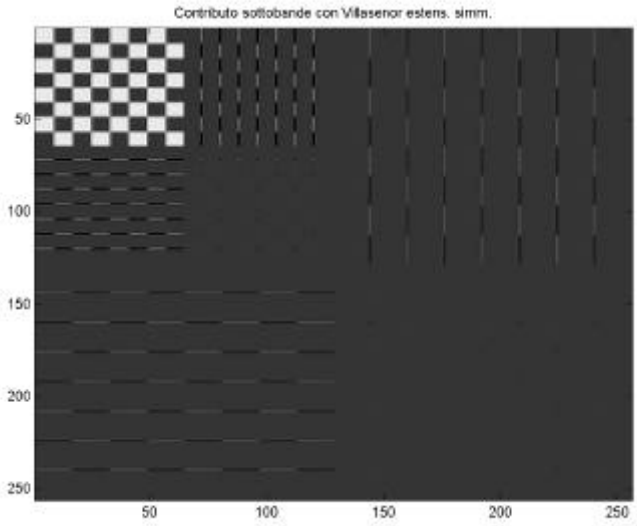


Figura 2.61.

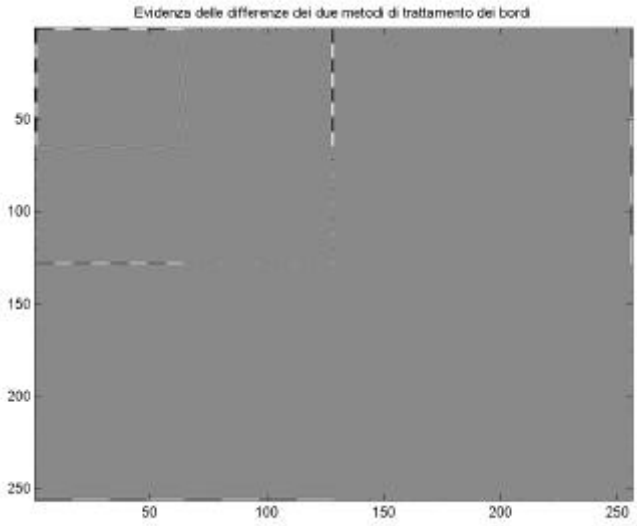


Figura 2.62.

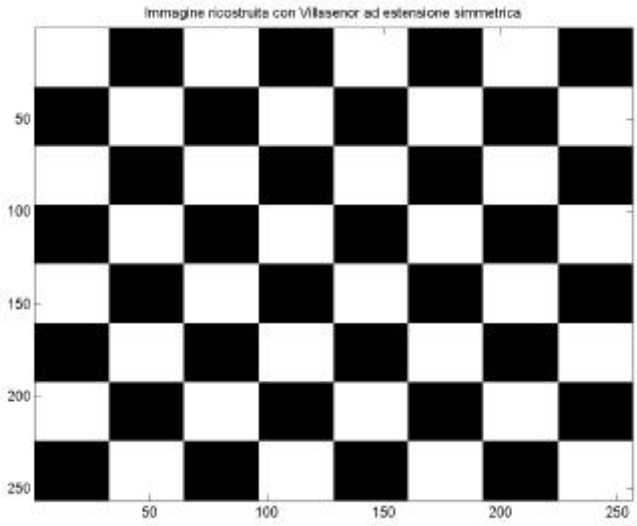


Figura 2.63.

Capitolo 3

Compressione di immagini mediante la DWT

3.1 Introduzione

La compressione delle immagini è una delle applicazioni principali del signal processing in quanto rende possibile la memorizzazione e la trasmissione di queste, contenendo l'occupazione in memoria e il bit rate, sfruttando la correlazione che sussiste dei campioni che le rappresentano. Per fare un esempio, un'immagine a colori di dimensioni 512×512 pixel di lato, codificata su 24 bit , richiede circa 6Mbit di memoria e un rate di circa 200Mbps per la trasmissione televisiva standard. Si tratta di numeri proibitivi se si pensa ad applicazioni basate sulla trasmisione su canale radiomobile. Esistono numerose tecniche di compressione che permettono di ridurre notevolmente tali richieste;tute si basano sulla stessa idea di fondo: applicare all'immagine una trasformazione lineare che ne scorreli i campioni, in modo da compattare l'energia, e quindi l'informazione, sul minimo numero possibile di coefficienti nel dominio trasformato. Ciò significa che l'obiettivo è quello di rappresentare il segnale in un dominio in cui la maggior parte dei coefficienti sia prossima a zero. Ovviamente per assicurare il ripristino del segnale originale occorre che la trasformazione sia invertibile.

Attualmente, la trasformata coseno (DCT, Discrete Cosine Transform) occupa una posizione di rilievo in questo ambito, essendo utilizzata dagli standard JPEG e MPEG.

Applicata la trasformazione lineare, il punto di forza su cui si basa la riduzione del numero di bit richiesti per la rappresentazione del segnale consiste nell'operazione di *quantizzazione*. Questa operazione non fa altro che rappresentare i campioni con un numero di bit variabile in base alla loro

"rilevanza"; in altre parole, un campione prossimo a zero viene codificato con un numero di bit molto più piccolo di quello destinato a un campione rilevante: se il numero di campioni prossimo a zero è molto elevato è ovvio che si possono raggiungere livelli di compressione decisamente elevati.

La trasformata wavelet rappresenta in quest'ottica, una valida alternativa alla trasformata coseno. La ragione di tutto questo sta nel fatto che applicando la DWT ad una immagine, la decomposizione in sottobande fa si che molti dei coefficienti relativi alle bande di frequenza più alte siano prossimi a zero; pertanto quantizzando ciascuna sottobanda in modo indipendente,e utilizzando una quantizzazione a precisione variabile, si riescono a conseguire tassi di compressione alti.

Alla luce di tutto questo, i problemi trattati in questa sezione saranno verificare le prestazioni della DWT nel caso di una compressione con perdite. Essendo basata su di una decomposizione mediante banchi di filtri, essa definisce a differenza della DCT una classe di trasformate; pertanto le prestazioni veranno valutate anche al variare dei banchi. Infine verranno messi in luce quali sono i difetti che questa trasformata presenta.

3.2 DWT su immagini

Si è detto che applicando la DWT ad una immagine, molti dei coefficienti nel dominio trasformato sono nulli. Il motivo è facilmente spiegabile: la DWT fornisce una versione approssimata del segnale dal ramo passabasso del banco di filtri e una versione di dettaglio dal ramo passalto; la versione approssimata è molto "simile" all'originale e di conseguenza la versione di dettaglio è costituita principalmente da coefficienti molto prossimi a zero. Iterando la decomposizione sul ramo passabasso, si ottengono molte sottobande di dettaglio i cui coefficienti hanno una dinamica ridotta, e quindi richiedono pochi bit per essere rappresentati, ed inoltre una sola sottobanda di approssimazione che, dal punto di vista della percezione visiva, "assomiglia" molto all'immagine di partenza la quale verrà quantizzata con il massimo numero di bit disponibile, ma che è molto più piccola dell'immagine originale.

Ovviamente, più cresce il livello di decomposizione, maggiore diventa l'importanza delle sottobande di dettaglio al fine della ricostruzione e quindi superiore risulta il numero di bit ad esse destinate. Il criterio tipicamente utilizzato per stabilire il numero di bit da associare a una certa sottobanda consiste nel valutarne l'energia: più elevata è l'energia, e più raffinata deve risultare la quantizzazione. Chiaramente tutto è normalizzato rispetto al massimo numero di bit utilizzabile.

La soluzione spesso utilizzata per le immagini consiste nel decomporre

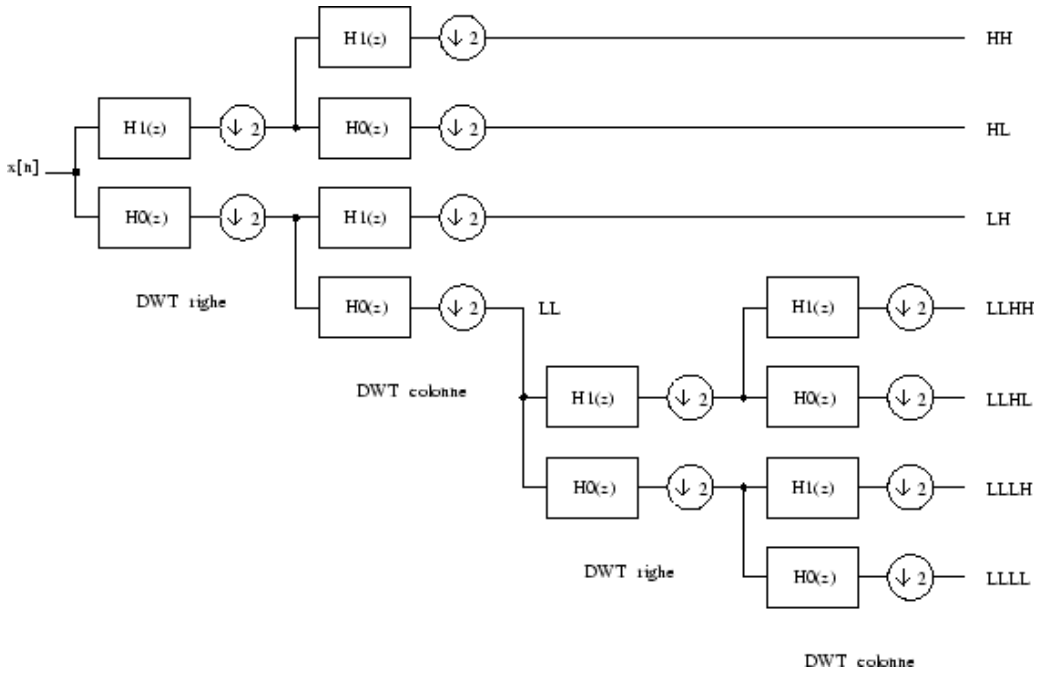


Figura 3.1. Suddivisione in due livelli di un'immagine

l'immagine per righe e per colonne alternativamente e riapplicando i filtri sulla sequenza passa-basso. Questa suddivisione in due operazioni distinte per le righe e per le colonne è possibile grazie al fatto che il filtro bidimensionale da applicare è separabile in due componenti e l'ordine di applicazione di queste non influisce sul risultato finale. Quindi per filtri bidimensionali la convoluzione bidimensionale si riduce all'applicazione di due convoluzioni successive. Per esempio possiamo applicare una decomposizione su due livelli (sia sulle righe che sulle colonne) ottenendo uno schema a blocchi del tipo mostrato in figura 3.1.

Graficamente invece si ottiene una mappa di coefficienti avente area pari all'immagine di partenza e distribuzione dei sottolivelli di decomposizione visibile in figura 3.2, ove il livello LLLL, corrispondente al secondo filtraggio (un filtraggio corrisponde in verità a due filtraggi uno sulle righe e uno sulle colonne) rappresenta la sottobanda di approssimazione e contiene l'immagine alla risoluzione minore. Le altre zone (LLHH, LLHL, LLLH) consentono di ricostruire il dettaglio per ottenere il livello LL, che conterrà l'immagine a una risoluzione intermedia. La banda LL, insieme a HH, HL e LH consente di risalire all'immagine originale.

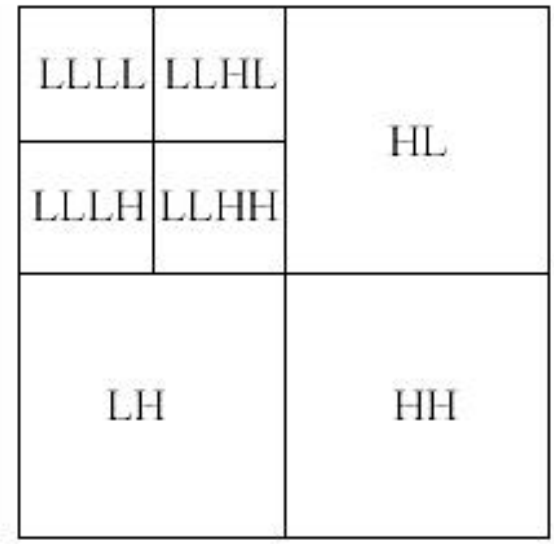


Figura 3.2. Posizione delle sottobande nella griglia dei coefficienti della WT

3.3 Descrizione dell'esperienza e risultati ottenuti

Lo scopo di questa esperienza è utilizzare la DWT per comprimere un immagine e valutarne quindi in base ai risultati ottenuti, le prestazioni. La trattazione sarà divisa in diverse fasi: una prima nella quale utilizzando una DWT che effettua una estesione simmetrica ai bordi e con diversi banchi di filtri biortogonali (Villasenor 1,2,3), si comprimerà un immagine per diversi valori di bit-rate; di questa se ne valuterà la qualità sia dal punto di vista soggettivo che oggettivo. La seconda fase prevederà la stessa elaborazione però con una DWT che effettua una estensione periodica dei bordi. Utilizzando poi il migliore banco di filtri biortogonale tra quelli trovati e l'estensione migliore tra le due, si comprimeranno alcune immagini con determinate caratteristiche. Infine si evidenzieranno i difetti di questa trasformata.

Per semplicità, il numero di livelli di decomposizione con il quale verrà eseguita la DWT è pari a due. Rimane ancora un punto fondamentale da affrontare: la modalità con la quale effettuare la compressione e quindi la quantizzazione. Una prima idea potrebbe essere quella di utilizzare un algoritmo entropico, che codifica l'informazione in base alle caratteristiche statistiche del segnale, associando ai campioni più probabili stringhe molto corte

e a quelli meno probabili stringhe più lunghe; questo genere di approccio, tuttavia, è tutt'altro che banale e richiede, inoltre, la memorizzazione della tabella di codifica per la successiva ricostruzione.

L'idea su cui ci si basa è la seguente: data la DWT di un'immagine, si vuole determinare il numero di bit su cui quantizzare ciascuna sottobanda, in modo che l'immagine ricostruita a partire dalla DWT quantizzata sia il più possibile simile all'originale (in termini di PSNR). L'algoritmo che ci permette questo è il seguente:
Si supponga di avere a disposizione la DWT dell'immagine, calcolata con L livelli di decomposizione. Quindi il numero totale di sottobande è $P = 3L+1$. Si fissi un bit-rate medio R su cui rappresentare l'immagine, o meglio, la sua DWT (occorre ricordare che il rate delle immagine non compresse prese in considerazione è 8 bpp). L'algoritmo funziona come segue:

1. Per ogni sottobanda ($i = 1, \ldots, P$) si calcoli la deviazione standard σ_i dei coefficienti della sottobanda

2. Si calcoli la media geometrica di queste deviazioni standard:

$$\rho = \left(\prod_{i=1}^{P} \sigma_i \right)^{\frac{1}{P}}$$

3. Il numero di bit ottimo su cui quantizzare ogni sottobanda è dato da:

$$R_i = R + \log_2 \frac{\sigma_i}{\rho}$$

4. Ovviamente i coefficienti R_i non sono interi, e quindi vanno arrotondati o troncati. Inoltre, R_i potrebbe essere negativo, nel qual caso deve essere posto a zero

5. È importante, alla fine dell'allocazione, calcolare qual'è il bit-rate medio ottenuto in pratica, e confrontarlo con R

Nota questo algoritmo si può utilizzare quando gli oggetti da quantizzare (nel nostro caso le sottobande) sono indipendenti, come avviene con buona approssimazione per la DWT. L'algoritmo è ottimo se questi oggetti hanno densità di probabilità gaussiana, ch nel caso della DWT è un assunzione accettabile.



