

AA 2024-2025 - Metodi del Calcolo Scientifico - Progetto 2

Compressione di immagini tramite la DCT

Lo scopo di questo progetto è di utilizzare l'implementazione della DCT2 in un ambiente open source e di studiare gli effetti di un algoritmo di compressione tipo jpeg (senza utilizzare una matrice di quantizzazione) sulle immagini in toni di grigio. Comprende la implementazione di un codice e la scrittura di una relazione da consegnare al docente.

Prima parte

Implementare la DCT2 come spiegata a lezione in un ambiente open source a vostra scelta e confrontare i tempi di esecuzione con la DCT2 ottenuta usando la libreria dell'ambiente utilizzato, che si presuppone essere nella versione fast (FFT).

In particolare, procurarsi array quadrati $N \times N$ con N crescente e rappresentare su un grafico in scala semilogaritmica (solo le ordinate) al variare di N il tempo impiegato ad eseguire la DCT2 col vostro algoritmo fatto in casa e con l'algoritmo della libreria.

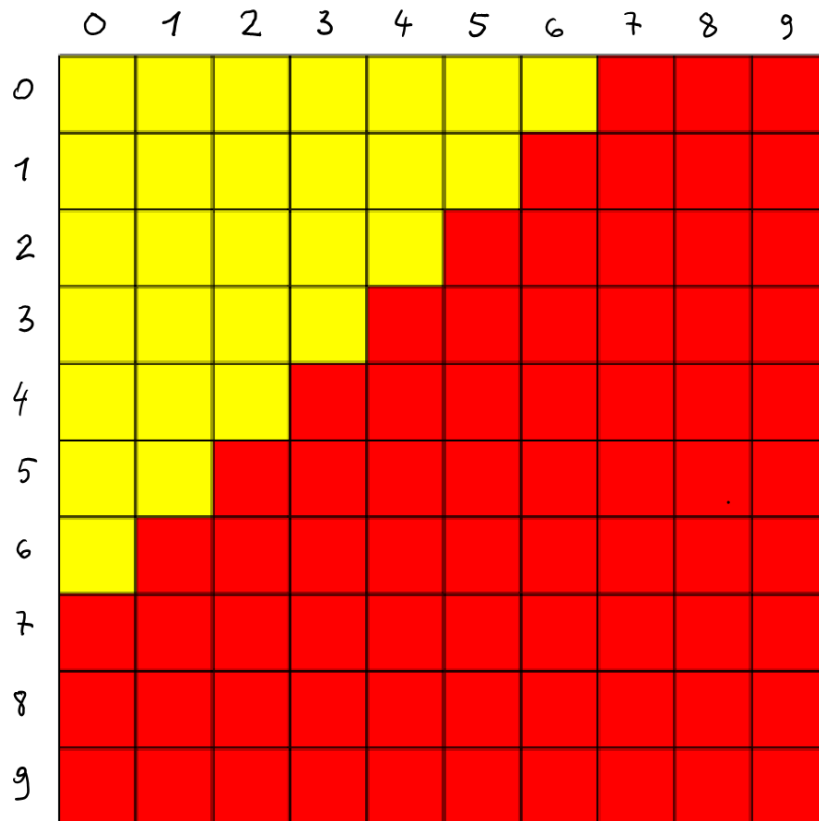
I tempi dovrebbero essere proporzionali a N^3 per la DCT2 fatta in casa e a N^2 per la versione fast (più precisamente a $N^2 \log(N)$). I tempi ottenuti con la versione fast potrebbero avere un andamento irregolare dovuto al tipo di algoritmo utilizzato.

Nella relazione includete anche un breve resoconto preliminare su questa parte.

Seconda parte

Scrivere un software che esegua i seguenti task:

