

PROVA SCRITTA DI ELABORAZIONE DEI SEGNALI MULTIMEDIALI del 11.5.09
(Ingegneria delle Telecomunicazioni)

Tempo: 2 ore e mezza. NON è consentito l'uso di materiale didattico e appunti propri.

EX. 1 Data l'immagine foto.jpg, si vogliono realizzare le seguenti operazioni:

1. il color-balancing dell'immagine X nello spazio CMY allo scopo di ridurre la quantità di rosso (le camicie della mamma e del bambino dovrebbero diventare bianche). Scrivete una funzione `function Y = colorbalancing(X)` e mostrate l'immagine elaborata;
2. il filtraggio dell'immagine Y nel dominio della frequenza mediante il seguente filtro:

$$H(\mu, \nu) = \begin{cases} 1 & |\mu| \leq 0.10, |\nu| \leq 0.25, \\ 0 & \text{altrimenti} \end{cases}$$

Scrivete allora una funzione `function Z = filtraggio(Y)` e mostrate l'immagine filtrata.

N.B. Il codice necessario per richiamare le funzioni va scritto in un unico script dal nome ex1.m.

EX. 2 Scrivete una funzione `function Y = coder(X)` che comprime l'immagine X attraverso le seguenti operazioni. L'immagine è divisa in blocchi 8×8 , quindi per ogni blocco:

1. si effettua la DCT;
2. si conserva solo la continua (DC);
3. si effettua la codifica predittiva delle componenti DC, cioè si trasmette DC_cur - DC_old (eccetto per il primo blocco);
4. l'errore di predizione è quantizzato uniformemente con passo $\Delta = 2$;

Scrivete quindi una funzione `function Xrec = decoder(Y)` che realizza le operazioni inverse e ricostruisce una versione distorta di X a partire da Y .

Nello script `ex2.m` usate le funzioni descritte sopra per codificare e decodificare l'immagine house.y (di dimensioni 512×512 e formato unsigned char), valutate l'MSE tra immagine originale e decodificata e mostrate il risultato.

EX. 3 Si vuole filtrare l'immagine upupaN.raw, versione rumorosa di upupa.raw, operando nel dominio wavelet. A tale scopo si scriva la funzione di prototipo `YF=Denoise1(Y,T)` che per ogni coefficiente trasformato Y (eccetto quelli della banda LL) effettua la trasformazione non lineare

$$YF(i, j) = \begin{cases} Y(i, j) - T(i, j) & \text{se } Y(i, j) > T(i, j) \\ 0 & \text{se } Y(i, j) \in [-T(i, j), T(i, j)] \\ Y(i, j) + T(i, j) & \text{se } Y(i, j) < -T(i, j) \end{cases} \quad (1)$$

dove $T(i, j)$ è costante e pari a T . Per realizzare il filtraggio si effettui la DWT dell'immagine rumorosa su 5 livelli di decomposizione (funzione `fw2d.m`), si calcoli la soglia T pari alla deviazione standard della banda trasformata HH, si richiami la funzione `YF=Denoise1(Y,T)` e si effettui infine la trasformata inversa di YF (funzione `iwt2d.m`). Si mostrino quindi immagine originale e filtrata, e si calcoli il corrispondente MSE.

Si ripeta l'esperimento con una diversa funzione, di prototipo `YF=Denoise2(Y,T)`, nella quale la soglia varia da punto a punto secondo la relazione $T(i, j) = T^2/(|Y(i', j')| + 0.1)$, dove $i' = \lceil i/2 \rceil$, $j' = \lceil j/2 \rceil$, e $\lceil x \rceil$ è il primo intero maggiore di x .

N.B. Il codice necessario per richiamare le funzioni va scritto in un unico script dal nome ex3.m.