Figura 3.3. Immagine originale: Lenna

3.3.1 Compressione mediante DWT con estensione simmetrica dei bordi

L'immagine presa in esame è "*Lenna*" di dimensioni 256×256 pixel codificata su 8 bpp (figura 3.3). Utilizzando tre banchi di filtri biortogonali rispettivamente Villasenor 1, 3 e 5 , si proverà a comprimere l'immagine ai seguenti bit-rate: 0.25, 0.5, 1, 2, 3, 4, 5 bpp. Il motivo per il quale si è scelto questi filtri è perchè mediamente hanno prestazioni superiori agli altri in quanto: hanno fase lineare, sono corti e regolari. Nelle tabelle che seguono sono riportati i risultati ottenuti in termini di bit-rate medio effettivamente ottenuto con l'allocazione, PSNR e errore quadratico medio.

Villasenor 1

Bit-Rate [bpp]	Bit-Rate effettivo [bpp]	PSNR [dB]	MSE
0.25	0.714	25.388	188.017
0.5	1.142	26.660	140.306
1	1.285	27.388	125.656
2	2	29.387	74.871
3	3	31.693	44.03
4	4	35.333	19.043
5	5	39.524	7.255

Villasenor 3

Bit-Rate [bpp]	Bit-Rate effettivo[bpp]	PSNR [dB]	MSE
0.25	0.8571	25.416	182.588
0.5	1.142	26.509	145.267
1	1.285	26.931	131.793
2	2	29.103	79.936
3	3	31.604	44.939
4	4	35.023	20.45
5	5	38.907	8.363

Villasenor 5

Bit-Rate [bpp]	Bit-Rate effettivo[bpp]	PSNR [dB]	MSE
0.25	0.714	25.167	188.017
0.5	0.857	25.168	140.306
1	1.285	26.588	125.656
2	2	28.588	74.871
3	3	31.123	44.03
4	4	34.860	19.043
5	5	39.155	7.255

Riportiamo in forma grafica gli stessi dati per avere una lettura più rapida e comprensibile (figura 3.4). Dai dati trovati, si può affermare che in ogni caso anche per valori di bit-rate bassi (0.25 bpp) l'immagine che si ottiene è comunque ancora comprensibile[1] anche se affetta da una pesante distorsione e da tipici difetti della DWT. Questo secondo punto sarà affrontato più avanti. In ogni caso si osserva che la compressione porta a dei risultati diversi a seconda del banco di filtri utilizzato. Notiamo che per valori bassi del bit-rate si ha la differenza maggiore; mentre per il Villasenor 1 e 3 il comportamento è simile, per il Villasenor 5 le prestazioni in termini di PSNR sull'immagine compressa sono le peggiori. Al crescere del rate , i risultati tendono ad euguagliarsi. Per capire il motivo di tutto ciò occorre andare ad indagare all'uscita del banco di analisi nei tre casi. In figura 3.5, 3.6, 3.7 la griglia dei coefficienti della DWT per i tre banchi.

[1]Si osservi che un'immagine ricostruita può essere definita ottima, dal punto di vista della visione umana, quando il PSNR supera 30dB, ovvero quando l'errore di ricostruzione risulta inferiore a 8. È evidente che questi limiti sono indicativi perché la qualità della ricostruzione dipende dalle caratteristiche dell'immagine.

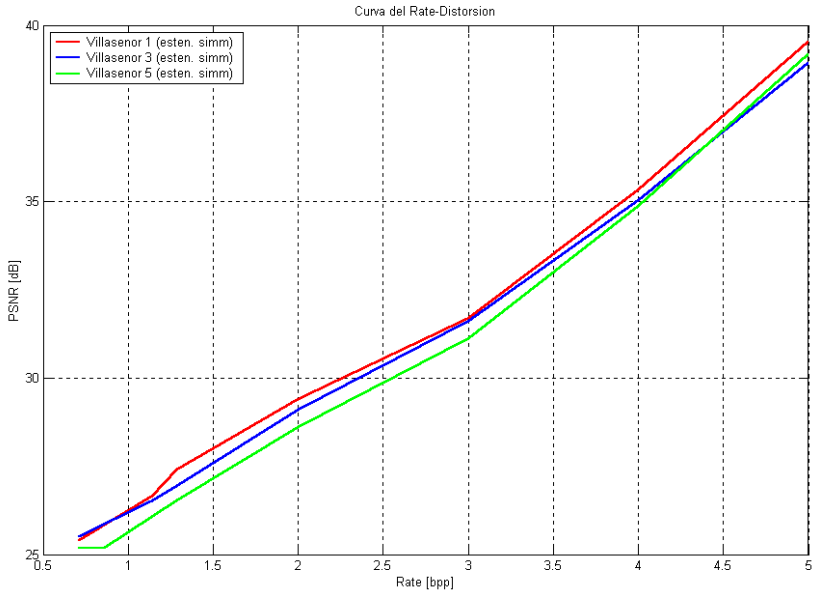


Figura 3.4. PSNR vs bit-rate: DWT con estensione simmetrica ai bordi



Figura 3.5. DWT, decomposizione su due livelli. Banco: Villasenor 1



Figura 3.6. DWT, decomposizione su due livelli. Banco: Villasenor 3



Figura 3.7. DWT, decomposizione su due livelli. Banco: Villasenor 5

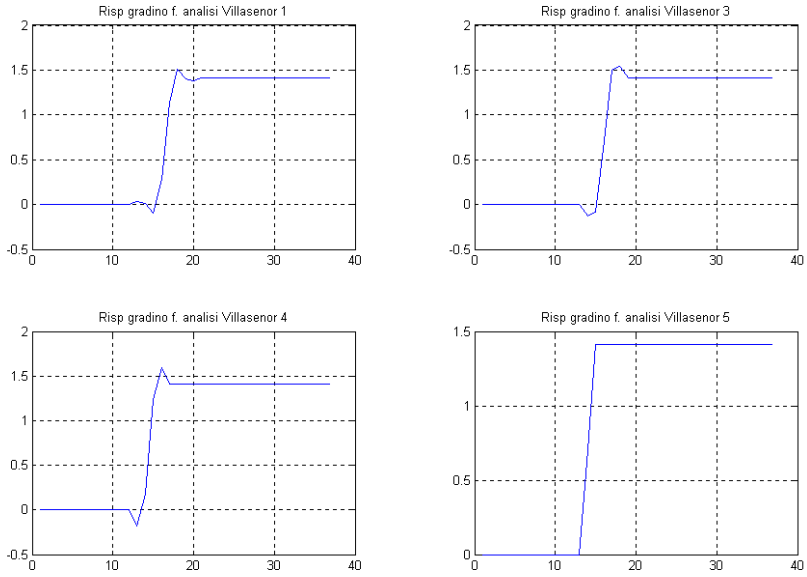


Figura 3.8. Risposta al gradino. Villasenor 1, 3, 4, 5.

Osservando le griglie delle tre decomposizioni, si nota che le differenze per i tre banchi di filtri considerati stanno nelle bande ad alta frequenza. Risulta che per Villasenor 3 e Villasenor 5 le sottobande LLHH, LH , HL, HH hanno un maggiore contenuto energetico, cioè un maggior numero di coefficienti relativi ad alte frequenze spaziali non nulli. Il motivo di tutto ciò è presto spiegato se si vanno a vedere come sono fatte le risposte all'impulso di questi filtri: meglio ancora le risposte al gradino in quanto le discontinuità presenti nell'immagine sono appunto di questo tipo (figura 3.8). La differenza sostanziale, sta nella regolarità del gradino. Mentre per Villasenor 1 e 3 la risposta presenta delle lievi oscillazioni ma è abbastanza regolare, con Villasenor 4 e 5 il gradino diviene molto ripido. Questa non regolarità nella risposta del filtro è la causa della distribuzione dei coefficienti anche nelle sottobande più alte in frequenza; questo non rappresenterebbe un problema quando tra il banco di analisi e quello di sintesi non ci sono perturbazioni dato che i filtri impiegati sono a perfetta ricostruzione.

Altresì, questo fenomeno non è auspicabile quando si parla di compressione in quanto la quantizzazione come si sa tende distruggere l'informazione contenuta in queste bande ed introducendo in ricostruzione distorsione in quantità sempre maggior, quanto più alto è il numero di coefficienti alle alte



Figura 3.9. Lenna: 0.25 bpp. Villasenor 1

frequenze. Ma la distorsione non è l'unico problema, in quanto mentre le bande più alte possono essere tranquillamente scartate senza introdurre errori visibili, per le altre alte bande non scartate, per le quali è possibile ancora conseguire un margine di guadagno, critico è il problema dell'allocazione dei bit. Questo perchè se il numero di bit assegnato è troppo basso, l'immagine ricostruita può risultare eccessivamente sfumata (errore di *blurring*).

Eseguita la quantizzazione dei coefficienti della DWT e quindi l'allocazione dei bit per le singole sottobande, si procede alla ricostruzione dell'immagine applicando il banco di sintesi. Valutiamo il risultato della compressione per i seguenti bit-rate: 0.25, 0.5, 2 bpp.



Figura 3.10. Lenna: 0.25 bpp. Villasenor 3



Figura 3.11. Lenna: 0.25 bpp. Villasenor 5



Figura 3.12. Lenna: 0.5 bpp. Villasenor 1



Figura 3.13. Lenna: 0.5 bpp. Villasenor 3



Figura 3.14. Lenna: 0.5 bpp. Villasenor 5



Figura 3.15. Lenna: 2 bpp. Villasenor 1



Figura 3.16. Lenna: 2 bpp. Villasenor 3



Figura 3.17. Lenna: 2 bpp. Villasenor 5

Indipendentemente dal tipo di banco utilizzato, si nota che per bassi valori di bit-rate, l'immagine ricostruita è affetta dal fenomeno di *blurring*. Vale a dire che l'immagine ricostruita presenta dei contorni sfumati, segno che il numero di bit allocati per le bande intermedie di frequenza, non sono sufficienti a rappresentare la quantità d'informazione che esse rappresentano. Se per 0.25 bpp l'errore di blurring è alquanto marcato, si nota che , passando a 0.5 bpp, una riduzione del tasso di compressione possa avere un impatto notevole sulla diminuzione di questo tipo di errore. A 2 bpp il fenomeno è completamente scomparso; la bontà della ricostruzione è confermata dal valore del PSNR che è largamente superiore di 30 dB.

Una conferma che molte volte alla bonta di una misura di distorsione non corrisponda un'altrettanto buona valutazione in termini di qualità soggettiva, la si ha se si osservano le immagini codificate a 2 bpp. Le immagini ricostruite sono affette da una distorsione ad alta frequenza che comporta presenza di ondulazioni in prossimità delle discontinuità nette. Tale distorsione va sotto il nome di *ringing* ed è legato alla natura delle risposte all'impulso (e/o a quelle del gradino) dei filtri di ricostruzione. Se queste sono lunghe e caratterizzate da molte oscillazioni, l'errore di quantizzazione dovuto alla pesante soppressione dei coefficienti nelle bande alte di frequenza, non rimane confinato sui contorni, ma si propaga in zone circostanti più o meno estese. Da un'analisi di figura 3.18 risulta che Villasenor 1 e 3 hanno delle risposte oscillanti e più lunghe rispetto al Villasenor 5 e l'osservazione delle rispettive immagini compresse ci conferma quanto affermato prima: l'utilizzo di Villasenor 5 limita l'influenza di questo tipo di distorsione ad alta frequenza. Decidere quale dei due filtri Villasenor 1 e 5 sia migliore, è essenzialmente un compromesso (almeno in questa situazione). Se la decisione è basata su termini oggettivi, e cioè il PSNR, la scelta cade sul Villasenor 1, mentre dal punto di vista soggettivo la scelta è Villasenor 5 in quanto introduce meno artefatti dovuti alla compressione.

3.3.2 Compressione mediante DWT con estensione periodica dei bordi

Si vuole eseguire ora una prova di compressione mediante una DWT che utilizza una funzione che estende i bordi in modo periodico e confrontare i risultati ottenuti con quelli trovati precedentemente. Il banco di filtri che prenderemo in considerazione è il Villasenor 5 e la trasformata è basata su dui livelli di decomposizione Già nel capitolo 2 si sono viste le differenze tra i due tipi di estensioni; ovvero la maggiore distribuzione di coefficienti nelle sottobande relative alle alte frequenze spaziali nel caso della estensione

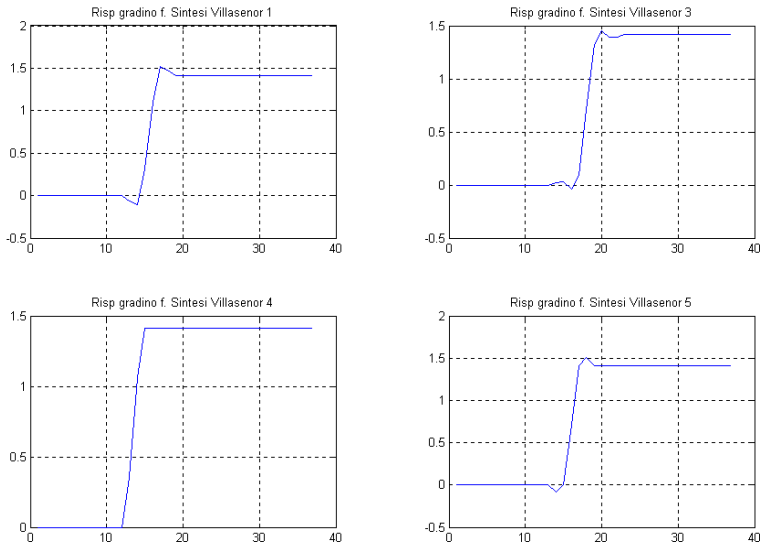


Figura 3.18. Risposta al gradino. Villasenor 1, 3, 4, 5.

periodica a causa della discontinuità che si manifesta al bordo. Questo fatto ci permette subito di trarre alcuni conclusioni: avendo un maggior contenuto energetico alle alle frequenze, la DWT con estensione simmetrica soffrirà maggiormente la quantizzazione. La conseguenza sarà una maggiore distorsione dell'immagine ricostruita. Verifichiamo ciò; in tabella sono riportati i dati ottenuti.

Bit-Rate[bpp]	Bit-Rate eff[bpp]	PSNR[dB]	MSE	$\Delta PSNR(sbs)[dB]$
0.25	0.714	25.016	204.854	0.151
0.5	0.857	25.018	204.726	0.15
1	1.142	25.914	166.603	0.607
2	2	28.234	97.630	0.354
3	3	30.817	53.862	0.306
4	4	34.640	22.337	0.22
5	5	39.050	8.076	0.105

Sia dalla tabella che dal grafico in figura 3.19 è evidente il guadagno in termini di PSNR della estensione simmetrica rispetto a quella periodica. Si evidenzia che il guadagno in questo caso è irrisorio, e che mentre è più marcato per bassi valori di bit-rate, tende ad annullarsi per valori alti del rate, in accordo con il fatto che la quantizzazione è meno pesante.