- Creare una semplice interfaccia in modo che l'utente possa scegliere dal filesystem un'immagine `.bmp` in toni di grigio;
- permettere all'utente di scegliere:
 - un intero F che sarà l'ampiezza delle finestrelle (macro-blocchi) in cui si effettuerà la DCT2;
 - un intero d compreso tra 0 e $(2F - 2)$ che sarà la soglia di taglio delle frequenze (vedi sotto).
- suddividere l'immagine in blocchi quadrati \mathbf{f} di pixel di dimensioni $F \times F$ partendo in alto a sinistra, scartando gli avanzati;
- per ogni blocco \mathbf{f} eseguire le seguenti operazioni:
 - applicare la DCT2 (della libreria): $\mathbf{c} = \text{DCT2}(\mathbf{f})$;
 - eliminare le frequenze $c_{k\ell}$ con $k + \ell \geq d$ (sto assumendo che le frequenze partano da 0: se $d = 0$ le elimino tutte, se $d = (2F - 2)$ elimino solo la più alta, cioè quella con $k = F - 1, \ell = F - 1$). In sostanza bisogna eliminare i coefficienti in frequenza a destra della diagonale individuata dall'intero d , come esemplificato qui sotto: abbiamo $F = 10$ e $d = 7$. I coefficienti da eliminare sono indicati in **rosso**:



Blocco 10×10 con $d = 7$

- applicare la DCT2 inversa all'array **c** così modificato: $\mathbf{ff} = \text{IDCT2}(\mathbf{c})$;
- arrotondare **ff** all'intero più vicino, mettere a zero i valori negativi e a 255 quelli maggiori di 255 in modo da avere dei valori ammissibili (1 byte);
- ricomporre l'immagine mettendo insieme i blocchi **ff** nell'ordine giusto;
- visualizzare sullo schermo affiancate: l'immagine originale e l'immagine ottenuta dopo aver modificato le frequenze nei blocchi.

Nella relazione fornite una descrizione della struttura (incluso il linguaggio utilizzato, etc..) del codice implementato e fate qualche esperimento con le immagini proposte sul sito E-learning o con altre a vostra scelta; infine commentate i risultati. All'esame potrà essere richiesto di eseguire il vostro software su immagini fornite dal docente al momento.

NOTE:

- Il software dovrà essere open-source (per esempio MATLAB è escluso).
- Se il software permette solo il calcolo della DCT monodimensionale si può ricavare la DCT2 operando prima per righe e poi per colonne come spiegato a lezione.
- Prestare molta attenzione a come viene scalata la DCT2 (o la DCT). Infatti non sempre si usa lo scaling che abbiamo visto a lezione per le funzioni di base. Come caso test dovete verificare che il seguente blocchetto 8×8 :

231	32	233	161	24	71	140	245
247	40	248	245	124	204	36	107
234	202	245	167	9	217	239	173
193	190	100	167	43	180	8	70
11	24	210	177	81	243	8	112
97	195	203	47	125	114	165	181
193	70	174	167	41	30	127	245
87	149	57	192	65	129	178	228

venga trasformato in questo modo dalla DCT2:

1.11e+03	4.40e+01	7.59e+01	-1.38e+02	3.50e+00	1.22e+02	1.95e+02	-1.01e+02
7.71e+01	1.14e+02	-2.18e+01	4.13e+01	8.77e+00	9.90e+01	1.38e+02	1.09e+01
4.48e+01	-6.27e+01	1.11e+02	-7.63e+01	1.24e+02	9.55e+01	-3.98e+01	5.85e+01
-6.99e+01	-4.02e+01	-2.34e+01	-7.67e+01	2.66e+01	-3.68e+01	6.61e+01	1.25e+02
-1.09e+02	-4.33e+01	-5.55e+01	8.17e+00	3.02e+01	-2.86e+01	2.44e+00	-9.41e+01
-5.38e+00	5.66e+01	1.73e+02	-3.54e+01	3.23e+01	3.34e+01	-5.81e+01	1.90e+01
7.88e+01	-6.45e+01	1.18e+02	-1.50e+01	-1.37e+02	-3.06e+01	-1.05e+02	3.98e+01
1.97e+01	-7.81e+01	9.72e-01	-7.23e+01	-2.15e+01	8.13e+01	6.37e+01	5.90e+00

Dovete inoltre controllare che la prima riga del blocchetto 8×8 sopra:

231	32	233	161	24	71	140	245
-----	----	-----	-----	----	----	-----	-----

venga trasformata dalla vostra DCT monodimensionale in

4.01e+02	6.60e+00	1.09e+02	-1.12e+02	6.54e+01	1.21e+02	1.16e+02	2.88e+01
----------	----------	----------	-----------	----------	----------	----------	----------

- Nella relazione riportate il vostro codice e notizie sulla libreria utilizzata per le trasformate di Fourier.

Se avete dei dubbi mandate pure una email.