

DOMANDE TEORIA ESAME

FONDAMENTI DI ELABORAZIONE DI IMMAGINE

SEZIONE 1 - OPERAZIONI SULLE IMMAGINI

1. Cosa si intende per **LookupTable (LUT)**? A cosa può servire?

La LookupTable è una struttura dati di supporto che è utile utilizzare quando il set di valori che i pixel di un'immagine possono assumere è significativamente inferiore al numero di pixel dell'immagine stessa. Per cui in questa struttura dati, che è essenzialmente un'array, si vanno a memorizzare i risultati pre-calcolati per ogni possibile valore di input dei pixel in modo da rendere l'elaborazione dell'immagine più efficiente.

Può essere adoperata in operazioni quali ad esempio contrast stretching, equalizzazione istogramma, variazione della luminosità, binarizzazione e più in generale quando dobbiamo lavorare su immagini a cui vadano applicate funzioni di mapping che permettano di calcolare il valore di un pixel in output a partire dal valore di un pixel in input.

2. Quali **problemi** presenta l'applicazione affine a un'immagine tramite mapping diretto? Com'è possibile risolverli?

Con il mapping diretto si vanno a mappare ogni pixel della vecchia immagine nella nuova immagine. Il mapping diretto è dato dalla seguente funzione:

$$\begin{bmatrix} x_{new} \\ y_{new} \end{bmatrix} = \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix} \cdot \begin{bmatrix} s & 0 \\ 0 & s \end{bmatrix} \cdot \begin{bmatrix} x_{old} \\ y_{old} \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}$$

$[tx, ty]$: è la traslazione

θ : è la rotazione

s : la scala

Con il mapping diretto si hanno i seguenti **problemi**:

- Valori dei nuovi pixel non necessariamente interi (approssimazione);
- Alcuni pixel vengono mappati al di fuori della nuova immagine;
- Alcuni pixel della nuova immagine non sono coperti ("buchi");

Un modo efficace di risolvere tali problemi consiste nell'eseguire la scansione della nuova immagine e, per ogni pixel $[X_{new}, Y_{new}]$, determinare il punto di riferimento $[X_{old}, Y_{old}]$ nella vecchia immagine attraverso la funzione precedente ma inversa. Il mapping inverso è dato per cui dalla seguente funzione:

$$\begin{bmatrix} x_{old} \\ y_{old} \end{bmatrix} = \begin{bmatrix} 1/s & 0 \\ 0 & 1/s \end{bmatrix} \cdot \begin{bmatrix} \cos(-\theta) & \sin(-\theta) \\ -\sin(-\theta) & \cos(-\theta) \end{bmatrix} \cdot \left(\begin{bmatrix} x_{new} \\ y_{new} \end{bmatrix} - \begin{bmatrix} t_x \\ t_y \end{bmatrix} \right)$$

Il punto di riferimento $[X_{old}, Y_{old}]$, che è in coordinate continue (floating point), potrebbe cadere:

- fuori dalla vecchia immagine: tipicamente si utilizza un valore fisso (colore dello sfondo se noto, oppure nero);
- su di un pixel della vecchia immagine: se ne copia l'intensità
- in una posizione intermedia tra 4 pixel della vecchia immagine: si può applicare una tecnica di interpolazione.

Con l'interpolazione di Lagrange il valore di intensità del nuovo pixel $[X_{new}, Y_{new}]$ viene calcolato cercando il piano che meglio approssima i 4 pixel ai minimi quadrati

3. Che cos'è l'istogramma di un'immagine digitale? A cosa può servire?

L'istogramma di un'immagine digitale indica il numero di pixel dell'immagine per ciascun livello di grigio.

Con l'utilizzo dell'istogramma si possono ottenere informazioni utili, quali ad esempio:

- se la maggior parte dei valori sono "condensati" in una zona, ciò significa che l'immagine ha uno scarso contrasto;
- se nell'istogramma sono predominanti le basse intensità, l'immagine è molto scura e viceversa;
- se i diversi oggetti in un'immagine hanno livelli di grigio differenti, l'istogramma può fornire un primo semplice meccanismo di classificazione. Esempio: un istogramma bimodale denota spesso la presenza di un oggetto abbastanza omogeneo su uno sfondo di luminosità pressoché costante.