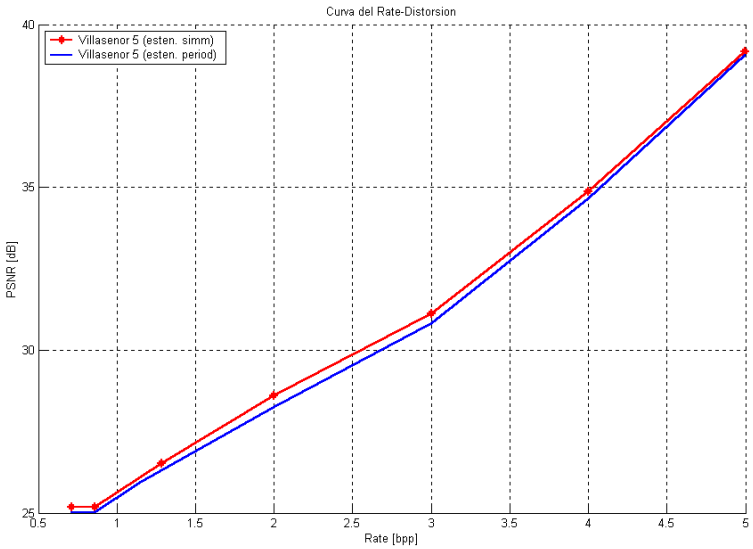


Figura 3.19. PSNR vs bit-rate: estensione simmetrica e periodica

Di seguito alcune delle immagini compresse ottenute dall'elaborazione.



Figura 3.20. Lenna: 0.25 bpp. Villasenor 5 (est. periodica)



Figura 3.21. Lenna: 0.5 bpp. Villasenor 5 (est. periodica)



Figura 3.22. Lenna: 2 bpp. Villasenor 5 (est. periodica)

3.3.3 Alcune prove di compressione

Si sono volute effettuare alcune prove di compressione mediante DWT al fine di verificarne le prestazioni in presenza di immagini con determinate caratteristiche. Il banco di filtri scelto è il Villasenor 1 e la trasformata è calcolata su due livelli di decomposizione. Le immagini scelte sono le seguenti:
Goldhill, Barbara: immagini a 8 bpp contenenti un alto livello di dettaglio e zone uniformi (figure 3.24, 3.27)
Sintetica, Pavimento: immagini a 8 bpp con vaste zone uniformi e zone a forte contrasto (figure 3.30, 3.26)
Circuito: immagine a 8 bpp con vaste zone uniformi e zone a forte contrasto ma anche dettagli (figura 3.25)
Stella: immagine a 8 bpp costituita da un alternanza di zone a forte contrasto (figura 3.28)
Testo: immagine a 8 bpp contenente del testo (figura 3.29)

Riportiamo in tabella per ogni immagine e per diversi valori del bit-rate, il PSNR misurato.

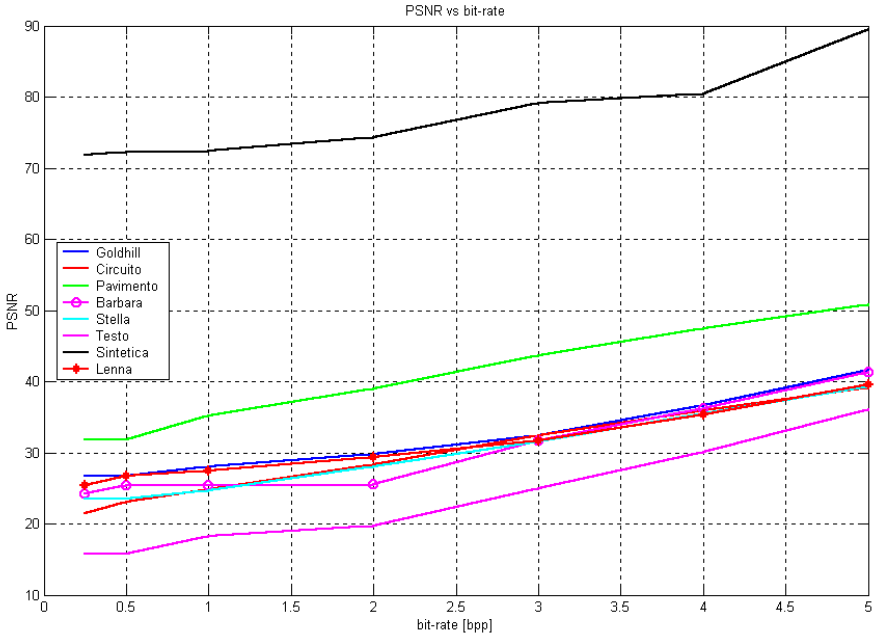


Figura 3.23. PSNR vs bit-rate per diversi tipi di immagini

Immagine	0.25	0.5	1	2	3	4	5
Goldhill	26.69	26.72	29.06	29.76	32.42	36.65	41.65
Circuito	21.40	23.01	24.76	28.36	32.45	35.89	39.07
Pavimento	31.87	31.89	35.22	38.94	43.63	47.47	50.78
Barbara	24.79	25.41	25.44	25.57	31.70	36.20	41.29
Stella	23.43	23.47	24.63	27.98	31.59	35.47	39.24
Testo	15.76	15.84	18.20	19.71	25.03	30.10	36.09
Sintetica	71.83	72.31	72.37	74.34	79.10	80.47	89.47

Dal grafico è possibile osservare che esiste una fascia di valori entro cui è contenuta gran parte delle immagini testate. Due immagini si discostano in modo visibile dal comportamento comune: sintetica, testo, pavimento. È interessante osservare il perché di questo comportamento. Nel caso di sintetica e anche di pavimento si nota come grazie allesistenza di una vasta area di colore uniforme sia possibile arrivare a compressioni elevate senza tuttavia introdurre troppo errore e per questo il PSNR dell'immagine risulta in genere superiore a tutte le altre a parità di rapporto di compressione.



Figura 3.24. Goldhill

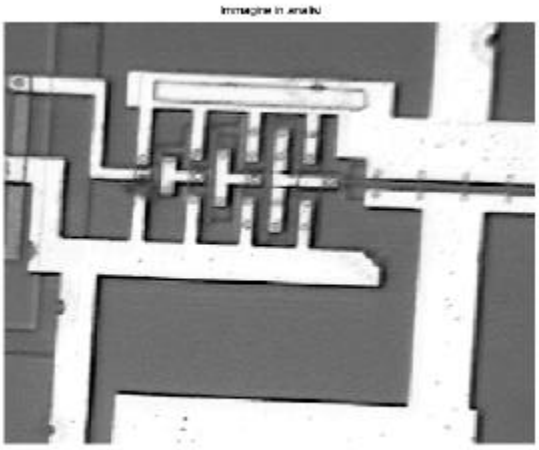


Figura 3.25. Circuito

Invece Testo possiede sì aree uniformi ma la complessità dovuta alla presenza di testo e di linee ad elevato contrasto rende molto più dannosa la presenza di artefatti di compressione e quindi fa decrescere il PSNR.

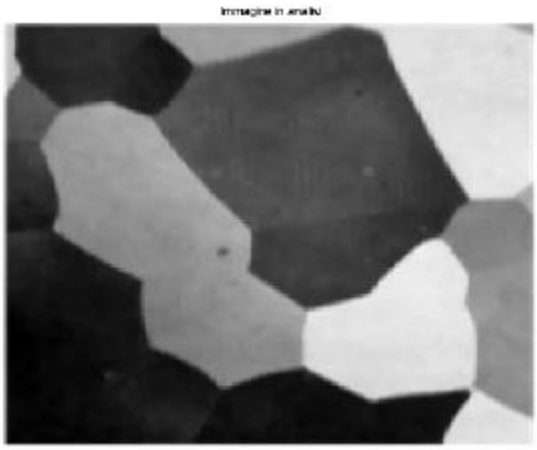


Figura 3.26. Pavimento



Figura 3.27. Barbara

3.3.4 I difetti della DWT

La compressione di un immagine effettuata mediante la DWT permette di ottenere dei tassi di compressione maggiori rispetto alla DCT ed inoltre,

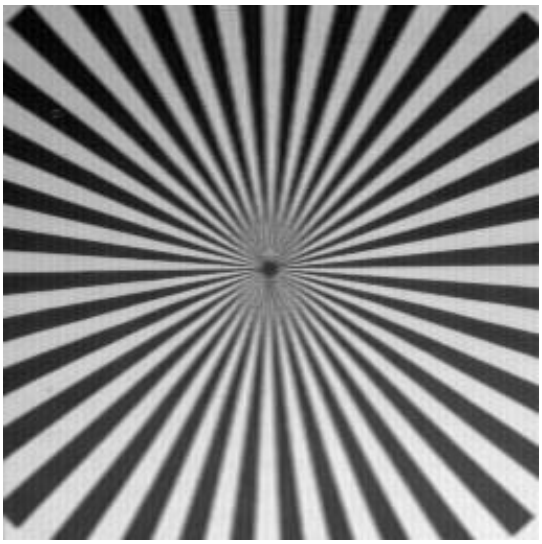


Figura 3.28. Stella



Figura 3.29. Testo

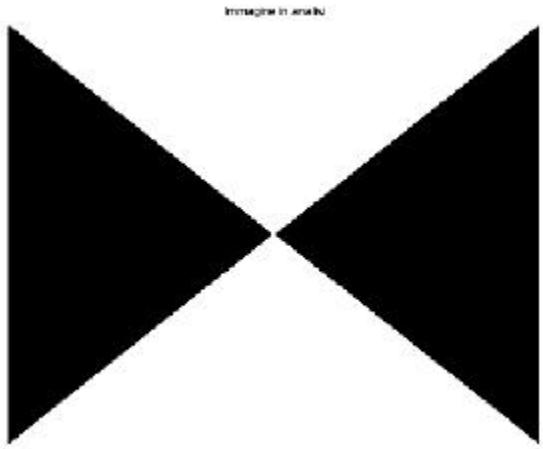


Figura 3.30. Sintetica

non presenta il fenomeno sgradevole della quadrettatura. Tuttavia anche la DWT introduce delle fastidiose distorsioni che sono diverse a seconda delle caratteristiche del banco di filtri che si utilizza. I principali tipi di distorsione sono: *ringing*, *distorsione di fase* e *distorsione a scacchiera*. Il ringing di cui si è già parlato in precedenza nasce dall'effetto combinato della pesante quantizzazione introdotta alle alte frequenze e dalla risposta dei filtri di sintesi quando questa ha un andamento oscillante ed non è corta. La distorsione di fase è provocata dalla non linearità della fase del filtro, mentre per evitare la distorsione a scacchiera è necessario un filtro passa-basso di sintesi molto regolare.

Per avere un idea di cosa sono, si confrontano le prestazioni di un banco di filtri Daubechies e di un Villasenor con lo stesso numero di momenti nulli delle rispettive ondine, ovvero con lo stesso numero di zeri (2) di H_0 in $w = \pi$ (stesso grado di regolarità). Di seguito sono riportate le risposte all'impulso dei filtri di analisi e sintesi. La regolarità dell'ondina per Villasenor 4, permette di conseguire un'elevata concentrazione dei coefficienti in analisi; infatti le funzioni dello sviluppo in serie wavelet danno coefficienti nulli quando rappresentano polinomi di grado uguale all'ordine dell'ondina. In sintesi l'uso di filtri regolari limita le oscillazioni nell'immagine dovute ad eventuali errori nei coefficienti della ricostuzione.

Si osserva che i filtri Villasenor sono più regolari e simmetrici di quelli Daubechies. Inoltre nel caso del banco Villasenor, risulta che la risposta

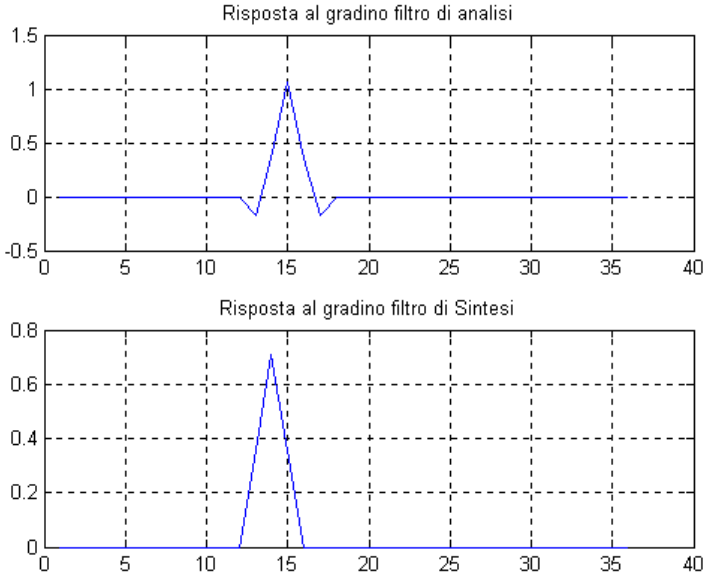


Figura 3.31. Risposta all'impulso dei banchi di analisi e sintesi del filtro Villasenor 4

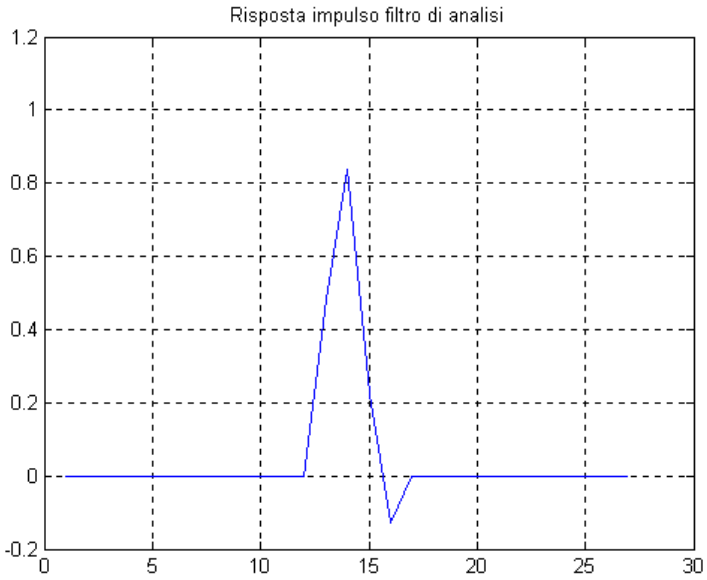


Figura 3.32. Risposta all'impulso del filtro Daubechies 4

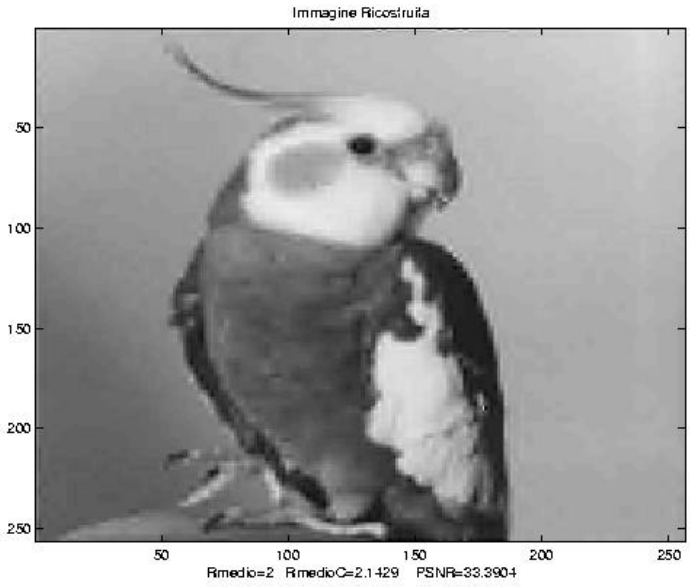


Figura 3.33. Bird: codifica a 2 bpp (Daubechies)

all'impulso qmf è più oscillante e più lunga di quella dqmf; quindi conviene usare il primo in analisi ed il secondo in sintesi. In tal modo la quantizzazione sulle alte frequenze provocherà effetti meno pesanti sulle discontinuità nette. Quantizzare i coefficienti relativi alla parte passa-alto dell'uscita del anco di analisi è come introdurre una perturbazione prima del banco di sintesi. Tale perturbazione sarà più evidente nel caso di filtri Daubechies dato che hanno la caratteristica di essere a fase non lineare. In questo caso, si è scelto un bit-rate pari a 2 bpp per evitare il fenomeno del blurring e meglio mettere in evidenza gli altri fenomeni di distorsione. Nelle figure 3.33, 3.35, è ben visibile la distorsione dovuta al ringing e alla non linearità della fase del filtro Daubechies: i contorni risultano essere tutti ondulati. Le migliori prestazioni del filtro Villasenor sono confermate anche dai valori di PSNR: 35.19 dB per il filtro biortogonale e 33.39 dB per il filtro ortonormale.

