

Segmentazione per discontinuità

1 Determinazione di punti isolati

Il caso più semplice di segmentazione è la determinazione dei **punti isolati**, cioè di singoli pixel, o piccole aree di forma pressoché circolare, con valore diverso dallo sfondo.

Tali punti definiscono una discontinuità in tutte le direzioni, quindi vengono segnalati in modo molto evidente dagli operatori basati sulla derivata seconda. Per segmentarli, è allora sufficiente:

- applicare l'operatore Laplaciano (sull'intorno a 8) all'immagine;
- eseguire una sogliatura.

Un esempio di applicazione di questa procedura è l'individuazione di determinati tipi di difetti nei materiali.

2 Estrazione di linee

Per estrarre linee orientate in direzioni specifiche, si utilizzano degli *operatori derivativi direzionali*:

-1	-1	-1	-1	2	-1	-1	-1	2	2	-1	-1
2	2	2	-1	2	-1	-1	2	-1	-1	2	-1
-1	-1	-1	-1	2	-1	2	-1	-1	-1	-1	2
orizzontale			verticale			diagonale +45°			diagonale -45°		

Avendo somme dei coefficienti pari a 0, essi danno risposte nulle nelle zone omogenee dell'immagine (così come gli altri operatori derivativi).

La direzione di questi operatori è data dalla disposizione dei coefficienti con valore 2: per una linea su uno sfondo costante, il risultato è massimo in corrispondenza delle posizioni nelle quali tale linea risulta disposta lungo la riga di coefficienti a 2 della maschera.

- Se si vogliono estrarre le linee in tutte le direzioni, si passano sull'immagine tutte e 4 le maschere. La direzione della linea a cui appartiene un certo pixel può poi essere ricavata determinando quale delle 4 maschere ha dato la risposta maggiore nella posizione corrispondente a tale pixel.
- Se, invece, si è interessati a individuare solo linee in una specifica direzione, nota a priori, è sufficiente applicare l'operatore corrispondente a tale direzione ed effettuare una sogliatura.

3 Bordi e contorni

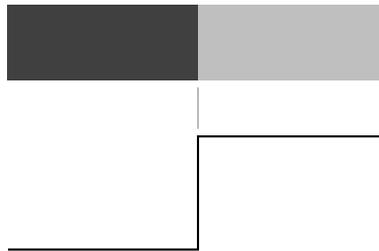
Il **contorno** di un oggetto è un concetto *globale*. Invece, un **bordo (edge)** è un concetto *locale*: un insieme di pixel connessi che risiedono nelle zone di confine tra due regioni, in cui l'intensità subisce brusche variazioni (cioè è presente una discontinuità).

Un bordo è quindi un frammento di contorno, perché non è detto che il contorno di un oggetto sia costituito da un singolo bordo continuo. Allora, per ottenere un contorno:

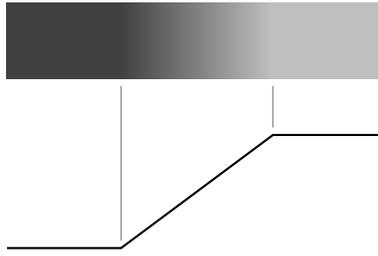
1. si individuano gli edges, mediante operatori derivativi;
2. si effettuano delle operazioni di *edge linking*, che collegano i bordi individuati per formare un contorno.

3.1 Tipi di edge

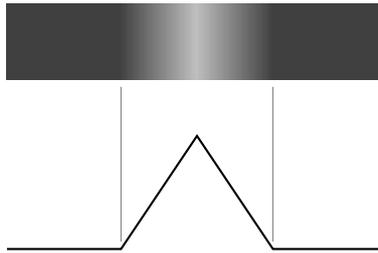
Idealmente, un edge è un insieme di pixel connessi, posti ortogonalmente alla transizione dei livelli di grigio. Questo tipo di edge si chiama *step*.



Realisticamente, a causa di imperfezioni nel processo di acquisizione, gli edge subiscono spesso un effetto di sfocamento, e perciò tendono ad avere un profilo a *rampa*: più la rampa è lunga e più l'edge è spesso.



Come caso particolare, una struttura lineare presente nell'immagine determina un edge di tipo *roof*.



Siccome in una stessa immagine possono essere presenti tipi diversi di edge, è necessario usare operatori di edge detection robusti, che siano in grado di gestirli tutti.

