Discrete Cosine Transform

Chiara Ravazzi

National Research Council of Italy
Institute of Electronics, Computer and Telecommunication Engineering, c/o Politecnico di Torino, Italy
Systems Modeling & Control Group

Analisi tempo-frequenza e multiscala – 01RMQNG

Discrete Cosine Transform

In most applications

- signals are real and spectrum is symmetric

La parte immaginaria è nulla in molti dei segnali che troviamo

- half of the data is redundant in frequency domain and in time domains
- DFT : half of the computational time and storage space in the transform is unnecessary

Discrete Cosine Transform

About DCT

- Express a signal in terms of a sum of sinusoids with different frequencies
- DCT-II (used in signal and image processing)

$$X_k = a_k \sum_{n=0}^{N-1} x_n \cos\left(\frac{\pi}{N}\left(n+\frac{1}{2}\right)k\right) \qquad k = 0, \ldots, N-1,$$

Siamo dividendo il segnale in una somma di sinusoidi di differenti frequenze

with $a_0 = \sqrt{1/N}$, $a_k = \sqrt{2/N}$ if $k \neq 0$

- real transform with better computational efficiency (for real signals) than DFT
- X_k corresponds to sinusoid of frequency $k/2N$
- Both DCT and DFT spectra contain N coefficients

Breve confronto tra DFT e DCT

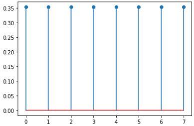
  - half of the N DFT coefficients represents the negative frequencies of the complex exponentials
  - while each of the N DCT coefficients represents a different frequency of a sinusoid
  - same frequency range but the resolution of the DCT spectrum is twice that of the DFT

Discrete Cosine Transform

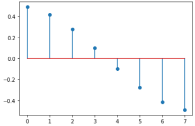
Lunghezza del segnale $N = 8$

Rappresenta una media del segnale

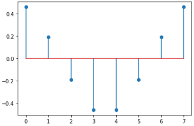
$k = 0$



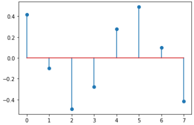
$k = 1$



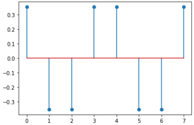
$k = 2$



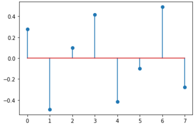
$k = 3$



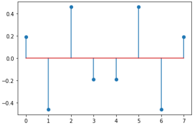
$k = 4$



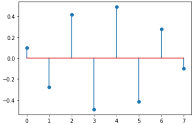
$k = 5$



$k = 6$



$k = 7$



Discrete Cosine Transform in Matrix Form

$k = 0, ..., N-1$, $n = 0, ..., N-1$

$$C_{n,k} = \begin{cases} \frac{1}{\sqrt{N}} \cos\left(\frac{(n+1/2)k\pi}{N}\right) & \text{if } k = 0 \\ \sqrt{\frac{2}{N}} \cos\left(\frac{(n+1/2)k\pi}{N}\right) & \text{otherwise} \end{cases}$$

- columns of C form orthonormal basis (check !)
- $C^T = C^{-1}$
- Discrete Cosine coefficients of signal $x \in \mathbb{R}^N$: $X = C^T x$

Inverse Discrete Cosine Transform

About DCT Operatore di sintesi

- DCT-III

$$X_k = \frac{1}{2}x_0 + \sum_{n=0}^{N-1} x_n \cos\left(\frac{\pi}{N} n \left(k + \frac{1}{2}\right)\right) \qquad k = 0, \ldots, N-1$$

- inverse of DCT : $x = (C^\top)^{-1} X = CX$ (DCT-III multiplied by $2/N$)

Digital Images

Consider

$$\{x : \mathbb{Z}_M \times \mathbb{Z}_N \to \mathbb{R}\}, \qquad \mathbb{Z}_N = \{0, \dots, N-1\}$$

- $x_{m,n}$ intensity of pixel (m,n), $m = 0, \dots, M-1$, $n = 0, \dots, N-1$
- Inner product :

$$\langle x, y \rangle = \mathrm{Tr}(y^\top x) = \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} x_{m,n} y_{m,n}$$

- Frobenius norm :

$$\|x\|_F = \sqrt{\langle x, x \rangle} = \sqrt{\sum_{n=0}^{N-1} \sum_{m=0}^{M-1} x_{m,n}^2}$$

- Standard basis

$$E_{m,n}^{(k,\ell)} = \begin{cases} 1 & \text{if } m = k, n = \ell \\ 0 & \text{otherwise} \end{cases}$$

2-Dimensional DCT

Two-dimensional DCT

- DCT-II performed along the rows and then along the columns (or viceversa)
- defined by (omitting normalization and other scale factors) :

Decomponiamo l'immagine come somme di sinusoide prima sulle righe e poi sulle colonne

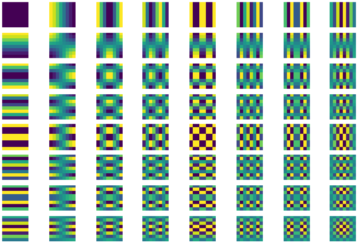
$$X_{k_1,k_2} = a_{k_1} a_{k_2} \underbrace{\sum_{n_1=0}^{N_1} \sum_{n_2=0}^{N_2}}_{\text{somma righe e colonne}} x_{n_1,n_2} \cos\left(\frac{\pi}{N_2}\left(n_2+\frac{1}{2}\right)k_2\right)\cos\left(\frac{\pi}{N_1}\left(n_1+\frac{1}{2}\right)k_1\right)$$

and

Prodotto di Kronecher

- $x = C_{N_1} X C_{N_2}^\top$ or $\mathrm{vec}(x) = C_{N_2} \otimes C_{N_1} \mathrm{vec}(X)$
- $X = C_{N_1}^\top x C_{N_2}$ or $\mathrm{vec}(X) = C_{N_2}^\top \otimes C_{N_1}^\top \mathrm{vec}(x)$

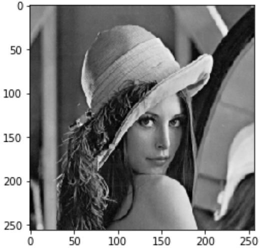
Utilizzando gli operatori matriciali definite prima e successivamente utilizzando le proprietà del prodotto di Kronecher



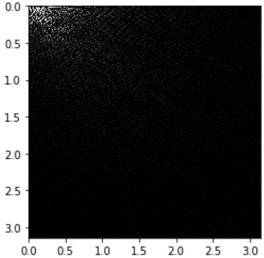
Combination of horizontal and vertical frequencies 2-DCT with $N_1 = 8$ and $N_2 = 8$

2-DCT and Inverse 2-DCT

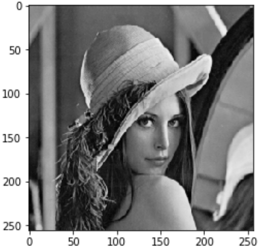
Cosa faremo nell'esercitazione -->
- Importiamo immagini bidimensionali
- Calcoliamo i coefficienti di DCT
- Verificheremo che partendo dai coefficienti della DCT e operando la trasformata inversa otterremo l'identità



Original image



DCT Coefficients



Identity