Inoltre, definendo opportune funzioni di mapping f è possibile:

- aumentare il contrasto (espansione range dinamico)
- scurire/schiarire l'immagine
- evidenziare/nascondere dettagli
- equalizzare l'istogramma

4. Cosa si intende con "convoluzione di un'immagine con un filtro digitale"? Elencare alcune applicazioni.

La convoluzione è una operazione di media pesata locale.

Essa fa utilizzo di un filtro digitale: una maschera discreta di pesi che indicano come ogni elemento dell'immagine debba essere modificato sulla base del valore dei pixel vicini.

Sia F un filtro definito su una griglia $m \times m$ (m dispari);

L'applicazione di F a un'immagine I nel punto $[i, j]$ modifica il pixel $I[i, j]$ come segue:

$$I'[i, j] = \sum_{y=1}^m \sum_{x=1}^m (I[i + \lceil \frac{m}{2} \rceil - y, j + \lceil \frac{m}{2} \rceil - x] \cdot F[y, x])$$

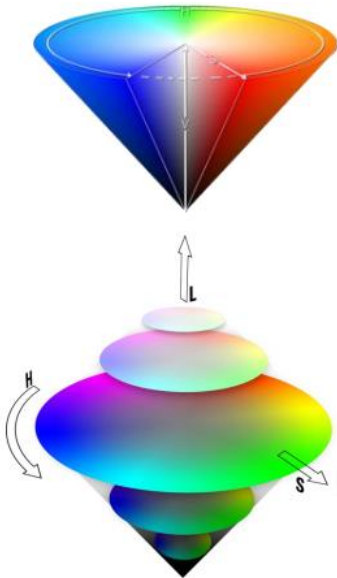
Alcune applicazioni:

- **Smoothing** (regolarizzazione): produce una sfocatura più o meno evidente, in grado di nascondere piccole imperfezioni e brusche variazioni di luminosità. Possono essere utili come passo iniziale prima di ulteriori elaborazioni;
- **Sharpening** (affilamento): evidenziano dettagli fini dell'immagine e le brusche variazioni di luminosità (contorni);
- **Estrazione contorno di oggetti** presenti in un'immagine tramite filtri quali Roberts, Prewitt e Sobel

5. Descrivere gli **spazi colore HSL/HSV** e illustrarne brevemente vantaggi e svantaggi.

Il modello HS* è basato sulle caratteristiche con cui un essere umano usualmente definisce un colore:

- **Tinta (Hue) → H**
- **Saturazione → S**
- **Luminosità → L oppure Valore (Value) → V**



Vantaggi

- Possibilità di specificare i colori in modo intuitivo
- Possono essere utilizzati più efficacemente per localizzazione e riconoscimento di pattern

Svantaggi:

- Non è adatto ad una descrizione quantitativa di un colore

HSV: Può essere rappresentato come un cono in cui l'asse verticale codifica V

HSL: Può essere rappresentato come un doppio cono in cui l'asse verticale codifica L. Esso meglio rappresenta i concetti di saturazione e luminosità

- Variando S ci si muove sempre da un tono di grigio (S=0) al colore completamente saturo (S=1)
- Variando L ci si muove sempre dal nero (L=0) al bianco (L=1)

SEZIONE 2 – ESTRAZIONE DEI BORDI E SEGMENTAZIONE

6. Che cosa si intende con **segmentazione degli oggetti dal background**? In quali casi è un'operazione semplice? In quali casi può essere molto complicata?

La segmentazione è la separazione di uno o più oggetti di interesse dallo sfondo (background).