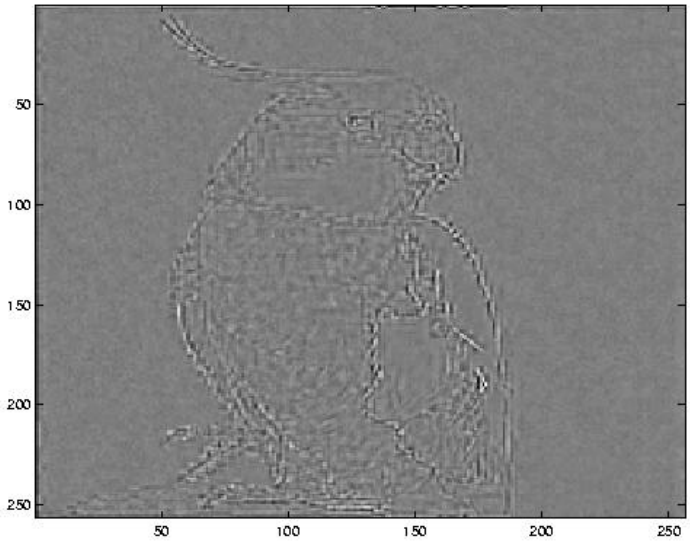


Figura 3.34. Bird: errore della ricostruzione con Daubechies

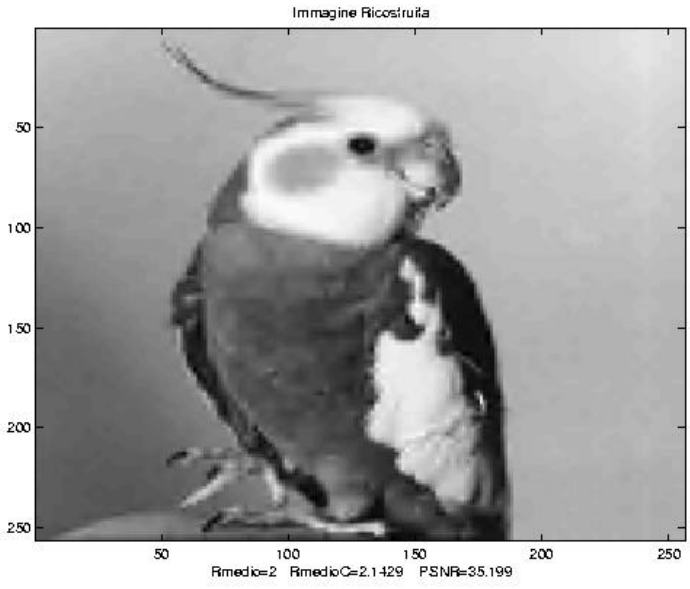


Figura 3.35. Bird: codifica a 2 bpp (Villasenor)

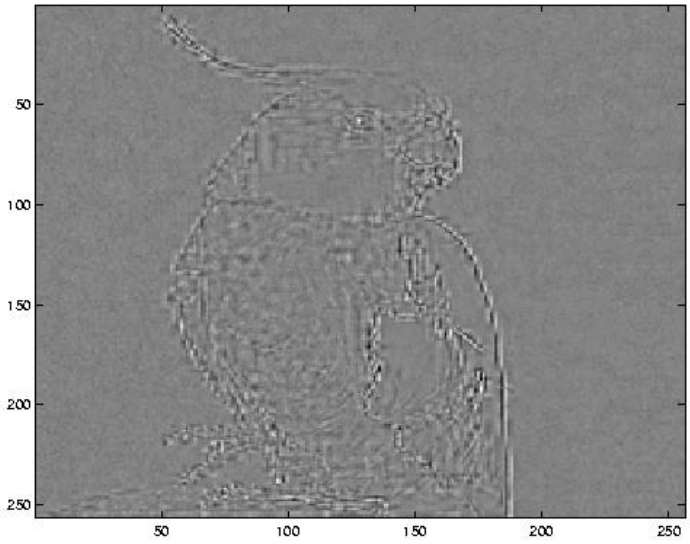


Figura 3.36. Bird: errore della ricostruzione con Villasenor

Capitolo 4

Jpeg vs Jpeg2000

4.1 Introduzione

La ragione di un nuovo standard di compressione delle immagini è da ricercarsi nell'incremento dell' utilizzo di tecnologie multimediali che richiedono grandi risorse a livello di spazio d'immagazzinamento di dati e di capacità computazionale. In questo senso ogni miglioramento nelle prestazioni degli algoritmi di compressione dei dati permette di sfruttare maggiormente le risorse a disposizione. L'evoluzione delle tecnologie comporta inoltre il sorgere di nuove necessità che prima non erano presenti o erano sottovalutate; quali ad esempio la codifica con regioni di interesse, la scalabilità, il controllo fine della compressione. Lo standard JPEG2000 rende disponibili queste e ulteriori caratteristiche che cercano di soddisfare i bisogni sorti negli ultimi anni e che gli standard esistenti non riescono a soddisfare in modo efficiente. Il risultato finale è il possibile utilizzo del JPEG 2000 sia in quei campi in cui si è sempre utilizzata la compressione dei dati, sia per introdurre queste tecniche laddove non sono ancora sfruttate. É indirizzato all'utilizzo in campi come stampa, acquisizione, memorizzazione di immagini mediche, fotografia digitale, Internet, E-commerce...
Ogni area ha differenti richieste in termini di qualità, occupazione di memoria, risoluzione. In base a questo è possibile individuare le caratteristiche che deve possedere il nuovo standard:

- Alte prestazioni per bassi bit-rate: necessaria nella trasmissione di immagini.
- Compressione possibile sia per immagini a tono continuo, sia per immagini a due livelli.
- Compressione sia senza perdite sia con perdite.

- Progressività nella trasmissione, per consentire la ricostruzione progressiva in base alla qualità o risoluzione che il destinatario ha scelto.
- Accesso casuale all'immagine e regioni di interesse: per consentire la ricostruzione di zone particolari dell'immagine o permettere una migliore qualità in zone particolari.
- Robustezza agli errori: necessaria nelle applicazioni di trasmissione di immagini (es. trasmissione su canale radiomobile).
- Architettura aperta.
- Possibilità di includere informazioni di copyright.

Nelle pagine seguenti verranno trattate le caratteristiche base della parte 1 dello standard focalizzando l'attenzione sui passi fondamentali di elaborazione dei dati numerici che compongono l'immagine originale per formare la sequenza di bit finale. In particolare si descriverà il metodo di suddivisione dell'immagine in sottoimmagini, la trasformata wavelet, la quantizzazione e la codifica che esse subiscono. Per valutare le prestazioni, si eseguiranno alcuni confronti qualitativi e quantitativi tra lo standard JPEG ed il nuovo standard di compressione d'immagini; si valuterà la robustezza dei due standard in termini di "immunità" al rumore ed infine verranno mostrate alcune delle potenzialità offerte da JPEG2000.

4.2 Jpeg2000

4.2.1 Schema generale

Architettura

Lo schema a blocchi riportato di seguito (figura 4.1) rappresenta la sequenza delle operazioni che vengono effettuate sull'immagine o su porzioni di essa. Per quanto riguarda la parte di codifica, a partire dall'immagine si ricava la sua Trasformata Wavelet, i coefficienti ottenuti vengono quantizzati e poi codificati mediante un codificatore entropico per ottenere l'immagine compressa. Il decodificatore opera in modo inverso per risalire all'immagine originale; siccome la compressione può essere sia lossless che lossy, l'immagine ricostruita potrà differire da quella di partenza a seconda del grado di compressione che le era stato applicato. Occorre ora specificare su quali insiemi di dati vengono effettuate le varie operazioni e in che modo i dati dell'immagine di partenza vengono convertiti nel bit-stream finale costituente il file in formato JPEG2000.

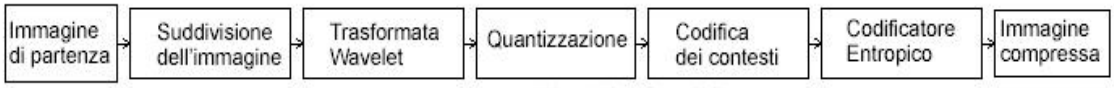


Figura 4.1. Schema a blocchi del codificatore

Suddivisione dell'immagine

Benché possa sembrare che sia un passo di elaborazione poco importante, il modo in cui l'immagine viene suddivisa consente di ottenere due caratteristiche importanti: la possibilità di accesso casuale all'immagine e di maggior resistenza agli errori. L'immagine viene divisa in varie componenti, in genere è possibile dividerla nelle tre componenti RGB, da cui si passa alle componenti YCrCb (YUV) con una trasformazione irreversibile o reversibile, in quest'ultimo caso la matrice di conversione è semplicemente:

$$\begin{bmatrix} Y \\ U \\ V \end{bmatrix} = \begin{bmatrix} \left[\frac{R+2G+B}{4}\right] \\ R-G \\ B-G \end{bmatrix}$$

Ogni componente può essere campionata con un diverso passo di campionamento e quindi la suddivisione porta ad avere in generale un numero x di sottoimmagini ognuna con una sua dimensione, ma in tal caso è possibile applicare solamente la trasformazione irreversibile. Queste componenti vengono trattate singolarmente dal codificatore come se fossero immagini distinte. L'immagine è divisa in *tile*, cioè in parti di uguale dimensione che verranno trattate in modo separato tra loro, permettendo la decodifica indipendente di ogni tile dagli altri. Le due divisioni sopra descritte portano dunque ad avere un'immagine divisa in riquadri (i tile) ognuno formato da varie componenti.
Ogni componente di ogni tile viene poi decomposta in vari *livelli* dall'applicazione della trasformata wavelet; fornendo quindi un insieme di sottobande contenenti i coefficienti che descrivono l'immagine in termini di frequenze orizzontali e verticali. Il numero di coefficienti risultante è pari al numero di pixel presenti nell'immagine sorgente ed in modo analogo a quanto avviene in altri metodi, i coefficienti significativi tendono ad essere pochi, mentre la gran parte risulta possedere valore molto basso. Grazie alla quantizzazione vengono eliminati i coefficienti non significativi e viene poi ridotto il numero di bit necessario per rappresentare ogni coefficiente quantizzato mediante la codifica. Per svolgere quest'ultima operazione ogni sottobanda viene divisa in porzioni, chiamate *code-block*, che possono venire estratte in

modo indipendente le une dalle altre. Noti i coefficienti quantizzati di questi code-block, si codificano in successione i *bit-plane* dei coefficienti mediante tre successive scansioni. Un bit-plane è l'insieme dei bit che possiedono lo stesso peso nell'insieme dei byte che rappresentano tutti i coefficienti; quindi la codifica avviene in ordine di importanza dei bit che compongono i byte del code-block.

Il flusso di bit generato dalla codifica è raggruppato in *layer*, cioè in strati, ognuno dei quali permette, se aggiunto al layer precedente di ottenere una porzione di immagine di qualità superiore. Ogni layer a sua volta può essere suddiviso in pacchetti (packet). Questi pacchetti, presi a turno da ogni livello (appartenente a un tile che a sua volta appartiene a una componente), permettono la generazione del cosiddetto *codestream*, cioè della sequenza di dati che rappresenta l'immagine compressa. Nel paragrafo successivo verranno spiegati più in dettaglio le suddivisioni dell'immagine e del flusso di dati generato.

4.2.2 Il sistema di coordinate

I canvas e i tile

Si considera un'immagine posizionata all'interno di un'area detta canvas e posta nell'angolo in basso a destra del canvas stesso; si parla di canvas quando l'immagine che si considera è quella in alta risoluzione, cioè quella non ancora scomposta nelle componenti. Si applica all'immagine risultante una divisione in aree rettangolari dette tile: nell'immagine in alta risoluzione possiedono tutti le stesse dimensioni Tx,Ty. Le coordinate di inizio dei tile sono (Px,Py) e naturalmente devono essere tali che $0 \leqslant P_x \leqslant a_x$ e $0 \leqslant P_y \leqslant a_y$ come in figura 4.2. Inoltre il primo tile deve avere intersezione non nulla con l'immagine; questo per evitare che ci siano tile che non portano alcuna informazione utile poiché infatti risulterebbero contenere solamente dati provenienti dall'area vuota del canvas con notevole spreco di bit nella sequenza finale.

I code-block e i layer

Una volta che è stata calcolata la trasformata wavelet sul tile si dividono le varie sottobande in *code-block* sui quali viene effettuata la quantizzazione dei coefficienti. Infine ogni code-block viene codificato in modo indipendente dagli altri e in modo scalabile. Gran parte delle qualità del JPEG2000 provengono da questa particolarità che consente di controllare in modo fine la qualità dell'immagine che si vuole ricostruire (progressività del metodo) e aumentare

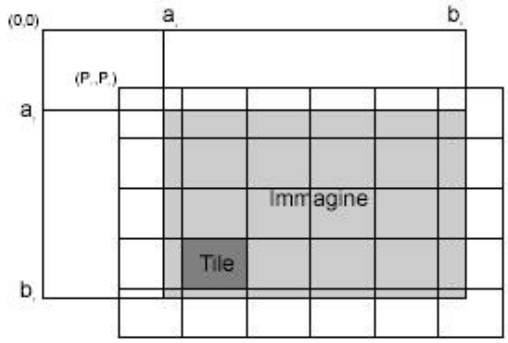


Figura 4.2. L'immagine nel canvas e i tile

la resistenza agli errori (in caso di errore solo il code-block interessato risulterà danneggiato e non tutta l'immagine). A questo punto ogni code-block contiene i dati per ricostruire completamente i coefficienti della Trasformata Wavelet (WT) per quella determinata porzione di livello di decomposizione. Si dividono opportunamente (secondo criteri di minima distorsione/qualità voluti) in varie parti i code-block ottenendo in uscita gruppi di bit. L'insieme di questi gruppi appartenenti ai diversi code-block costituiscono il *layer*: questa tecnica viene sfruttata nel JPEG2000 all'interno del codificatore chiamato EBCOT (Embedded Block Coding with Optimized Truncation) che verrà spiegato successivamente. Dopo la codifica dei blocchi si memorizzano i dati iniziando dal layer inferiore e via via andando verso layer superiori (che contengono i dati di raffinamento dell'immagine).