4 Rumore

Gli operatori derivativi, in particolare quelli basati sulla derivata seconda, sono sensibili al rumore. Di conseguenza, in presenza di rumore, può essere difficile/impossibile effettuare la soglia dei bordi sul risultato di tali operatori. In questi casi, è quindi utile effettuare un'operazione di smoothing, prima di applicare l'operatore derivativo.

5 Operatori basati sul gradiente

Gli operatori basati sulla derivata prima implementano varie approssimazioni del gradiente.

$$\nabla f = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

In particolare, per determinare i contorni di un oggetto, oltre al modulo del gradiente

$$|\nabla f| = \sqrt{G_x^2 + G_y^2} \approx |G_x| + |G_y|$$

può essere utile anche la **direzione del gradiente**:

$$\alpha(x, y) = \tan^{-1} \left(\frac{G_y}{G_x} \right)$$

se pixel vicini hanno gradienti con direzioni simili, è probabile che appartengano entrambi al contorno.

I principali operatori di questa categoria sono:

- operatore di *Roberts*;

-1	0
0	1

0	-1
1	0

- operatore di *Sobel*,

-1	-2	-1
0	0	0
1	2	1

-1	0	1
-2	0	2
-1	0	1

che esiste anche in versione diagonale (la quale enfatizza prevalentemente, appunto, gli edges diagonali);

-2	-1	0
-1	0	1
0	1	2

0	1	2
-1	0	1
-2	-1	0

- operatore di *Prewitt*, simile a Sobel ma meno “sosticato” (perché non dà peso maggiore ai pixel più vicini al centro).

-1	-1	-1
0	0	0
1	1	1

-1	0	1
-1	0	1
-1	0	1

In generale, questi operatori producono bordi piuttosto spessi, che “resistono” meglio a eventuali operazioni di smoothing successive, ma non sono molto precisi.

6 Laplaciano

Gli operatori di edge detection basati sulla derivata seconda presentano alcuni vantaggi rispetto a quelli basati sul gradiente:

- in corrispondenza degli edge, producono degli *zero-crossing*, che sono facili da individuare, e tendono a formare catene abbastanza continue (indipendentemente dall'intensità del contorno);
- segnalano con precisione l'inizio e la fine delle transizioni (permettendo quindi, ad esempio, di posizionare il bordo a metà tra inizio e fine).

Il principale operatore basato sulla derivata seconda è il *Laplaciano*:

0	-1	0	-1	-1	-1
-1	4	-1	-1	8	-1
0	-1	0	-1	-1	-1

Esso, però, è molto sensibile al rumore: per poter ottenere risultati affidabili, è spesso necessario effettuare uno smoothing prima di applicarlo.

7 Laplaciano della Gaussiana

Il **Laplaciano della Gaussiana** integra un filtro gaussiano (smoothing) e il Laplaciano in un'unica maschera: in questo modo, si riduce la sensibilità al rumore del Laplaciano, mantenendone però i vantaggi.

Infatti, sfruttando la linearità degli operatori Gaussiano e Laplaciano, è possibile comporli:

- Gaussiana bidimensionale:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

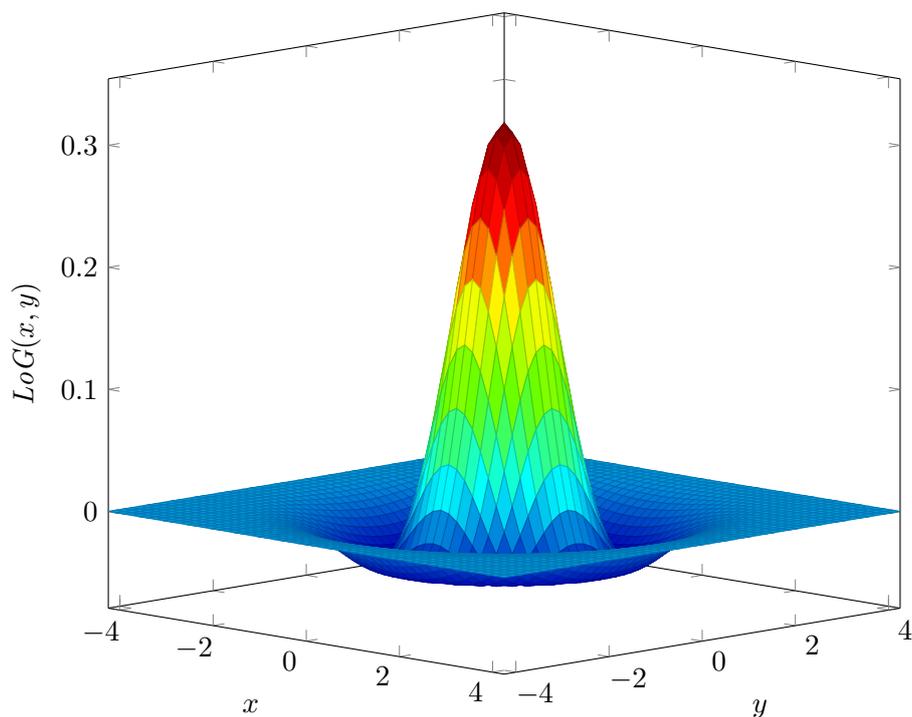
- Laplaciano:

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

- Laplaciano della Gaussiana:

$$LoG(x, y) = \nabla^2 G(x, y) = -\frac{1}{\pi\sigma^2} \left[1 - \frac{x^2 + y^2}{2\sigma^2} \right] e^{-\frac{x^2+y^2}{2\sigma^2}}$$

La funzione così ottenuta ha una forma a “cappello messicano”:¹



Mediante un campionamento, si ricava poi una maschera convolutiva (qui mostrata con valore centrale positivo, ma, equivalentemente, si può usare una maschera con valore centrale negativo, cambiando i segni di tutti i coefficienti):

0	0	-1	0	0
0	-1	-2	-1	0
-1	-2	16	-2	-1
0	-1	-2	-1	0
0	0	-1	0	0

Per ottenere buoni risultati su immagini con rumore, occorre definire maschere molto ampie.

¹La formula riportata sopra definisce una funzione con valore centrale negativo ($LoG(0,0) < 0$), che ha quindi una forma a “cappello messicano capovolto”. Per ricavare la versione raffigurata nel grafico, è sufficiente cambiare il segno (ottenendo, così, un operatore che dà risultati equivalenti, ma con segni opposti).

L'uso di un nucleo convolutivo unico è comodo ed efficiente, ma, invece, applicando separatamente il filtro gaussiano e il Laplaciano, si avrebbe maggior controllo sulla funzione di smoothing.

8 Edge linking

Idealmente, gli operatori derivativi dovrebbero restituire direttamente contorni continui. In realtà, a causa di rumore, differenze di illuminazione, ecc., si ottengono solitamente dei bordi spezzati. Allora, per ricostruire i contorni degli oggetti, si adottano delle tecniche di **edge linking**.

Un esempio di tecnica di linking locale è la seguente: dati due pixel di bordo, individuati con le tecniche basate sul gradiente, si decide se essi appartengono allo stesso contorno in base alla similarità di:

- intensità del modulo del gradiente;

$$|\nabla f(x, y) - \nabla f(x_0, y_0)| \leq E$$

- direzione del vettore gradiente.

$$|\alpha f(x, y) - \alpha f(x_0, y_0)| \leq A$$

In casi complessi, può essere invece necessario ricorrere ad approcci di linking globali.

9 Metodo di Canny

Il **metodo di Canny** è un metodo di edge detection composto da quattro fasi:

1. applicazione di un filtro di smoothing gaussiano;
2. calcolo del gradiente, mediante un filtro basato sulla derivata prima;
3. soppressione dei non-massimi lungo la direzione ortogonale all'edge (cioè i non-massimi vengono azzerati, e si tengono solo i massimi);
4. selezione degli edge significativi mediante una sogliaatura con *isteresi*, che si basa su due soglie, T_1 e T_2 , con $T_1 > T_2$:
 - a) inizialmente, sono considerati validi i pixel con valore (in modulo) $> T_1$ (e quindi anche $> T_2$);
 - b) i pixel con valore $< T_1$ ma $> T_2$ sono considerati validi solo se sono adiacenti a pixel con valore $> T_1$.

Siccome vengono effettuate delle operazioni sofisticate sul risultato dell'operatore derivativo applicato al passo 2, non è necessario che esso sia particolarmente raffinato: si può usare anche solo un operatore semplice e veloce, come ad esempio l'operatore di Roberts.

Complessivamente, i parametri da impostare per applicare il metodo di Canny sono:

- il σ della gaussiana;
- la dimensione della maschera del filtro gaussiano;
- le due soglie per l'isteresi (T_1 e T_2).