• **Caso semplice:**

- Se lo sfondo e l'oggetto da estrarre sono omogenei e hanno gradazioni di grigio diverse, si procede con la scelta automatica della soglia a partire dall'istogramma: la presenza di due o più picchi nell'istogramma è spesso causata da oggetti con luminosità medie diverse. La soglia ottimale viene determinata come minimo locale tra i due picchi;
- Sottrazione dello sfondo: (se è noto a priori che lo sfondo e l'illuminazione sono costanti);
- Utilizzo del colore.
- Immagini binarizzate.

• **Caso complesso:**

Quando oggetti e sfondo non sono uniformi, la scelta della soglia globale è un'operazione molto critica.

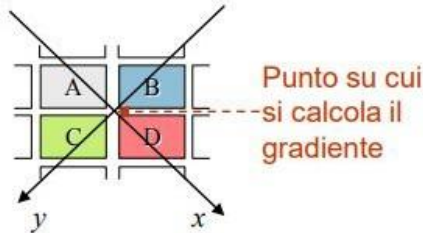
Ad esempio, se lo sfondo presenta un gradiente di illuminazione (ossia la luminosità varia gradualmente da una zona all'altra dell'immagine), l'approccio non può essere utilizzato con successo.

Una possibile soluzione è usare soglie locali, ovvero per ogni pixel la soglia è determinata considerando una porzione dell'immagine

7. Descrivere gli **operatori di Roberts** e illustrarne brevemente le applicazioni.

$$\mathbf{F}_x = \begin{bmatrix} +1 & 0 \\ 0 & -1 \end{bmatrix} \rightarrow \nabla x = D - A$$

$$\mathbf{F}_y = \begin{bmatrix} 0 & +1 \\ -1 & 0 \end{bmatrix} \rightarrow \nabla y = C - B$$



Gli operatori di Roberts vengono impiegati per il calcolo del gradiente allo scopo di estrarre contorni di oggetti da un'immagine.

In particolare, viene fatta una convoluzione con i seguenti filtri misurando il gradiente lungo assi ruotati di 45° rispetto agli assi dell'immagine: questo consente di calcolare le due componenti del gradiente nel medesimo punto.

Possono essere utilizzati anche per il calcolo dei massimi locali in direzione del gradiente ad esempio nel metodo Canny edge detector.

$$\|\nabla\| = \sqrt{\nabla x^2 + \nabla y^2} \cong |\nabla x| + |\nabla y|$$

$$\theta = \arctan_q(\nabla y, \nabla x) + \frac{\pi}{4}$$

Caratteristiche

- Pro: possono essere calcolati in modo rapido ed efficiente
- Contro: sono molto sensibili al rumore

8. Descrivere gli **operatori di Prewitt** ed illustrarne brevemente alcune applicazioni.

Gli operatori di Prewitt vengono impiegati per il calcolo del gradiente allo scopo di estrarre contorni di oggetti da un'immagine.

Possono essere utilizzati anche per il calcolo dei massimi locali in direzione del gradiente ad esempio nel metodo Canny edge detector.

Per il calcolo del gradiente viene eseguita una convoluzione con i seguenti filtri:

$$\mathbf{F}_x = \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix} \cdot \frac{1}{3}$$

$$\mathbf{F}_y = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix} \cdot \frac{1}{3}$$

$$\|\nabla\| = \sqrt{\nabla x^2 + \nabla y^2}$$

$$\theta = \arctan_q(\nabla y, \nabla x)$$

Caratteristiche

- Meno sensibili a variazioni di luce e rumore
- Simmetrici rispetto al punto di applicazione
- Assi x e y orientati in modo tradizionale

9. Descrivere gli **operatori di Sobel** ed illustrarne brevemente alcune applicazioni.

Wikipedia: L'operatore calcola il gradiente della luminosità dell'immagine in ciascun punto, trovando la direzione lungo la quale si ha il massimo incremento possibile dal chiaro allo scuro, e la velocità con cui avviene il cambiamento lungo questa direzione. Il risultato ottenuto fornisce una misura di quanto "bruscamente" oppure "gradualmente" l'immagine cambia in quel punto, e quindi della probabilità che quella parte di immagine rappresenti un contorno, e fornisce anche un'indicazione del probabile orientamento di quel contorno.

