

PROVA SCRITTA DI ELABORAZIONE DEI SEGNALE MULTIMEDIALI del 9.2.11
(Ingegneria delle Telecomunicazioni)

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

EX. 1 Si vuole realizzare lo smoothing dell'immagine a colori volto.tif nello spazio HSI con un filtro gaussiano con $\sigma = 0.15$. A tale scopo si scriva una funzione dal prototipo `function y=rgb2hsi(x)` che vi permette di passare dalle componenti RGB a quelle HSI, ricordate che la componente H è:

$$H = \begin{cases} \theta & \text{se } B \leq G \\ 360 - \theta & \text{se } B > G \end{cases} \quad \theta = \cos^{-1} \left\{ \frac{\frac{1}{2}[(R - G) + (R - B)]}{[(R - G)^2 + (R - B)(G - B)]^{1/2}} \right\}$$

La componente S è:

$$S = 1 - \frac{3}{(R + G + B)} [\min(R, G, B)]$$

Infine la I si ottiene come:

$$I = \frac{1}{3}(R + G + B)$$

Scrivete uno script dal nome ex1.m in cui realizzate il filtraggio in frequenza lavorando solo sulla componente di intensità, quindi usate la funzione contenuta nel file hsi2rgb.m per ritornare nello spazio RGB e poter visualizzare il risultato dell'elaborazione.

EX. 2 Si vuole proteggere un'immagine inserendo una firma digitale al suo interno. Scrivete uno script ex3.m in cui inserite l'immagine binaria firma.y nel bit-plane meno significativo dell'immagine upupa.y, avente dinamica pari a 256 e dimensioni 256×512 , e visualizzate il risultato.

A questo punto per verificare la robustezza dell'elaborazione, scrivete una funzione `function mse = comprimi(x,Q)` in cui usate lo standard JPEG per la compressione, estraete la firma digitale e visualizzate su di un grafico l'MSE tra firma originale e compressa al variare del fattore di qualità $Q = 80, 85, 90, 95, 100$.

Per valutare invece la robustezza rispetto al filtraggio, scrivete una funzione `function mse = filtra(x,B)` in cui applicate all'immagine marcata un filtro media aritmetica, estraete la firma digitale, quindi visualizzate su di un grafico l'MSE tra firma originale e filtrata al variare della dimensione del blocco del filtro $k = 3, 5, 7, 9, 11$.

N.B. Il codice necessario per richiamare le funzioni va scritto in un unico script dal nome ex2.m.

EX. 3 Nell'algoritmo di denoising Non Local Means (contenuto nel file NLMeans.m) il pixel ricostruito $\hat{x}_0$ si ottiene come media pesata $\sum_i w_i x_i$ dei pixel x_i appartenenti ad un'area di ricerca, dove il peso w_i dipende dall'errore quadratico medio (MSE) fra i blocchi centrati su x_0 e su x_i . Si vuole confrontare l'algoritmo con la seguente versione modificata: si individuano, nell'area di ricerca, i 10 blocchi con minore MSE rispetto a quello centrato su x_0 , e si usano solo i pixel centrali corrispondenti nel processo di media.

Per verificare le prestazioni, nello script ex3.m, prendete l'immagine barbara.gif, aggiungete rumore gaussiano con $\sigma = 20$, e filtrate la sezione di 100×100 pixel al centro dell'immagine con i tre algoritmi usando blocchi 5×5 e area di ricerca 20×20 . Valutate il PSNR della sezione corrotta e di quelle filtrate rispetto all'originale e mostrate le sezioni delle immagini corrispondenti.