4.2.3 La trasformata wavelets

Il JPEG2000 utilizza un algoritmo relativamente nuovo rispetto al precedente standard JPEG che invece sfruttava la trasformata coseno discreta (DCT). Le ragioni che hanno portato alla scelta della Trasformata Wavelet sono dovute alle migliori prestazioni che offre rispetto a precedenti tecniche. Nel Jpeg2000 per default vengono utilizzati filtri monodimensionali e di lunghezza dispari per semplicità di implementazione, inoltre è supportato sia il calcolo in virgola fissa che quello in virgola mobile. Due sono i banchi di filtri supportati dallo standard (relativamente alla parte 1 dello standard):

- Per compressione con perdite, per default si utilizza il filtro Daubechies(9,7)

- Nel caso di WT reversibile, cioè per compressione lossless si usa un filtro Legal(5,3)

Il numero di livelli di decomposizione con il quale eseguire la WT, è libero. Lo standard permette di realizzare la trasformata Wavelet in due modi differenti. Il più tradizionale fa uso dell'implementazione a banco di filtri; il secondo si basa sul *lifting scheme*. Sia che si utilizzi lifting scheme, sia che si opti per una Wavelet a banco di filtri, i campioni da trasformare devono essere estesi ai bordi dell'immagine. Lo standard prescrive l'utilizzo di un'estensione simmetrica e periodica.

4.2.4 La quantizzazione

La quantizzazione dei coefficienti della trasformata wavelet avviene mediante un quantizzatore scalare uniforme. Nel caso di WT reversibile (intera) sostanzialmente il passo di quantizzazione viene posto pari a 1 e quindi i coefficienti quantizzati coincidono con quelli di partenza, negli altri casi il passo di quantizzazione viene scelto in modo da soddisfare particolari requisiti (qualità, bit-rate). La scelta del passo di quantizzazione viene fatta a livello di sottobanda e quindi tutti i code-block appartenenti alla stessa sottobanda possiedono lo stesso passo.

4.2.5 La codifica entropica: EBCOT

I coefficienti di ogni code-block dopo essere stati quantizzati vengono codificati partendo dal bit-plane più significativo e via via andando verso quello meno significativo. Per ogni bit-plane la codifica avviene in base a criteri riguardanti la significatività del bit confrontandola con i bit adiacenti (modellizzazione del contesto) per determinare il modello di probabilità dei simboli. Successivamente mediante il codificatore aritmetico (MQ coder) viene prodotto il bit-stream di dati compressi. La necessità di effettuare una compressione scalabile ha reso opportuna la definizione di un metodo per generare un bit-stream suddiviso in sottoinsiemi di dati ognuno contenente una rappresentazione dell'immagine a diversi livelli di risoluzione o qualità. La scalabilità rispetto alla risoluzione si ha quando è possibile ottenere dal bit-stream diversi sottoinsiemi di dati ognuno contenente la rappresentazione dell'immagine a differenti risoluzioni. La scalabilità rispetto alla qualità si ha quando l'unione di sottoinsiemi di dati fornisce un'immagine alla qualità voluta. Ogni code-block viene codificato in modo indipendente e il bit-stream ad esso associato viene costruito come unione di insiemi di diversa lunghezza ottenuti da considerazioni sulla distorsione voluta ad ogni successivo punto

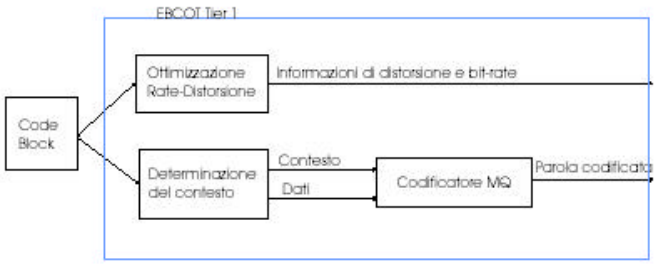


Figura 4.3. Struttura del tier 1 di EBCOT

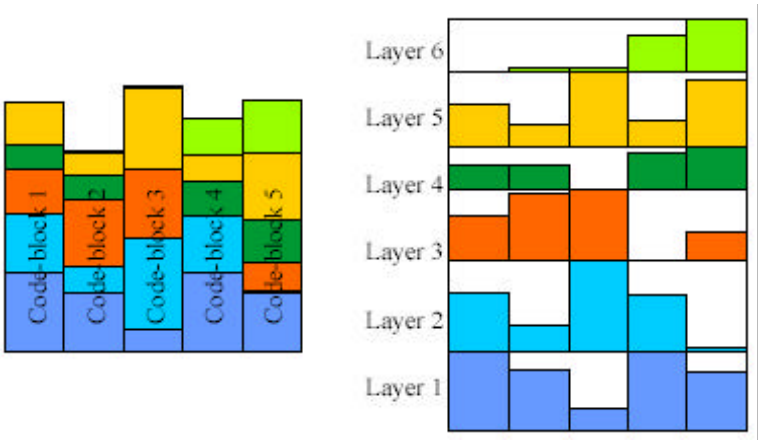


Figura 4.4. Generazione dei layer dai code-block

di troncamento. Questo è il processo che genera i layer. Ogni singolo layer è un insieme di informazioni codificate che aumenta progressivamente la qualità dell'immagine. Utilizzando un'opportuna segnalazione all'interno dello stream si possono identificare i layer. EBCOT viene suddiviso in due parti (detti tier 1 e 2); la prima si occupa della codifica dei code-block e rappresenta il processo di generazione del bit-stream per ogni blocco, mentre la seconda si occupa in prevalenza di gestire le informazioni riguardanti i layer e le informazioni di costruzione dei code-block e il loro inserimento nella struttura del file: in questa fase i code-block non sono più trattati in modo indipendente. Durante la codifica, è utilizzato un algoritmo che consente di determinare i punti di troncamento tra i vari layer in modo tale da consentire un buon controllo della distorsione.

4.2.6 Codifica a bit-plane dei blocchi

Elemento essenziale per permettere una codifica aritmetica corretta è la stima della distribuzione di probabilità dei simboli da codificare, per questo motivo il primo passo di elaborazione consiste nell'analisi del campione e dei pixel adiacenti secondo la supposizione che esista una certa correlazione tra i vari valori; questa operazione equivale a ricercare il valore del suo *contesto*. Per far questo legame si utilizza un particolare tipo di codificatore detto *bit-plane coder*. Si definisce bit-plane *p-esimo* l'insieme dei bit di peso 2^p di tutti i coefficienti appartenenti ad un determinato codeblock. Un *bit-plane coder,* è un processo che codifica l'informazione esplorando singolarmente ogni bit-plane a partire da quello più significativo. Il processo si compone di tre passi:

1. Significance propagation pass
2. Magnitude refinement pass
3. Cleanup pass

Occorre osservare che a ciascun elemento del bit-plane è applicato al fine della codifica solo uno dei tre passi sopra descritti.

4.2.7 Codifica con regioni d'interesse

Con il termine regione di interesse (ROI) si suole indicare l'intezione di voler codificare in modo migliore una porzione di immagine rispetto al resto. Questa operazione viene fatta in modo agevole grazie alla codifica a bit-plane in cui vengono codificati prima i bit più significativi dei coefficienti. Dunque è possibile scalare (diminuendone il valore) tutti i campioni non appartenenti alla ROI in modo tale che essi diventino significativi dopo che lo sono diventati gli altri; in altre parole è come aver quantizzato in modo più preciso i coefficienti all'interno della regione di interesse che non quelli esterni.

4.3 Descrizione dell'esperienza e risultati ottenuti

I criteri per valutare le prestazioni di un algoritmo di compressione delle immagini possono essere numerosi, a seconda della finalità dell'analisi: oltre a valutazioni sulla qualità e occupazione finale dell'immagine paragonata a quella originale, è possibile fare valutazioni riguardo alla complessità computazionale, alla memoria necessaria durante l'elaborazione e, nel caso di

implementazioni a livello hardware, il numero di porte logiche necessarie per l'implementazione. Nei prossimi paragrafi verrà trattato il primo tipo di valutazioni poiché in genere si cerca sempre di arrivare al compromesso di buona qualità e bassa occupazione di memoria, mentre altri tipi di confronti dipendono in modo più marcato dall'obiettivo che ci si propone quando si sviluppa il sistema di compressione.
L'efficienza della compressione sta tutta nel riuscire a eliminare la ridondanza, mantenendo però la stessa qualità dell'immagine di partenza o al limite riducendone la qualità al livello voluto. È evidente quindi che esistono due modi di eliminare la ridondanza: il primo, quello senza perdite consiste nella codifica vera e propria; il secondo, con perdite, elimina i dati ritenuti ininfluenti basandosi su considerazioni di tipo percettivo. Metteremo pertanto a confronto i due standard di compressione JPEG e JPEG2000 sia dal punto di vista qualitativo che quantitativo; per quanto riguarda i confronti qualitativi ci si limiterà alla qualità percettiva dell'immagine ricostruita, per quelli oggetivi si farà impiego della nota metrica basata sul rapporto segnale/rumore di picco (PSNR).

Verranno prese in cosiderazione per le prove diversi tipi di immagini campione ciascuna definita su 256 livelli di grigio (8 bit per pixel); queste immagini saranno codificate successivamente ai seguenti bit-rate di riferimento:

- 0.125 bit/pixel
- 0.25 bit/pixel
- 0.5 bit/pixel
- 1 bit/pixel

Per quanto riguarda la scelta delle immagini di test l'ideale sarebbe avere a disposizione un gran numero di immagini per poter fare valutazioni di tipo statistico, tuttavia questa è una strada difficilmente percorribile e quindi si è scelto di utilizzare un set di immagini che cerchi di essere il più rappresentativo possibile delle diverse condizioni cui un compressore d'immagini deve lavorare. Questa scelta è purtroppo di tipo arbitrario anche se la presenza d'immagini comunemente utilizzate nei test (ad esempio per il JPEG) rappresenta un buon punto di partenza. Le immagini trattate sono le seguenti (di cui si riportano le caratteristiche):
Lenna: immagine con dettagli, aree uniformi, contrasti
Boat, Goldhill: immagini ricche di dettagli,forti contrasti
Bird: immagine con ampie zone uniformi e zone con contrasti sfumati
Frog: immagine ricca di dettagli molto fini

4.3.1 JPEG

Per poter comprimere le immagini di riferimento secondo lo standard JPEG, si farà uso di un programmino (cjpeg) che attraverso una intefaccia Matlab, fornendo l'immagine desiderata e il rate obbiettivo, produce una versione compressa della stessa al bit-rate voluto. Poiché il JPEG non consente di controllare finemente il rapporto di compressione ma permette di variare un parametro di *qualità*, si è reso necessario definire un procedimento di conversione che metta in relazione il rate voluto con il corrispondente fattore di qualità. Si lavora per approssimazioni successive, sino a che il rate obbiettivo sia stato raggiunto o per lo meno sta all'interno di una fascia di tolleranza. Questo parametro è l'altro dato che si introduce oltre all'immagine e al rate desiderato. Fondamentale è la scelta di questo valore (la tolleranza) in quanto deve essere commensurato al rate che si vuole conseguire: bit-rate bassi richiedono tolleranze molto basse (strette); questo ovviamente fa aumentare i tempi di elaboazione. Le scelte effettuate sono le seguenti:

$Rate[bpp]$	0.125	0.25	0.5	1
ε	0.0001	0.001	0.01	0.01

La prima immagine elaborata è "*Lenna*". Si tratta di un immagine di dimensioni 256×256 pixel, codificata su 256 livelli di grigio e dimensioni su disco pari a 64 Kb (Figura 4.5). La caratteristica di questa immagine è quella



Figura 4.5. Immagine di prova: Lenna

di possedere zone pressochè uniformi, quindi a bassa frequenza spaziale e zone ricche di dettaglio e a forte contrasto, di conseguenza ad alta frequenza spaziale. Rappresenta pertanto un buon test per verificare la compressione che è conseguibile con lo standard JPEG. I risultati ottenuti in termini di Rate, PSNR e dimensioni su disco, sono riportati in tabella:

Rate[bpp]	PSNR[dB]	Dimensioni[Kb]
1	33.4078	8.02
0.5	28.8974	4.01
0.25	23.8757	2.13
0.125	20.9583	1.44

L'analisi dei dati ottenuti, conferma le aspettative: è possibile conseguire tasso di compressione alti, mantenendo una buona qualità dell'immagine come dimostra il primo caso; questo essenzialmente è dovuto alla presenza nell'immagine di ampie zone uniformi che fanno si che la maggior parte dell'informazione a livello di coefficienti DCT, sia trasportata dai coefficienti DC e da quelli subito adiacenti. Sono si presenti zone con contrasti e zone con dettaglio, ma la quantizzazione a tale rate è tale che i coefficienti relativi alle alte frequenze spaziali non siano sacrificati in modo drastico.

Notiamo che diminuendo il bit-rate, diminuisce il PSNR per effetto della maggiore quantizzazione, ma almeno dal punto i vista quantitativo la qualità dell'immagine dovrebbe mantenenrsi su livelli accettabili. Questo fatto è comunque da verificare mediante l'osservazione, in quanto la trasformata DCT ha il difetto di introdurre nell'immagine degli artefatti che hanno scarso impatto oggettivo (in termini di PSNR), mentre dal punto di vista soggetivo sono molto fastidiosi. In corrispondenza di un rate pari a 0.125 bpp si nota come la qualità dell'immagine crolli. È fondamentale una comparazione soggettiva dei risultati ottenuti, al fine di valutare gli effetti della compressione. Nelle figure 4.6, 4.7, 4.8, 4.9 sono mostrate le immagini ottenute:



Figura 4.6. Lenna codificata a 1 bpp



Figura 4.7. Lenna codificata a 0.5 bpp



Figura 4.8. Lenna codificata a 0.25 bpp



Figura 4.9. Lenna codificata a 0.125 bpp

Per l'immagine codificata ad 1 bpp, si può dire che è praticamente indistinguibile rispetto l'originale; la quadrettatura non si avverte e i dettagli son ben definiti. Notiamo però come, diminuendo il bit rate a 0.5 bpp, sia in contrasto l'impatto oggettivo ottenuto dalle misure di distorsione, con quello soggettivo.

Se il PSNR ci suggerisce che la qualità è ancora accettabile, e del resto è vero a parte la leggera perdita di dettaglio che si ha nelle zone a forte contrato e ricche di particolari, quello che non ci dice è che si ha il manifestarsi dell'effetto di *quadretattura* che ha un notevole impatto soggettivo. In questo caso è appena percepibile soprattutto nelle zone uniformi dell'immagine. A 0.25 bpp questo effetto fastidioso è ancora più marcato e l'effetto combinato di quadrettatura e perdita di dettaglio dovuta alla quantizzazione rende l'immagine pessima. A 0.125 bpp, l'immagine è pressochè inesistente, si vede che le uniche zone risparmiate dalla quantizzazione sono quelle uniformi, mentre quelle a forte contrasto, con contorni sfumati e ricche di dettaglio, sono completamente eliminate.