Gli operatori di Sobel vengono impiegati per il calcolo del gradiente allo scopo di estrarre contorni di oggetti da un'immagine. Possono essere utilizzati anche per il calcolo dei massimi locali in direzione del gradiente ad esempio nel metodo Canny edge detector.

Per il calcolo del gradiente viene eseguita una convoluzione con i seguenti filtri:

$$\mathbf{F}_x = \frac{1}{4} \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}, \quad \mathbf{F}_y = \frac{1}{4} \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}, \quad \|\nabla\| = \sqrt{\nabla x^2 + \nabla y^2}$$
$$\theta = \arctan_q(\nabla y, \nabla x)$$

Rispetto agli operatori di Prewitt, gli operatori di Sobel danno un peso maggiore al pixel centrale

Caratteristiche

- **Meno sensibili** a variazioni di luce e rumore
- **Simmetrici** rispetto al punto di applicazione
- Assi x e y orientati in **modo tradizionale**

10. Descrivere sinteticamente il metodo "Canny edge detector".

Il metodo Canny edge detector è impiegato per il riconoscimento di contorni in un'immagine.

L'approccio prevede **quattro fasi**:

1. **Smoothing gaussiano dell'immagine**

Ovvero gli elementi del filtro sono pesati secondo una funzione gaussiana (di cui se ne può regolare l'ampiezza);

2. **Calcolo del gradiente**

Dato che lo smoothing dovrebbe aver rimosso la maggior parte del rumore, l'implementazione più efficiente si avvale degli operatori di Roberts. Risultano tuttavia di più semplice applicazione gli operatori di Prewitt, in quanto non ci si deve preoccupare della rotazione degli assi di 45°;

3. **Soppressione dei non-massimi in direzione ortogonale all'edge**

In questa fase vengono eliminati dall'immagine modulo-gradiente i pixel che non sono massimi locali rispetto all'orientazione del gradiente;

4. **Selezione degli edge significativi mediante isteresi**

Al fine di selezionare solo gli edge significativi si utilizza il concetto di isteresi: vengono impiegate due soglie T1 e T2, con $T1 > T2$:

Sono inizialmente considerati validi solo i pixel in cui il modulo del gradiente è superiore a T1.

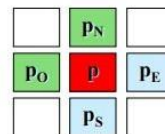
I pixel il cui modulo è inferiore a T1 ma superiore a T2 sono considerati validi solo se adiacenti a pixel validi.

SEZIONE 3 – TOPOLOGIA DIGITALE

11. Descrivere sinteticamente il funzionamento della **Trasformata Distanza** e alcune possibili applicazioni. La trasformata distanza di F rispetto a F^* è una replica di F in cui i pixel sono etichettati con il valore della loro distanza da F^* , calcolata secondo una data metrica.

■ Si considera il caso della metrica d_4 .

- La trasformata può essere calcolata con due semplici scansioni dell'immagine:
 - una diretta (dall'alto verso il basso da sinistra verso destra),
 - una inversa (dal basso verso l'alto da destra verso sinistra).
 - Durante le scansioni i pixel di F vengono trasformati; i pixel di F^* sono posti a zero (distanza nulla).
- $\forall p \in F$, il valore trasformato $\text{Img}'[p]$ è calcolato come segue:
 - Scansione diretta: la distanza viene propagata dai vicini in alto a sinistra (che sono già stati considerati).
 - $\text{Img}'[p] = \min \{ \text{Img}'[p_O], \text{Img}'[p_N] \} + 1$
 - Scansione inversa: la distanza viene aggiornata tenendo conto anche dei percorsi verso il basso e a destra.
 - $\text{Img}'[p] = \min \{ \text{Img}'[p_E] + 1, \text{Img}'[p_S] + 1, \text{Img}'[p] \}$



