

# PROGETTO MATLAB **Commentato**

09 febbraio 2012

Data una immagine I realizzare uno script Matlab e la relativa GUI che permetta di effettuare i passi descritti di seguito. La GUI deve avere la barra dei menu. La prima voce "file" deve avere come sottomenu "apri" e "salva" che permettano di fare le classiche operazioni abbinate a questi comandi. La seconda voce "trasformazioni" deve contenere come sottomenu:

1. "RGB -> YCbCr" che permetta di cambiare da RGB a YCbCr. Mostrare il risultato nella GUI. *Non usare le funzioni di Matlab per effettuare tale trasformazione.*
2. "segmenta" che per ogni intorno 3x3 di un dato pixel, permetta di verificare che ci siano più di 5 pixel con valore Y superiore alla media dell'intorno. In questo caso sostituisce al pixel il valore medio dell'intorno. Altrimenti lascia il valore originario. Non devono essere alterati i valori della crominanza. **Alla media partecipano tutti e 9 i pixel. Mostrare la nuova Y (chiamata Ys) nella GUI.** *Non usare le funzioni di Matlab per effettuare tale trasformazione.*
3. "selezione" che cliccando su un punto dell'immagine si includono nella selezione tutti i pixel che hanno valore Y segmentato inferiore ad una soglia T (**inserita dall'utente tramite l'interfaccia**). Tra tutti i pixel che soddisfano la legge, si tengano solo quelli adiacenti al pixel selezionato o adiacenti ad un suo adiacente (il risultato deve essere simile alla "bacchetta magica" di Photoshop). Mostrare il risultato nella GUI. *Non usare le funzioni di Matlab per effettuare tale trasformazione.*

## **Commento:**

**Nella immagine Ys (cioè Y segmentato) ci saranno alcuni pixel che hanno valore inferiore alla T e altri no. Tra quelli che hanno valore inferiore, ce ne saranno alcuni raggruppati e altri lontani. Voi dovrete scegliere solo quei pixel che fanno parte del gruppetto in cui si trova il pixel selezionato dall'utente con un click. Stabilire quali sono vicini e quali no è un problema non molto semplice, ma fattibile. Partendo dal punto cliccato si verificano gli 8 punti dell'intorno. Quelli che soddisfano la legge si scrivono in una coda e gli altri no. Si prende il punto in testa alla coda e si riverifica il suo intorno. Quelli che soddisfano la legge vanno in coda e gli altri no. E si continua così fino a quando la coda non si svuota. Attenzione, tenete traccia dei pixel già visitati nei vari intorni, altrimenti fate il controllo troppe volte e potete anche andare in loop. Usate una matrice apposita.**

**L'output YL è una nuova immagine in cui alcuni pixel soddisfano la legge hanno valore 255 e gli altri che non soddisfano la legge hanno valore 0. In sostanza in questa matrice si vedrà UNA sola macchia bianca.**

4. "traforma" che in tutti i pixel selezionati al punto precedente (**quelli che hanno 255 in YL**), si sostituisca il valore della crominanza Cb con Cb/2 e Cr con Cr\*2. Mostrare il risultato nella GUI. *Non usare le funzioni di Matlab per effettuare tale trasformazione.*

**Adesso abbiamo nuovi valori di Cb (che non era mai stata modificata prima) e di Cr (che non era mai stata modificata prima). Chiamiamo queste matrici Cb1 e Cr1.**

5. “YcbCr -> RGB” che trasforma le nuove matrici YCbCr in RGB. Mostrare il risultato nella GUI. *Non usare le funzioni di Matlab per effettuare tale trasformazione.*

Si devono usare come YCbCr Ys, Cb1 e Cr1. I nuovi valori RGB potrebbero essere fuori range. Normalizzateli!

6. “PSNR” che calcola il PSNR tra l’immagine in input I e quella ottenuta al punto 5. Stampare il valore calcolato. *Non usare le funzioni di Matlab per effettuare tale trasformazione.*

Attenzione. Nel menu “trasformazioni” le voci non devono essere tutte attive al momento dell’apertura dell’interfaccia. Esse si devono abilitare mano a mano che il passaggio precedente è stato completato.

### **Istruzioni finali**

Tempo a disposizione 48 ore.

E’ essenziale che il file sia rinominato con il proprio cognome e numero di matricola (esempio stanco\_667000000.zip).

Il progetto è individuale.

Debbono essere zippati in un unico file zip i seguenti:

- a) tutti i file sorgenti adoperati
- b) una descrizione in formato pdf del progetto svolto. Lo scopo è aiutare il docente a comprendere la logica di progettazione adottata.
- c) l’indicazione di quale istruzione usare per fare funzionare il programma.

Spedire un file .zip che contenga il progetto a **fstanco@dmi.unict.it** entro le 12 del giorno 11 febbraio 2012.

I risultati e i dettagli per concludere l’esame saranno resi noti sul forum.