La seconda immagine presa in esame è "*Boat*". Si tratta di un immagine di dimensioni 512×512 pixel, codificata su 256 livelli di grigio e dimensioni su disco pari a 405 Kb (Figura 4.10) Si tratta di un immagine che mette a dura prova JPEG in quanto contiene tanti dettagli (si veda ad esempio gli alberi delle navi, le cime che corrono tra essi e le ondulazioni dovute alla sabbia), zone a forte contrasto e zone uniformi. I risultati ottenuti dalla compressione



Figura 4.10. Immagine di prova : Boat



Figura 4.11. Boat codificata a 1 bpp

eseguita con le modalità indicate precedentemente sono riportati in tabella:

Rate[bpp]	PSNR[dB]	Dimensioni[Kb]
1	30.8017	51.1
0.5	26.1321	25.4
0.25	21.7313	13
0.125	16.3365	7.07

I risultati ottenuti ci confermano che il codificatore si trova a lavorare in condizioni più critiche rispetto al caso visto con l'immagine *Lenna*. Qualunque sia il bit-rate considerato, si nota che il valore del PSNR ottenuto è inferiore, questo a causa dell'elevato numero di coefficienti della DCT relativi alle alte frequenze spaziali. Inizialmente si parte con una qualità in termini di distorsione sostanzialmente accettabile e man mano che diminuisce il rate dapprima iniziano a scomparire i dettagli quali le cime che collegano gli alberi, le ondulazioni della sabbia, le zone con i contorni sfumati. Ad un rate di 0.125 bpp, si nota come il dettaglio sia completamente rimosso e le uniche zone non eliminate dalla quantizzazione siano quelle che presentino una certa uniformità. È evidente anche l'effetto di quadrettatura.



Figura 4.12. Boat codificata a 0.5 bpp



Figura 4.13. Boat codificata a 0.25 bpp



Figura 4.14. Boat codificata a 0.125 bpp



Figura 4.15. Immagine di prova: Goldhill

La terza immagine presa in esame è "*Goldhill*". Si tratta di un immagine di dimensioni 720×576 pixel, codificata su 256 livelli di grigio e dimensioni su disco pari a 405 Kb (Figura 4.15)

Si tratta di un immagine simile per le sue caratteristiche a "*Boat*", se non che risulta essere più critica per il processo di compressione visto l'elevato numero di dettagli e il forte contrasto tra zone uniformi. I risultati ottenuti



Figura 4.16. Goldhill codificata a 1 bpp

e i riportati nella tabella che segue, testimoniano questo.

Rate[bpp]	PSNR[dB]	Dimensioni[Kb]
1	27.7668	51
0.5	24.6446	25.2
0.25	21.7469	13.1
0.125	16.5385	6.37

Valgono commenti analoghi al caso precedente; la cosa che vale la pena evidenziare e che ci consente di affermare l'alto contenuto di alte frequenze spaziali dell'immagine è quello che accade a 0.125 bpp. Si noti che l'immagine ottenuta ha una qualità infima ed inoltre è peggiore rispetto allo stesso risultato ottenuto con "*Boat*".



Figura 4.17. Goldhill codificata a 0.5 bpp



Figura 4.18. Goldhill codificata a 0.25 bpp



Figura 4.19. Goldhill codificata a 0.125 bpp



Figura 4.20. Immagine di prova: Bird

La quarta immagine presa in esame è "*Bird*". Si tratta di un immagine di dimensioni 256×256 pixel, codificata su 256 livelli di grigio e dimensioni su disco pari a 64 Kb (Figura 4.20). L'immagine in questione, rappresenta tra le



Figura 4.21. Bird codificata a 1 bpp

cinque scelte quella con la quale la codifica JPEG "lavora più efficentemente". Infatti presenta ampie zone uniformi, un livello di dettaglio moderato zone di contrasto e contorni sfumati. In tabella i risultati ottenuti:

Rate[bpp]	PSNR[dB]	Dimensioni[Kb]
1	41.8767	8.20
0.5	38.0747	3.94
0.25	32.5581	2.02
0.125	17.5546	1.18

Si nota che a parità di rate, il PSNR è sempre più alto rispetto agli altri casi, questo perchè le ampie zone uniformi fanno si che l'energia nel dominio trasformato si concentri per la maggior parte nei coefficienti DC e in quelli immediatamente adiacenti rendendo così meno drastici gli effetti della quantizzazione. Valgono per altro cosiderazioni analoghe a quelle fatte in precedenza relativamente agli altri casi; le immagini compresse ottenute sono mostrate in figura 4.21, 4.22, 4.23, 4.24. È evidente il fastidioso effetto della quadrettatura, e la perdita di dettaglio all'aumentare del tasso di compressione;



Figura 4.22. Bird codificata a 0.5 bpp



Figura 4.23. Bird codificata a 0.25 bpp

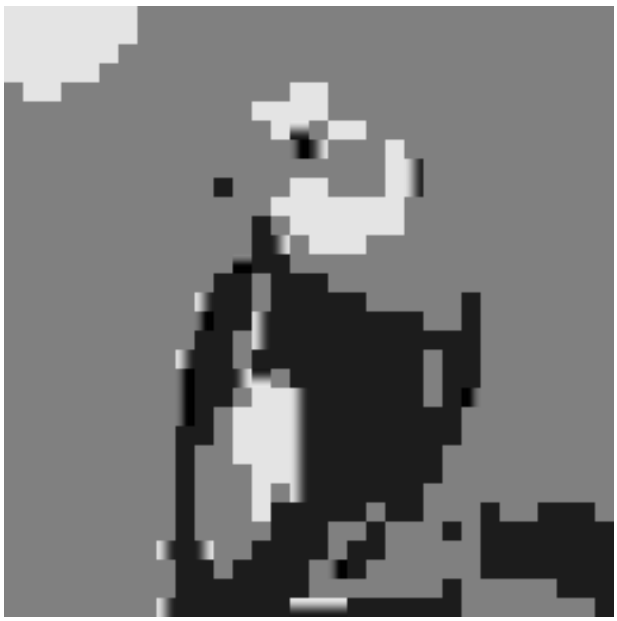


Figura 4.24. Bird codificata a 0.125 bpp



Figura 4.25. Immagine di prova: Frog

La quinta immagine presa in esame è "*Frog*". Si tratta di un immagine di dimensioni 320×256 pixel, codificata su 256 livelli di grigio e dimensioni su disco pari a 80 Kb (Figura 4.20). L'immagine in questione, la si può definire



Figura 4.26. Frog codificata a 1 bpp

come la peggiore immagine con la quale il codificatore di JPEG possa avere a che fare; questo in quanto si hanno zone di colore uniformi pressochè insignificanti e si ha un elevato dettaglio. Dettaglio che oltre essere rappresentato da contorni ben definiti è costituito da dei particolari piccolissimi come la pigmentazione della pelle della rana il cui contributo ai coefficienti della DTC relativi alle alte frequenze spaziali è rilevante. In tabella i risultati.

Rate[bpp]	PSNR[dB]	Dimensioni[Kb]
1	24.6030	9.91
0.5	22.9094	4.91
0.25	21.4953	2.51
0.125	16.8691	1.31

La criticità di questa immagine è confermata dai valori di PSNR ottenuti. Se ad 1 bpp la qualità è ancora soddisfacente, a 0.5 bpp mentre per tutte le altre immagini la distorsione è tale da rendere l'immagine accettabile, per *frog* la perdita di dettaglio è rilevante (notare i contorni sfocati e la pigmentazione che quasi scompare) e la quadrettatura è più che fasidiosa che mai. A 0.25 bpp il dettaglio scompare quasi del tutto, la nitidezza è pessima e la quadrettatura viene ancora di più esaltata. Con 0.125 bpp l'immagine è inesistente: la quantizzazione ha completamente distrutto l'informazione trasportata dai coefficienti della DCT, i quali come risulta evidente sono concentrati anche a frequenze spaziali alte.

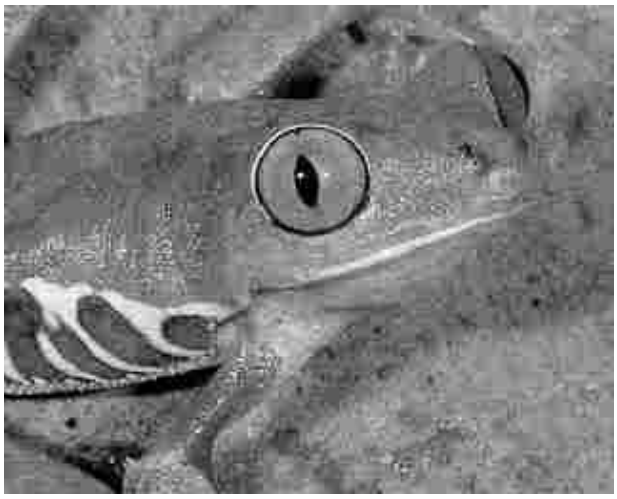


Figura 4.27. Frog codificata a 0.5 bpp



Figura 4.28. Frog codificata a 0.25 bpp

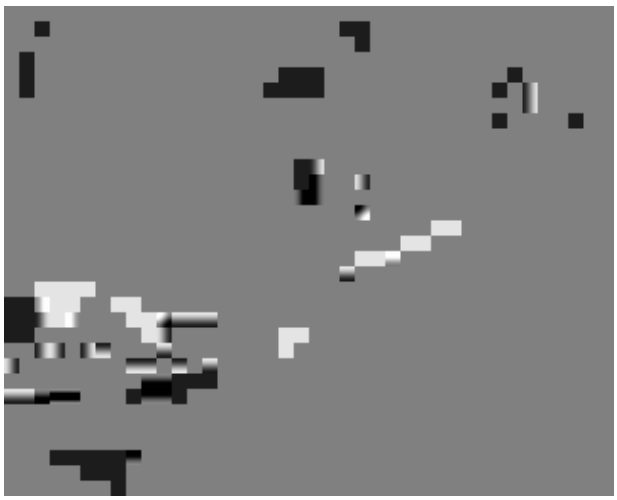


Figura 4.29. Frog codificata a 0.125 bpp

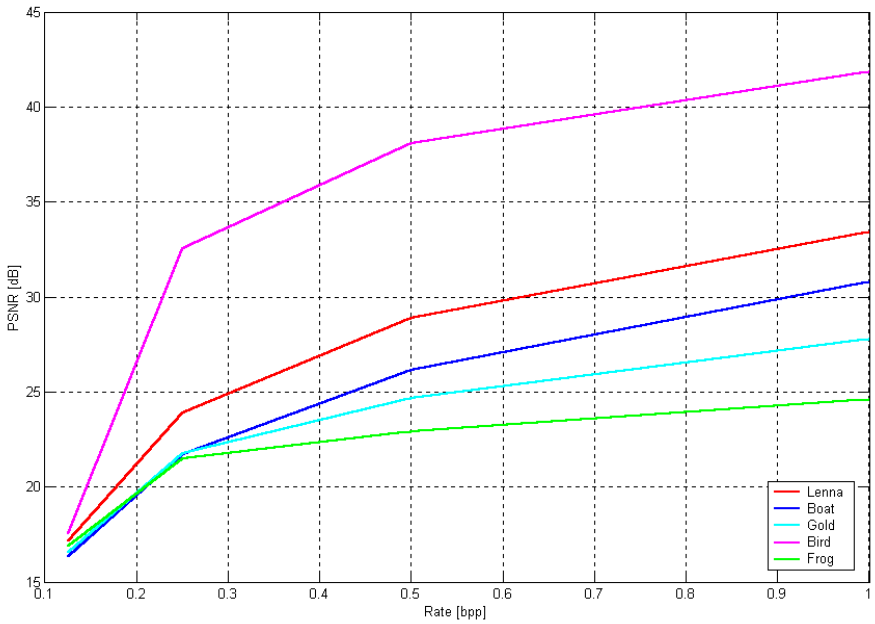


Figura 4.30. Compressione mediante JPEG

Riportiamo in figura 4.30 un grafico che riassume tutti i risultati ottenuti comprimendo le cinque immagini con lo standard JPEG. Per ogni immagine si riporta rate e PSNR relativo ottenuto. Notiamo che esiste una fascia di valori entro cui è contenuta la maggior parte delle immagini testate (lenna, goldhill, boat): questo è indice che hanno caratteristiche simili. Due immagini si discostano in modo visibile dal compartameto comune: bird e frog. Nel caso di bird, si nota come grazie all'esistenza di una vasta area uniforme si possa arrivare a compressioni elevate senza introdurre troppo errore e per questo il PSNR dell'immagine risulta superiore a tutte le altre a parità di rapporto di compressione. Frog possiede solo piccole aree uniformi e la complessità è dovuta alla presena di un elevato dettaglio e linee a forte contrasto che rendono molto più dannosa la presenza di artefatti di compressione facendo abbasare di molto il PSNR.

Se ne conclude pertanto che i valori di compressione conseguibili con JPEG sequenziale dipendono fortemente dal tipo di immagine eleborata. La codifica è efficiente se l'informazione portata dai coeffcienti DC e da quelli immediatamente adiacenti è predominante, e quindi si può operare una drastica quantizzazione dei coefficienti a frequenze spaziali superiori. Questo permette ai successivi algoritmi di *run-length* e Huffman di lavorare in modo ottimale.

4.3.2 JPEG2000

Per poter comprimere immagini secondo lo standard JPEG2000 si farà uso di uno dei vari codec al momento disponibile: Kakadu. Si tratta di un tool che consente di comprimere immagini conformi allo standard JPEG2000 e che permette di selezionare diversi parametri opzionali quali: rate, compressione lossy o losseless, banco di filtri, numero di livelli di decomposizione della DWT, dimensioni dei code-blocks,numero di layers, tiles, regioni di interesse etc ...

Le immagini campione prese come riferimento sono le stesse viste per JPEG, questo al fine di poter effettuare una comparazione dei risultati e prestazioni ottenute. La compressione è eseguita ai seguenti rate: 1 bpp, 0.5 bpp, 0.25 bpp, 0.125 bpp ed il codec è impostato con i seguenti parametri:

- Numero di livelli di decomposizione DWT: 5 (default)
- Dimensione dei code-blocks: 64×64 (default)
- Banco di filtri: W9X7 (default per compressione lossy)
- Numero di layers: 1
- Numero di tiles: 1

La prima immagine elaborata è *Lenna*. I risultati ottenuti sono mostrati in tabella:

Rate[bpp]	PSNR[dB]	Dimensioni[Kb]
1	37.7924	8.01
0.5	32.0535	3.95
0.25	28.1061	2.02
0.125	25.1938	1.07

Sono evidenti i migloramenti indotti dalla Trasformata Wavelets; qualunque sia il rate il PSNR misurato è superiore a quello trovato mediante la codifica con JPEG. In particolare quello che interessante osservare come questa sostanziale differenza si faccia sentire ai valori di bit-rate più bassi: si ha ad esempio un guadagno rispetto a JPEG di +8dB per un rate pari a 0.125 bpp. Questo miglioramento è dovuto al fatto che nel caso dell'immagine Lenna, o più in generale di un segnale immagine, mediante uno schema di scomposizione a sottobande, i coefficienti relativi alle bande più alte di frequenza spaziale, sono prossimi a zero e questo consente di conseguire elevati tassi di compressione senza introdurre una pesante distorsione. Lenna in questo senso si presta bene in quanto, come visto nel capitolo relativo alla compressione mediante DWT, la distribuzione di coefficienti nelle sottobande più alte di frequenza è esiguo e quindi la loro soppressione non induce una perdita di dettaglio rilevante. I risultati della compressone sono mostrati in figura 4.31, 4.32, 4.33, 4.34.