La Trasformata distanza può essere utilizzata per:

- la misura geometrica di lunghezze e spessori ad esempio;
- calcolo dello scheletro di un oggetto;
- template matching;
- robotica (ad esempio ricerca direzione di movimento ottimale e aggiramento ostacoli)

12. Descrivere il funzionamento **dell'algoritmo di etichettatura delle componenti connesse**.

■ In cosa consiste

- Individuare automaticamente le diverse componenti connesse in un'immagine, assegnando loro etichette (tipicamente numeriche)
- Procedura molto importante nell'analisi automatica di immagini

■ Algoritmo comunemente utilizzato:

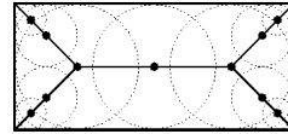
- 1) Si scandisce l'immagine e, per ogni pixel di foreground p , si considerano i pixel di foreground vicini già visitati:
 - se nessuno è etichettato, si assegna una nuova etichetta a p
 - se uno è etichettato, si assegna a p la stessa etichetta
 - se più di uno è etichettato, si assegna a p una delle etichette e si annotano le equivalenze
- 2) Si definisce un'unica etichetta per ogni insieme di etichette marcate come equivalenti e si effettua una seconda scansione assegnando le etichette finali

13. Descrivere l'algoritmo di Hilditch

Scheletro e Thinning

Definizioni

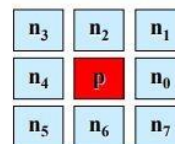
- Scheletro di un'immagine binaria (skeleton)
 - Preserva le **caratteristiche topologiche** dell'oggetto
 - Può essere definito come il **luogo dei centri dei cerchi, tangenti in almeno due punti**, che sono completamente contenuti nel foreground
 - Gli algoritmi che cercano di implementare direttamente tale definizione sono computazionalmente assai inefficienti
 - In genere si cerca di ottenere lo scheletro mediante algoritmi di thinning
- Thinning (assottigliamento)
 - Gli algoritmi di thinning procedono tipicamente in modo iterativo
 - Ad ogni iterazione eliminano da F (attribuendoli a F*) pixel di bordo che la cui cancellazione non altera la topologia locale, ossia:
 - Non rimuovono pixel terminali
 - Non eliminano pixel di connessione
 - Non causano erosione eccessiva
 - Le iterazioni proseguono fino a quando non vi sono più pixel da eliminare



Thinning – Algoritmo di Hilditch

Uno dei più noti metodi di thinning:

- Sia $A(p)$ il numero di transizioni *Background* → *Foreground* nella sequenza ordinata: $n_0, n_1, n_2, n_3, n_4, n_5, n_6, n_7, n_0$
- Sia $B(p)$ il numero di pixel appartenenti a F nell'8-intorno di p.



$A(p)=1, B(p)=2$



$A(p)=2, B(p)=2$

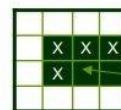
Condizioni

- a) $2 \leq B(p) \leq 6$
- b) $A(p) = 1$
- c) $n_2 = \text{back}$ or $n_0 = \text{back}$ or $n_4 = \text{back}$ or $A(n_2) > 1$
- d) $n_2 = \text{back}$ or $n_0 = \text{back}$ or $n_6 = \text{back}$ or $A(n_0) > 1$

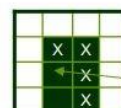
Ad ogni passo:

- l'algoritmo considera tutti i pixel in modo parallelo (l'operazione su un pixel non dipende da quelli già esaminati)
- i pixel p che rispettano le 4 condizioni a,b,c,d sono cancellati

L'algoritmo termina quando nessun nuovo pixel può essere cancellato



Pixel non eliminato per la condizione c)



Pixel non eliminato per la condizione d)

