

Esame del corso di Tecniche Avanzate per il Trattamento delle Immagini

TATI

Data: 29 Giugno 2009

Es.1. [4 Pt] Si desidera realizzare un sistema di visione industriale in grado di individuare il livello di riempimento di liquido all'interno di un flacone. Indicate un possibile approccio al problema elencandone punti di forza e punti deboli.

Es.2. [6pt] Si descriva il criterio adottato da Harris (e basato sulla matrice dei momenti di secondo ordine) per l'identificazione dei corner; Si descrivano, in particolare, le condizioni necessarie in tale metodo per la discriminazione tra edge e corner.



Es.3. [6 pt] Si vuole localizzare il logo delle olimpiadi riportato a sinistra all'interno di un'immagine. Tale logo potrà traslare e variare di scala all'interno dell'immagine ma non subirà rotazioni. Indicare come si potrebbe utilizzare la trasformata di Hough applicata alle circonferenze per localizzare tale logo.

Es.4. [5 pt] Si assuma un modello di rumore con distribuzione uniforme tra -1 e 1:

$$p(z) = \begin{cases} 1/2 & \text{per } -1 \leq z \leq 1 \\ 0 & \text{altrimenti} \end{cases}$$

si descriva come è possibile definire uno stimatore a massima

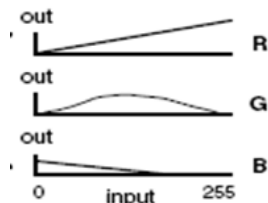
verosimiglianza per il valor medio.

Es.5 [5 Pt] Data un'immagine truecolor memorizzata in un file immagine.bmp ed avente i dati rappresentati ad 8bit per ogni piano di colore si implementino i seguenti punti mediante codice Matlab:

- Leggere, caricare nel workspace e visualizzare l'immagine.
 - convertirla a 256 toni di grigio.
 - Convertirla in Bianco e nero e con una soglia pari a 0.5.
- Rimuovere gli outlier definiti come un singolo pixel utilizzando un filtro mediano.

Es.6 [6 Pt] Data un'immagine truecolor memorizzata in un file immagine.bmp ed avente i dati rappresentati ad 8bit per ogni piano di colore si implementino i seguenti punti mediante codice Matlab:

- Leggere, caricare nel workspace e visualizzare l'immagine.
- convertirla a 256 toni di grigio.
- Applicare la seguente legge di trasformazione per portarla ad un'immagine agli pseudocolori e visualizzarla



Elenco di alcune
funzioni Matlab:

```
mean  
imerode  
end  
im2bw  
figure  
eig  
repmat  
rgb2gray  
imread  
title  
sqrt  
getimage  
strel  
diag  
imdilate  
imshow  
rgb2ind  
size
```

Soluzioni

- 1) La matrice corrisponde ad una trasformazione affine con una variazione di scala pari a 10 (ingrandimento) una rotazione di 45° in senso antiorario ed una traslazione di -5 lungo le ascisse e 5 lungo le ordinate (le trasformazioni sono eseguite nel suddetto ordine):

$$T = t \cdot R \cdot S = \begin{bmatrix} 1 & 0 & -5 \\ 0 & 1 & 5 \end{bmatrix} \cdot \begin{bmatrix} \cos \frac{\pi}{4} & -\sin \frac{\pi}{4} & 0 \\ \sin \frac{\pi}{4} & \cos \frac{\pi}{4} & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 10 & 0 & 0 \\ 0 & 10 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 10 \frac{\sqrt{2}}{2} & -10 \frac{\sqrt{2}}{2} & -5 \\ 10 \frac{\sqrt{2}}{2} & 10 \frac{\sqrt{2}}{2} & 5 \end{bmatrix}$$