L'immagine codificata a 1 bpp risulta essere praticamente identica all'originale; anche a 0.5 bpp la qualità è eccellente, si ha una perdita di dettaglio nelle zone di contorno a forte contrasto (vedi le linee diagonali e trasversali descritte dai contorni del cappello) ma è del tutto impercettibile. Questi sono gli effetti della quantizzazione dei coefficienti della banda HH la quale contenendo i coefficienti a più alta frequenza spaziale, sarà la prima ad essere sacrificata. Diminuendo il bit-rate, diminuisce la qualità dell'immagine ma non in modo drastico come visto in JPEG, la prova la si ha se si confrontano le immagini ottenute a 0.125 bpp: con JPEG l'immagine è completamente distrutta mentre in JPEG2000, al di la della perdita ragionevole di dettaglio, quello che si ottiene è ancora intellegibile. Ovviamente l'immagine ottenuta non può essere utilizzata per applicazioni specifiche ma può costituire ad esempio nel caso di applicazioni su web (archivio di immagini), una prima anteprima di versioni a qualità superiore. In ogni caso è da evidenziare il fatto che non si rileva la presenza di artefatti: infatti la DWT non comporta errori di quadrettatura in quanto la trasformazione non richiede segmentazione.



Figura 4.31. Lenna codificata a 1 bpp



Figura 4.32. Lenna codificata a 0.5 bpp



Figura 4.33. Lenna codificata a 0.25 bpp



Figura 4.34. Lenna codificata a 0.125 bpp



Figura 4.35. Boat codificata a 1 bpp

Passiamo a trattare l'immagine "*Boat*". Ricordiamo che quest'immagine presentava con JPEG non pochi problemi a bit-rate bassi dato l'alto contenuto di dettagli. I risultati conseguiti con JPEG2000 sono riportati in tabella.

Rate[bpp]	PSNR[dB]	Dimensioni[Kb]
1	33.6792	50.6
0.5	29.0214	25.3
0.25	25.0739	12.6
0.125	21.8494	6.38

L'incremento di qualità sia in termini di PSNR sia dal punto percettivo è notevole come è dimostrato dai risultati che dalle immagini mostrate in figura 4.35, 4.36, 4.37, 4.38. Il minore guadagno in termini di PSNR (rispetto a quanto ottenuto con Lenna) è giustificabile dal fatto che l'immagine in esame ha nel dominio trasformato una distribuzione maggiore di coefficienti nelle sottobande relative alle frequenze più alte. È significativo notare come la compressione vada ad eliminare tutti quei dettagli che l'occhio umano non percepisce o che non vengono notati se non dedicandoci proprio l'attenzione; è il caso del terreno sabbioso e dei rilievi e scanalature sullo scafo della barca. Mentre a 1 bpp sono ben evidenti, a 0.125 bpp mancano totalmente: la superficie sabbiosa è priva delle increspature e lo scafo è una superficie



Figura 4.36. Boat codificata a 0.5 bpp



Figura 4.37. Boat codificata a 0.25 bpp



Figura 4.38. Boat codificata a 0.125 bpp

pressochè uniforme e liscia. Ma questo non lo si nota, almeno in prima analisi.

Passiamo ora trattare le altre immagini, per le quali riportiamo i risultati e ci limiteremo ad evidenziare eventuali fatti interessanti. Per l'immagine "*Goldhill*" i risultati sono mostrati in tabella e in figura 4.39, 4.40, 4.41, 4.42.

Rate[bpp]	PSNR[dB]	Dimensioni[Kb]
1	29.8708	50.6
0.5	26.3998	25.1
0.25	23.8187	12.5
0.125	21.8032	6.40



Figura 4.39. Goldhill codificata a 1 bpp



Figura 4.40. Goldhill codificata a 0.5 bpp



Figura 4.41. Goldhill codificata a 0.25 bpp



Figura 4.42. Goldhill codificata a 0.125 bpp



Figura 4.43. Bird compressa a 1 bpp

La compressione dell'immagine "*Bird*" da i seguenti risultati:

Rate[bpp]	PSNR[dB]	Dimensioni[Kb]
1	44.2692	7.94
0.5	41.0312	4.01
0.25	37.2004	2.08
0.125	33.0755	1.06

Le caratteristiche dell'immagine ovvero vaste zone uniformi, bordi sfumati, fanno si che la codifica con scomposizione in sottobande concentri i coefficienti nel dominio trasformato nelle bande relative alle frequenze spaziali basse. Questo fa si che che le bande in alta frequenza possano essere tranquillamente quantizzate senza che la distorsione indotta si manifesti in modo rilevante. L'elevato valore del PSNR anche a bit-rate basse ne è la prova.



Figura 4.44. Bird compressa a 0.5 bpp



Figura 4.45. Bird compressa a 0.25 bpp



Figura 4.46. Bird compressa a 0.125 bpp

La compressione dell'immagine "*Frog*" da i seguenti risultati:

Rate[bpp]	PSNR[dB]	Dimensioni[Kb]
1	26.3092	10
0.5	24.1503	4.92
0.25	22.6601	2.5
0.125	21.5883	1.30

L'immagine risulta essere sempre critica per la codifica visto che l'alto contenuto di dettagli, nel dominio trasformato induce un numero di coefficienti non nulli nelle bande più alte di frequenza. In ogni caso, la compressione è tale per cui la riduzione del dettaglio e quindi la distorsione indotta renda ancora l'immagine intellegibile se confrontata con la versione compressa con JPEG a parità di bit-rate.



Figura 4.47. Frog codificata a 1 bpp



Figura 4.48. Frog codificata a 0.5 bpp



Figura 4.49. Frog codificata a 0.25 bpp



Figura 4.50. Frog codificata a 0.125 bpp

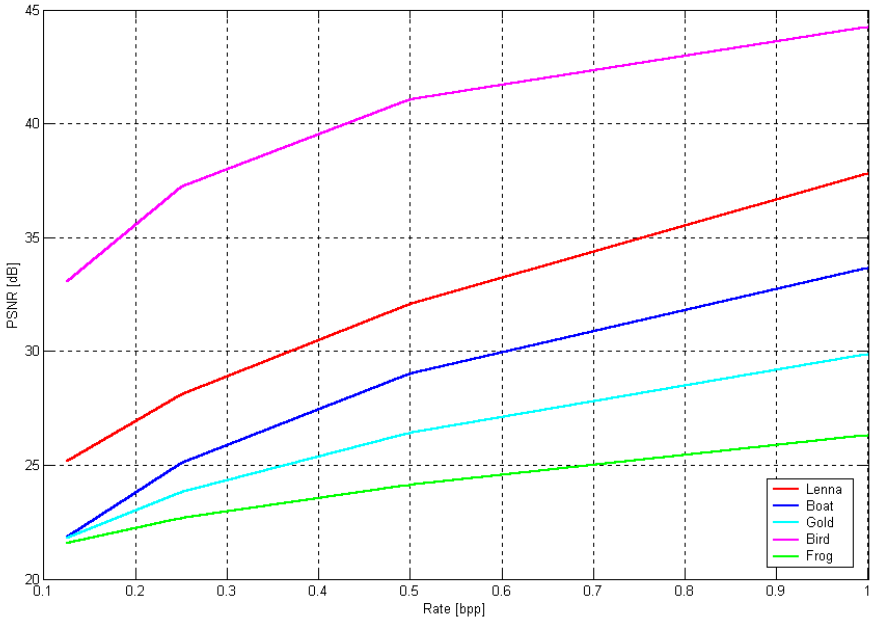


Figura 4.51. Compressione mediante JPEG2000

Con il grafico in figura 4.51, riassumiamo i risultati ottenuti con JPEG2000. Per ogni immagine, in corrispondenza del bit-rate desiderato si riporta il valore del PSNR conseguito. Per estrapolare un risultato vero e proprio occorrerebbe eseguire i test con un numero superiore di bit-rate e tipi di immagine; in ogni caso dal grafico è possibile osservare che esiste una fascia di valori entro cui è contenuta gran parte delle immagini testate. La qualità delle immagini risulta mediamente elevata fino a rapporti di compressione vicini a 40 (si ricorda che il bit rate delle immagini di partenza è 8 bpp). Due immagini si discostano in modo visibile dal comportamento comune: *Frog* e *Bird*. E interessante osservare il perché di questo comportamento.

Nel caso di *Bird* si nota come grazie all'esistenza di una vasta area di colore uniforme sia possibile arrivare a compressioni elevate senza tuttavia introdurre troppo errore e per questo il PSNR dell'immagine risulta in genere superiore a tutte le altre a parità di rapporto di compressione. Invece *Frog* non possiede pressochè aree uniformi e la complessità dovuta alla presenza di dettaglio molto fine e di linee ad elevato contrasto rendono molto più dannosa la quantizzazione; motivo per cui il PSNR è il più basso.

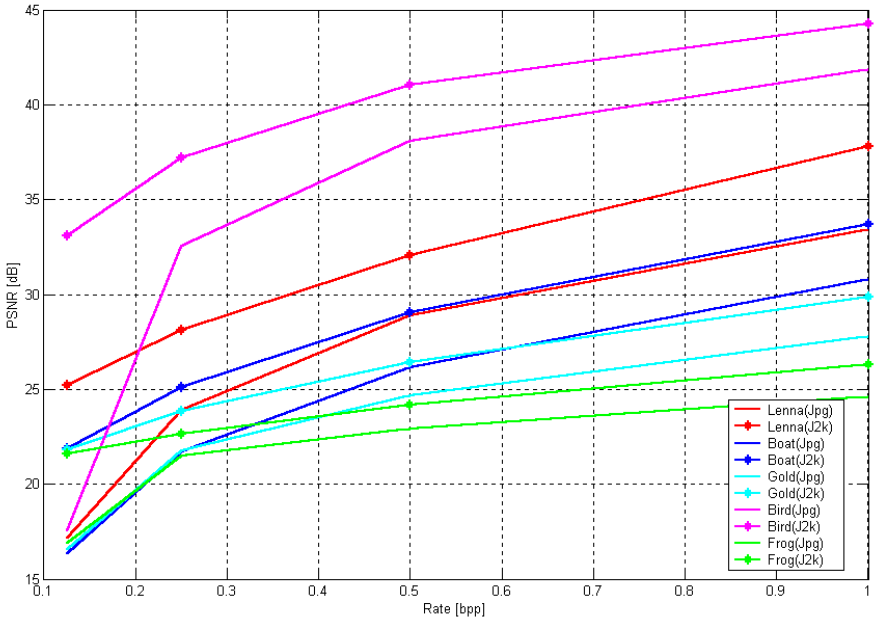


Figura 4.52. Confronto in termini di PSNR tra Jpeg-Jpeg2000

4.3.3 Jpeg e Jpeg2000 a confronto

È significativo mettere a confronto direttamente JPEG e JPEG2000 in termini di rapporto segnale/rumore di picco e occupazione di memoria, al fine di capire meglio le potenzialità del nuovo standard di compressione. In figura 4.52, è ben visibile il comportamento migliore di JPEG2000 rispetto lo standard di compressione basato sulla DCT; in particolare a bassi valori di bitrate. Significativi sono i casi dell'immagine Bird e Lenna, per le quali la presenza di aree uniformi consente di avere una qualità che consente ancora qualche applicazione dell'immagine, nonostante l'alto tasso di compressione: 64. Merito della scomposizione in sottobande della DWT, che consente di avre una rappresentazione compatta concentrando i coefficienti del dominio trasformato nelle sottobande più basse di frequenza, le quali sono le meno penalizzate dalla quantizzazione. Poiché il JPEG non consente di controllare finemente il rapporto di compressione ma permette di variare un parametro di qualità è stato deciso di utilizzare come asse di riferimento l'occupazione di memoria dell'immagine compressa (figura 4.53). Ciò permette anche di fare valutazioni rispetto a problemi di trasmissione data una larghezza di banda

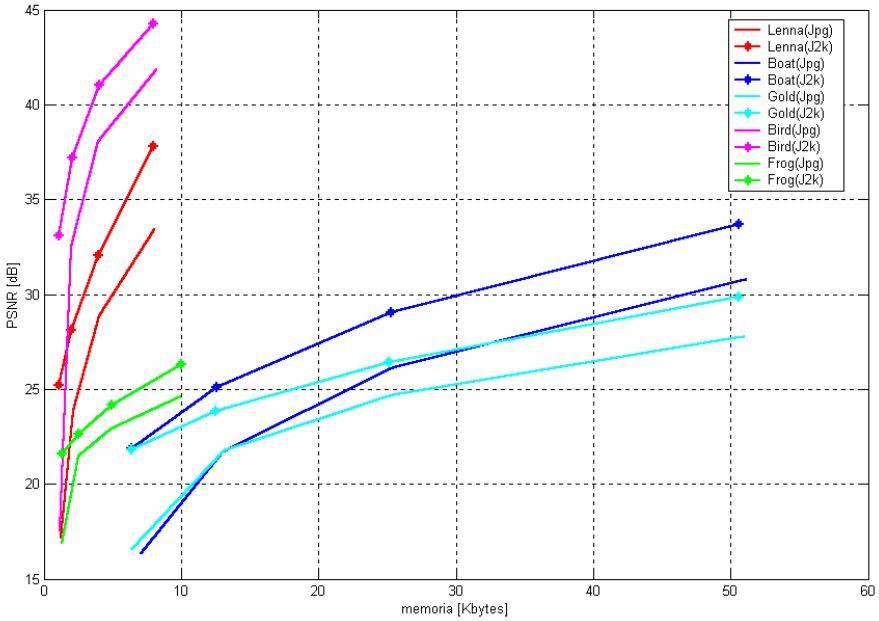


Figura 4.53. Confronto in termini di occupazione di memoria tra Jpeg-Jpeg2000

limitata tipici di applicazioni basate su web. Si nota come il JPEG2000 si comporti generalmente meglio rispetto al JPEG consentendo di avere buona qualità di immagine a bassi livelli di occupazione di memoria e quindi anche di bit-rate. Notare come le prestazioni siano notevolmente migliori in termini di qualità anche a bassi livelli di compressione ed inoltre nel caso di JPEG come sia necessario per poter avere uno stesso livello di qualità, aumentare il rate di una quantità sempre maggiore man mano che il tasso di compressione aumenta. Questo in accordo con i dati publicati da Fujifilm Software e riportati in tabella.

J2K bit-rate	Equivalent JPEG bit-rate	% larger for JPEG
0.25	0.53	112%
0.50	0.78	56%
0.75	0.92	23%
1.0	1.13	13%

4.3.4 Robustezza agli errori (Error Resilience)

È interessante valutare il comportamento di JPEG e quello di JPEG2000 in termini di "immunità" al rumore cioè di eventuali errori che cadano all'interno del bitstream. Ricordiamo che mentre per JPEG sequenziale[1] non esistono meccanismi di protezione e quindi di robustezza agli errori durante la trasmissione, l'Error Resilience è una delle caratteristiche maggiori del nuovo standard di compressione. In JPEG lo schema di compressione è tale per cui i dati vengono codificati e trasmessi blocchetto per blocchetto; pertanto se durante la trasmissione occorre un errore nella sequenza di dati, il decodificatore non sarà più in grado di decodificare la sequenza. Il risultato sarà che la ricostruzione della porzione di immagine rimanente sarà compromessa. L'idea di base con la quale lavora JPEG2000 è sostanzialmente diversa: si cerca di limitare la propagazione degli errori nel momento in cui sono stati commessi. Questo obbiettivo è raggiunto lavorando con una codifica a blocchi e inserendo dei marker di sincronismo nel flusso di dati generato da ciascun blocco Questi marker non aggiungono informazione ma aiutano il decodificatore a risincronizzarsi (tornare a decodificare correttamente) nel momento in cui un errore viene scoperto; varie sono le modalità con le quali operano i decodificatori in ogni caso si cerca sempre di minimizzare il numero di dati persi. La perdita che si ha in quanto si comprime meno aggiungendo ridondanza, è però giustificata dal possibile utilizzo dello standard nella trasmissione su canali rumorosi.