SEZIONE 4 – MORFOLOGIA MATEMATICA

14. Com'è possibile, con la morfologia matematica, **rimuovere piccole componenti connesse da un'immagine?**

È possibile tramite l'operazione di **apertura** se si utilizza come elemento strutturante un elemento che è più grande della componente connessa che si intende eliminare. Essa consiste in un'operazione di erosione seguita da una dilatazione, che sono le operazioni di base della morfologia matematica. L'apertura quindi separa oggetti debolmente connessi e rimuove regioni piccole.

$$F \circ S = (F \ominus S) \oplus S$$

15. Com'è possibile, con la morfologia matematica, **chiudere piccoli "buchi" in un'immagine binaria?**

È possibile tramite l'operazione di **chiusura** se si utilizza come elemento strutturante un elemento che è più grande dei "buchi" che si intendono chiudere. Essa consiste in un'operazione di dilatazione seguita da una erosione, che sono le operazioni di base della morfologia matematica. La chiusura quindi riempie buchi e piccole concavità e rafforza la connessione di regioni unite debolmente.

$$F \bullet S = (F \oplus S) \ominus S$$

16. Definire formalmente i **due operatori di base della morfologia matematica** e illustrarne brevemente il funzionamento.

I due operatori sono:

■ **Erosione** (Erosion)

- La nuova immagine è l'insieme dei pixel tali che, traslando in essi S , l'intero elemento strutturante è contenuto in F

$$F \ominus S = \{ \mathbf{q} \mid (S)_{\mathbf{q}} \subseteq F \}$$

■ **Dilatazione** (Dilation)

- La nuova immagine è l'insieme dei pixel tali che, traslando in essi S^r , almeno uno dei suoi elementi è sovrapposto a F

$$F \oplus S = \{ \mathbf{q} \mid (S^r)_{\mathbf{q}} \cap F \neq \emptyset \}$$

17. Che cosa è la “**hit-or-miss transform**”? A cosa può servire?

È un'operazione che rileva una data configurazione in una immagine binaria, usando l'operatore di erosione e due elementi strutturanti disgiunti. In particolare, localizza i punti in cui S_1 è contenuto nel foreground e S_2 nello sfondo (background).

Può servire per il pattern matching.

$$F * S = (F \ominus S_1) \cap (F^c \ominus S_2)$$

$$S = (S_1, S_2), S_1 \cap S_2 = \emptyset$$

SEZIONE 5 – TEMPLATE MATCHING

18. Si dia una definizione di “**template matching rigido**” e se ne discutano pregi e difetti.

Il termine template matching fa riferimento alla “ricerca” di un template all'interno di un'immagine con l'obiettivo di determinare se l'immagine contiene l'oggetto (match) e in quale posizione appare nell'immagine.

Si procede come segue:

1. il template T è costituito da un oggetto rigido;
2. T viene sovrapposto all'immagine in tutte le possibili posizioni (rispetto agli assi X e Y);
3. Per ogni istanza T_i viene calcolato il grado di similarità massimizzando la correlazione con la porzione di immagine “coperta” dal template.

Svantaggi:

- difficile gestione di **pattern deformabili**; esempio: localizzazione di un componente elettronico che può subire diverse deformazioni;
- **complessità computazionale**: il numero di operazioni richieste cresce linearmente con il numero di istanze e con il numero di pixel dell'immagine e del template;

Adottando però un approccio **multi-risoluzione** si riescono ad ottenere i seguenti vantaggi:

- “Scrematura” ai livelli iniziali;
- Una perfezione della localizzazione;
- Filtrazione delle “false somiglianze” ai livelli successivi;
- La complessità computazionale si riduce notevolmente.

19. Perché il confronto “**pixel a pixel**” generalmente non è efficace in applicazioni reali di ricerca di un oggetto all'interno di un'immagine?

Il confronto diretto di immagini, mediante distanza fra tali vettori nello spazio, in generale non funziona per una serie di ragioni, fra cui:

- differenze di traslazione, rotazione, scala e prospettiva;
- deformazione e variabilità dei pattern;
- cambiamenti di illuminazione;
- presenza di rumore nelle immagini e utilizzo di tecniche di acquisizione diverse.