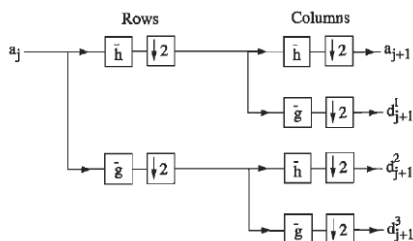
- 2) Si intende effettuare una convoluzione bidimensionale tra la matrice che rappresenta l'immagine e la matrice 3x3 indicata. In particolare tale matrice corrisponde al criterio di Sobel per l'estrazione di edge orizzontali. Tale filtro è costituito da un andamento passa-basso orizzontale e passa alto verticale. L'effetto sulla trasformata di Fourier 2D sarà quello di attenuare le componenti alle alte frequenze orizzontali e enfatizzare le componenti alle alte

frequenze verticali; in particolare, il filtro risulta essere separabile: $\begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix} \cdot [1 \ 2 \ 1]$. Per quanto

riguarda l'implementazione in Matlab si faccia riferimento alla funzione *imfilter* ed al materiale delle esercitazioni presenti sul sito.

- 3) Utilizzando il solito schema di decomposizione si ottiene al primo passaggio $\begin{matrix} 5 & 15 \\ -5 & -5 \end{matrix}$ ed al

secondo passaggio, secondo lo schema sottoriportato, $\begin{matrix} 10 & 5 \\ -5 & 0 \end{matrix}$



- 4) La trasformazione applicata per ridurre la dimensionalità delle feature in modo supervisionato (conoscenza a priori delle classi di appartenenza dei dati) sarà:

$$y = A^T x$$

Dove:

A (NxM) matrice di trasformazione le cui colonne sono gli autovettori corrisp. agli M autovalori più alti della matrice:

$$C = \text{inv}(S_w) * S_b$$

- S_w matrice della varianza complessiva intra-cluster

$$S_w = \sum_j p_j \text{Cov}_j$$

$$p_j = \text{prob. a priori classe } j$$

$$\text{Cov}_j = \text{covarianza dati classe } j$$

- S_b matrice della varianza complessiva inter-cluster

$$S_b = \sum_j (\mu_j - \mu) * (\mu_j - \mu)^T$$

$$\mu_j = \text{centroide dati classe } j$$

$\mu = \text{centroide complessivo dati}$

■ Si calcolano i rispettivi centroidi e le matrici S_w e S_b

```
mu_1 = mean(t_1, 1);
mu_2 = mean(t_2, 1);
mu_3 = size(t_1,1)/(size(t_1,1)+size(t_2,2))*mu_1 +
size(t_2,1)/(size(t_1,1)+size(t_2,2))*mu_2;

Cov_1 = (t_1-repmat(mu_1, size(t_1, 1), 1))' * (t_1-repmat(mu_1, size(t_1, 1), 1));

Cov_2 = (t_2-repmat(mu_2, size(t_2, 1), 1))' * (t_2-repmat(mu_2, size(t_2, 1), 1));

S_w = size(t_1,1)/(size(t_1,1)+size(t_2,2))*Cov_1 +
size(t_2,1)/(size(t_1,1)+size(t_2,2))*Cov_2;
S_b = (mu_1 - mu_3)'*(mu_1 - mu_3) + (mu_2 - mu_3)'*(mu_2 - mu_3);
```

■ Si calcola C , i suoi autovettori ed autovalori

```
C = S_b/S_w;
[V D] = eig(C);
```

■ Si calcola la matrice di trasformazione composta dall'autovettore associato all'autovalore più alto

```
P = diag(1./sqrt(D(end,end)))*V(:,end)';
```

■ Si procede alla trasformazione dei dati delle singole classi

```
t_r_1 = P*(t_1 - repmat(mu_3', 1, size(t_1,2)));
t_r_2 = P*(t_2 - repmat(mu_3', 1, size(t_2,2)));
```

5)

- `img_rgb = imread('immagine.bmp');`
`figure; imshow(img_rgb);`
(oppure: `figure`
`imshow('immagine.bmp');`
`img_rgb = getimage;`)
- `[img_ind map_ind] = rgb2ind(img_rgb, 16, 'nodither');`
`figure; imshow(img_ind, map_ind); title('Conversione Truecolor --> Indexed (mappa ad 8 valori)');`
- `imgray= rgb2gray(img_rgb);`
- `negativo=255- imgray;`
`figure; imshow(negativo); title('Conversione Truecolor --> Gray-scale -> negativo');`