In ogni caso la presenza di un canale di trasmissione rumoroso genera un degrado della qualità dell'immagine poiché la ricostruzione del segnale codificato mediante il codificatore aritmetico sarà affetta da errore e il simbolo ricostruito sarà differente da quello codificato. L'effetto è quello di avere la presenza di bit errati all'interno del bit-plane, ma ciò è analogo ad aver quantizzato in modo più grossolano il segnale originale. Dunque la presenza di errori di trasmissione sull'immagine influenza il code-block allinterno del quale essi si sono verificati causando un decremento della qualità della porzione di immagine in esame.

Si sono pertanto confrontati i due standard per diversi valori di BER (bit error rate) rispettivamente: 10^{-6}, $0.4*10^{-5}$, 10^{-5},10^{-4}, $2*10^{-4}$, $5*10^{-4}$, 10^{-3}. L'immagine oggetto dell'analisi è *"Goldhill"*, codificata su 256 livelli di grigio, di dimensioni 720×576 pixel e occupazione su disco nel formato originale pari a 405 Kb e nel formato compresso 97.2 Kb(jpeg) e 100 Kb(jpeg2000). Il codec di JPEG200 è stato regolato sui seguenti parametri:

[1] In JPEG sequenziale ogni componente dell'immagine a colori è letta in modo sequenziale, nel verso sinistra-destra e alto-basso.

- Numero di livelli di decomposizione DWT: 5 (default)
- Dimensione dei code-blocks: 64×64 (default)
- Banco di filtri: W9X7 (default per compressione lossy)
- Numero di layers: 1
- Numero di tiles: 1
- Rate: 2 bpp

I risultati ottenuti sono riportati nelle figure che seguono nelle prossime pagine.



Figura 4.54. Goldhill (jpeg) per BER=10^{-6}



Figura 4.55. Goldhill (jpeg2000) per BER=10^{-6}



Figura 4.56. Goldhill (jpeg) per BER=10^{-5}



Figura 4.57. Goldhill (jpeg2000) per BER=10^{-5}



Figura 4.58. Goldhill (jpeg) per BER=10^{-4}



Figura 4.59. Goldhill (jpeg2000) per BER=10^{-4}



Figura 4.60. Goldhill (jpeg) per BER=2*10^{-4}



Figura 4.61. Goldhill (jpeg2000) per BER=2*10^{-4}



Figura 4.62. Goldhill (jpeg) per BER=10^{-3}



Figura 4.63. Goldhill (jpeg)2000 per BER=10^{-3}



Figura 4.64. Goldhill (jpeg)2000 per BER=10^{-6} Code-block=32×32

I risultati si commentano da soli, con JPEG già con BER bassi, un errore che cade nel bitstream ha ripercussioni su tutta l'immagine causando la mancanza di vere e proprie parti che induce una errata riscostruzione dell'immagine quando è possibile. Con JPEG2000 le cose sono sostanzialmente diverse: l'immagine è comunque ricostruita, non si notano parti mancanti, ma bensì si evidenzia come l'errore che si manifesta nel bit stream e che rimane confinato nel code-block si traduca in decodifica in una quantizzazione più grossolana dell'immagine. Gli effetti degli errori dipendono essenzialmente da quale sotto banda risulta danneggiata; se ad esempio l'errore cade nella sottobanda LL, nell'immagine ricostruita si percepirà una pesante degradazione visuale, dato che la banda LL può essere considerata una versione sottocampionata dell'immagine originale. Contrariamente, se l'errore cadrà nelle banda ad alta frequenza, verrà intaccata pesantemente la qualità in termini di dettaglio.

Si è provato a verificare se prendendo code-block di dimensione inferiore rispetto al caso precedente, consenta di avere una degradazione minore della qualità dell'immagine. La dimensione dei code-blocks presa è 32 e i risultati sono mostrati nelle figure che seguono. Non si notano sostanziali differenze, c'è un leggero miglioramento come è visibile per BER=2*10^{-4}, ma comunque vale la pena ricordare che il tutto è funzione di dove cade l'errore.



Figura 4.65. Goldhill (jpeg)2000 per BER=10^{-5} Code-block=32×32



Figura 4.66. Goldhill (jpeg)2000 per BER=10^{-4} Code-block=32×32



Figura 4.67. Goldhill (jpeg)2000 per BER=2*10^{-4} Code-block=32×32

4.3.5 Prove di compressione lossless

La prova è stata effettuata con lo scopo di osservare in che modo il numero di livelli di scomposizione della WT contribuisce a incrementare il rapporto di compressione. Il codec JPEG2000 è stato regolato in modo che esegue una trasformazione reversibile; pertanto l'effetto combinato DWT e banco di filtri (5,3) non fa altro che implementare la IWT cioè la traformata wavelets intera. I parametri con cui il codec è stato impostato sono i seguenti:

- Banco di filtri: W5X3
- Numero di livelli di decomposizione dalla DWT: variabile
- Numero di layers: 1
- Trasformazione reversibile
- Numero di tiles: 1

Le immagini prese in esame sono:
Lenna,Gold: le stesse viste in precedenza
Biomedica: immagine ricavata da una TAC a 8 bpp
Ledge: immagine a 8 bpp contenete ampie zone uniformi e forti contrasti
Parrots: immagine fotografica ad alto dettaglio a 24 bpp
Siding: immagine a 24 bpp contenente ampie zone uniformi e contrasti
Text :immagine proveniente da scansione a 200dpi a due colori, e successivamente portata a 256 toni di grigio, di un documento di testo
In tabella sono riportati per ogni singola immagine al variare del numero di livello della DWT, il rapporto di compressione.

Immagine	1	3	5	7	9
Biomedica	1.79	1.83	1.84	1.84	1.84
Gold	1.71	1.74	1.75	1.75	1.75
Lena	1.70	1.71	1.76	1.75	1.75
Ledge	9.28	10.08	10.20	10.14	10.12
Parrots	2.01	2.08	2.09	2.09	2.09
Siding	2.15	2.21	2.21	2.21	2.21
Text	6.22	5.22	5.21	5.20	5.20

Dalla tabella si nota che il rapporto di compressione, tranne nel caso di text, cresce all'aumentare del numero di livelli da 1 a 5 circa, tendendo poi un valore costante, segno che la codifica dei coefficienti diviene meno efficiente a causa principalmente del gran numero di sottobande che vengono generate che rendono pesante il calcolo. Infine è utile notare che al crescere del numero di livelli cresce notevolmente il tempo impiegato per generare il file.

4.3.6 Scalabità in risoluzione e SNR

La caratteristica principale di EBCOT è la capacità di produrre stream compressi altamente scalabili e ad accesso diretto. Si parla di immagini compresse scalabili quando è possibile individuare, nel bitsteam, uno o più sottoinsiemi in grado di rappresentare l'immagine originale a diversi livelli di risoluzione e di distorsione. I due tipi di scalabilità supportati da JPEG 2000 sono:

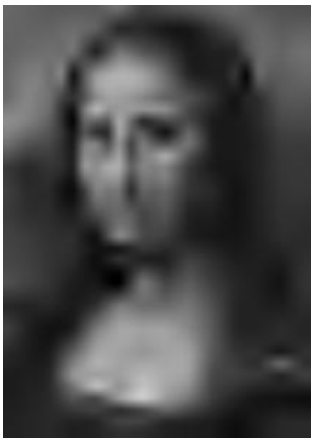
- Scalabilità in termini di SNR (qualità)
- Scabilità in risoluzione

Scalabilità SNR

Avere uno stream scalabile in termini di distorsione, significa individuare nello stream compresso dei sottoinsiemi distinti la cui unione fa aumentare progressivamente la qualità dell'immagine ricostruita e quindi diminuire la distorsione. Il codec che si è utilizzato per codificare le immagini secondo il formato JPEG2000, consente di produrre, sia nel caso di compressione lossless che lossy, una versione compressa scalabile dell'immagine originale, generando un numero di *layers* pari ai valori di bit-rate imposti dall'utente[2].

Si è eseguita una prova, codificando l'immagine "*Monna*" su sette layers cui corrispondono i seguenti rate: 2, 1, 0.5, 0.25, 0.125, 0.05, 0.03 bpp. Kakadu consente a partire dalla versione a maggiore qualità, che contiene tutti i layers, di visualizzare passo passo tutte le varie versioni sottraendo di volta in volta un layer. Il risultato ottenuto è mostrato nelle immagini seguenti

.



Monna : layer 1/7 a 0.03 bpp



Monna: layer 2/7 a 0.05 bpp

[2]Nel caso in cui il numero di layers sia inferiore rispetto ai valori di rate, i layers sono distribuiti logaritmicamente tra i rate imposti.



Monna: layer 3/7 a 0.125 bpp



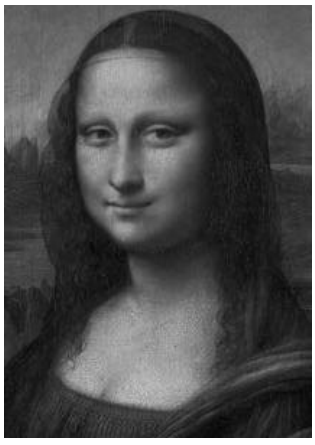
Monna: layer 4/7 a 0.25 bpp



Monna: layer 5/7 a 0.5 bpp



Monna: layer 6/7 a 1 bpp



Monna: layer 7/7 a 2 bpp

Scalabilità in risoluzione

Avere uno stream scalabile in risoluzione significa poter individuare al suo interno una successione di sottoinsiemi distinti di dati, ognuno dei quali fornisce una rappresentazione dell'immagine originale ad un livello di risoluzione opportuno. Muovendosi verso livelli di risoluzione maggiore, aumenta la risoluzione dell'immagine ricostruita. Il codec utilizzato, Kakadu, consente di poter impostare la scalabilità in risoluzione, agendo sul numero di livelli di decomposizione della DWT e ordinando il bitstrem in termini di risoluzione[3].

A dimostrazione di questo, si è codificata l'immagine "*Lenna*" ad un rate di 1 bpp, utilizzando 2 livelli di decomposizione per la DWT. Seguono i risultati in ordine crescente di risoluzione



Figura 4.68. Lenna: bassa risoluzione



Figura 4.69. Lenna: media risoluzione

[3]Il codec consente di ordinare il bitstream in termini di: Risoluzione, Layers, Posizione, Componente



Figura 4.70. Lenna: alta risoluzione



Figura 4.71. Lenna: risoluzione completa

4.3.7 I Tiles

Essenzialmente, il primo passo dell'algoritmo consiste nel suddividere l'immagine da codificare in regioni rettangolari disgiunte dette *tile*. Le dimensioni di tutte le tile sono le stesse, fatta eccezione per quelle ai bordi dell'immagine, che possono essere più piccole. Le elaborazioni successive sono applicate in modo indipendente per ogni tile di ogni componente: in questo modo sarà possibile estrarre e ricostruire anche solo singole porzioni dell'immagine originale.

Kakadu consente, specificando opportunamente le dimensioni del tile (in termini di dimensioni della partizione), di codificare ogni singolo tile su di un numero di layer diverso (questa è una delle possibili applicazioni). Si è



Figura 4.72. Lenna



Figura 4.73. Parrots

pertanto provato a dividere le immagini "*Lenna*" e "*Parrots*"in due tile e a codificare il tile di sinistra su 3 layers e quello di destra su un solo layers (i rate corrispondenti ai tre layers sono 1, 0.25, 0.125 bpp, per il tile di destra l'unico layers ha come bit-rate corrispondente 0.125 bpp). Le immagini ottenute sono mostrate in figura 4.72,4.73.

4.3.8 Regioni di interesse

Una delle novità introdotte in JPEG 2000 è la possibilità di dare maggior rilievo ad alcune regioni dell'immagine, favorendo in qualche modo la codifica



Figura 4.74. Lenna: codifica a 1 bpp

dei coefficienti appartenenti a queste zone. La tecnica che utilizza JPEG2000 è detta *MaxShift*. Questa tecnica fa si che la porzione d'immagine che interessa, venga elaborata in modo che:

- l'informazione della ROI venga codificata da EBCOT prima rispetto a quella appartenente allo sfondo
- l'informazione della ROI appaia, nello stream codificato, prima di quella appartenente allo sfondo.

Questo è ottenuto effettuando uno scalamento verso destra di tutti i coefficienti che non appartengono alle zone di interesse. Questa operazione fa si che i bit-plane più significativi siano occupati soltanto da valori che appartengono alla regione di interesse. La codifica entropica, partendo sempre dal piano più significativo, elaborerà i coefficienti delle ROI per primi. È necessario esplicitare, all'interno dell'immagine codificata, la posizione spaziale della regione di interesse, per consentire una corretta ricostruzione da parte del decodificatore.

Si è provato a codificare l'immagine "*Lenna*", a diversi bit rate, definendo una opportuna regione d'interesse. Kakadu, consente di specificare le dimensioni della ROI e il coefficiente di up-shift (scalamento). In figura 4.74, 4.75, 4.76, 4.77. notare come al diminuire del rate, la qualità della regione centrale sia sempre superiore della parte restante dell'immagine.



Figura 4.75. Lenna: codifica a 0.25 bpp



Figura 4.76. Lenna: codifica a 0.125 bpp



Figura 4.77. Lenna: codifica a 0.05 bpp

4.3.9 Conclusioni

In questa sezione si è descritto in modo essenziale il funzionamento dell'algoritmo di compressione di JPEG 2000 partendo dalla scomposizione dell'immagine in tile per permettere il calcolo della trasformata wavelet e la successiva quantizzazione dei coefficienti. Dopodiché si è descritto EBCOT e la sua struttura interna (generazione dei contesti e codifica aritmetica). Infine sono stati descritti i parametri principali con cui vengono valutate comunemente le prestazioni di un compressore di immagini e sono state fatte alcune prove per verificare in che modo il controllo sul bit-rate (e quindi sul rapporto di compressione) incide sulla qualità dell'immagine finale.

Sono state inoltre valutate le varie potenzialità di questo nuovo standard. L'introduzione di questo nuovo standard rappresenta un'innovazione per quanto riguarda l'approccio di elaborazione dei dati dell'immagine sorgente: si sfruttano nuove idee di codifica a sottobande, gestione fine del bitrate/qualità, definizione delle regioni di interesse, accesso casuale, progressività. La complessità aggiuntiva tuttavia può essere un fattore a sfavore del JPEG2000 ove esistono altri formati che producono approssimativamente gli stessi risultati in modo più semplice. In ogni caso il nuovo standard possiede una ricca serie di caratteristiche che lo rendono flessibile e perciò interessante in molti campi di applicazione.

4.3.10 Note sul software

Il software di riferimento al momento della scrittura di questo documento è definito nella parte 5 del documento FCD-15444.
Sono disponibili tre diverse applicazioni chiamate Kakadu, Jasper e JJ2000 (http://www.ece.ubc.ca/~mdadams/jasper, http://jj2000.epfl.ch). Jasper è scritto in linguaggio C mentre il secondo è scritto in Java. Per entrambi sono disponibili sia i sorgenti che i file già compilati. Nelle prove effettuate, si è usato Kakadu nella versione 3.2 (http://www.biodiversity.environment.gov.au/kakadu/kakadu/). L'autore è David Taubman che insieme a M.D.Adams (autore di Jasper) sono stati tra i principali fautori del Verification Model di JPEG2000.