

PROVA SCRITTA DI ELABORAZIONE DI SEGNALI MULTIMEDIALI del 12.6.12
(Ingegneria delle Telecomunicazioni)

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

EX. 1 Data l'immagine `viola.jpg`:

1. realizzate il color-balancing dell'immagine nello spazio CMY allo scopo di ridurre la quantità di magenta. A tal fine scrivete una funzione `function y = colorbalancing(x)`, mostrate l'immagine elaborata e confrontatela sia visivamente che in termini di MSE con l'immagine originale (contenuta nel file `viola_originale.jpg`);
2. a questo punto allo scopo di ridurre il contrasto fra la bambina e lo sfondo (troppo scuro) segmentate l'immagine in queste due regioni, effettuate un'equalizzazione dell'istogramma solo dello sfondo nell'intervallo $[0, 127]$ e se necessario applicate un filtraggio per migliorarne l'aspetto visivo.

EX. 2 Si vuole realizzare un filtraggio adattativo dell'immagine `mixed_noisy.y` (256×256 , float) corrotta da rumore moltiplicativo. In particolare la strategia da implementare consiste in una preliminare fase di classificazione dei pixel (omogenei e non omogenei) e un successivo filtraggio con finestra scorrevole che si adatta al tipo di pixel da elaborare. Scrivete una funzione dal prototipo `function mappa = classify(x)` in cui determinate la mappa binaria di classificazione applicando all'immagine `x` la seguente sequenza di elaborazioni:

1. filtraggio mediante filtro gaussiano con finestra 5×5 e deviazione standard pari a 1.2;
2. calcolo dell'immagine dei coefficienti di variazione locali $C = \sigma/\mu$ su finestre scorrevoli 9×9 ;
3. test su C : se risulta maggiore di 0.07 il pixel è etichettato come non omogeneo;
4. applicazione di un'adeguata operazione morfologica per eliminare pixel spuri.

A questo punto scrivete un'altra funzione dal prototipo `function y = denoise(x, mappa)` in cui usate un filtro media aritmetica di dimensioni 9×9 per i pixel omogenei e di dimensioni 3×3 per quelli non omogenei. Calcolate il PSNR globale (rispetto all'immagine pulita contenuta nel file `mixed.y`, 256×256 , uint8) confrontandolo con quello che si ottiene filtrando l'immagine con un filtro media aritmetica con finestra con dimensioni fisse 5×5 . Infine, confrontate anche i PSNR locali (calcolati sulla regione omogenea e non omogenea) nei due casi.

EX. 3 Nell'immagine `rice.tif` si vuole predire il generico pixel $x(m, n)$ a partire da quelli lungo le righe e colonne precedenti, e si provano diverse strategie di predizione. In tutte le strategie, il pixel in alto a sinistra è assegnato, $\tilde{x}(1, 1) = x(1, 1)$, quelli sulla prima riga sono predetti a partire da quelli che li precedono sulla riga stessa $\tilde{x}(1, n) = x(1, n - 1)$, e quelli sulla prima colonna dai precedenti sulla colonna. Per gli altri pixel utilizzate le strategie seguenti:

1. predizione lungo la riga se $\Delta_R < \Delta_C$ e lungo la colonna altrimenti, dove $\Delta_R = |x(m - 1, n) - x(m - 1, n - 1)|$ e $\Delta_C = |x(m, n - 1) - x(m - 1, n - 1)|$;
2. predizione pari alla media dei pixel precedenti lungo riga, colonna e diagonale secondo la formula $a(x(m - 1, n) + x(m, n - 1)) + (1 - 2a)x(m - 1, n - 1)$, dove il valore di a che minimizza l'MSE va trovato sperimentalmente (con passo 0.1) nell'intervallo $-1 \leq a \leq 1$.

Nello script `ex3.m` confrontate le due strategie calcolando e visualizzando l'errore di predizione $y(m, n) = \tilde{x}(m, n) - x(m, n)$ e valutandone il valore quadratico medio.