

Convoluzione

Convoluzione

Premessa

L'operazione di convoluzione richiede la presenza in gioco di due attori principali:

- ▶ L'immagine sorgente.
- ▶ Il filtro di convoluzione, o *kernel*.

2	-1	8
5	2	-2
7	1	0



Convoluzione

Il kernel

- ▶ È una raccolta di valori.
- ▶ Questi valori possono interi o decimali.
- ▶ La raccolta può disporsi in due dimensioni (*kernel 2D*), tre dimensioni (*kernel 3D*) o più in generale ad n dimensioni.
- ▶ Per l'elaborazione immagini sono comuni i primi due casi.



2	-1	8
5	2	-2
7	1	0

Convoluzione

Il kernel

Durante l'operazione di convoluzione, il kernel viene applicato all'immagine sorgente.

2	-1	8
5	2	-2
7	1	0

Generalmente, il kernel:

- È più piccolo dell'immagine sorgente.
- È di forma quadrata.
- Ha un numero dispari di valori in larghezza e altezza: 3, 5, 7...

0	0	0
0	1	0
0	0	0

0	-1	0
-1	5	-1
0	-1	0

-1	-1	-1
-1	8	-1
-1	-1	-1

Non è, comunque, impossibile trovare kernel rettangolari e/o con un numero pari di elementi in larghezza/lunghezza.

Convoluzione

Il kernel

La dimensione del kernel è infatti una delle sue caratteristiche distintive.

Quest'ultima, semplicemente, indica il numero di pixel dell'immagine sorgente che si dovranno andare a considerare nel momento in cui il kernel vi si andrà a sovrapporre.

1	1	0
0	1	0

14	-5	3
2	0	-4
2	7	9

6	-1
2	-2
-1	5
0	-1

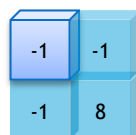
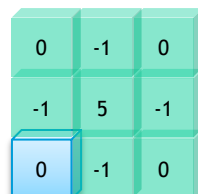
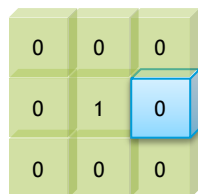
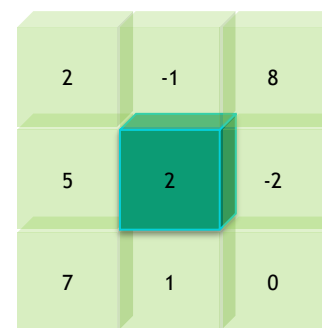
Convoluzione

Il kernel

Del kernel si identifica un centro. Questo centro sarà il punto di ‘aggancio’ del kernel stesso con i pixel dell’immagine sorgente.

In un kernel quadrato e di dimensioni dispari il centro si trova, *se non specificato diversamente*, nella posizione in cui ci si aspetterebbe di trovarlo. È comunque possibile:

- Specificare una diversa posizione per il centro.
- Specificare il centro per kernel di dimensioni pari.



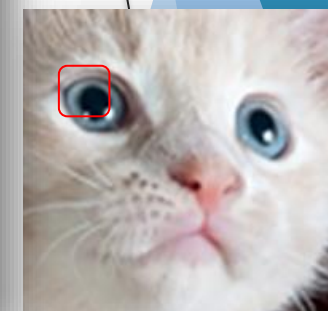
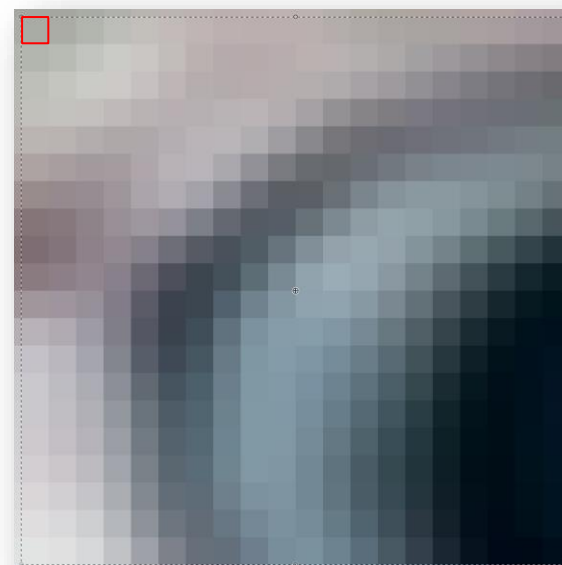
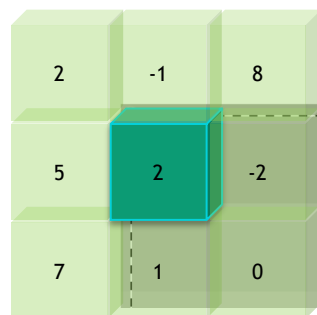
Per il fatto che il centro viene utilizzato come elemento di contatto fra kernel e immagine, non è inusuale vederlo chiamato anche ***ancora***.

Convoluzione

Funzionamento

Compreso come il kernel si definisce, l'*applicazione* di questo all'immagine target per eseguire l'operazione di convoluzione avviene in questo modo:

- ▶ Si considera un punto di origine nell'immagine target. *In genere il primo pixel in alto a sinistra.*
- ▶ Al pixel si aggancia il centro del kernel, l'ancora.

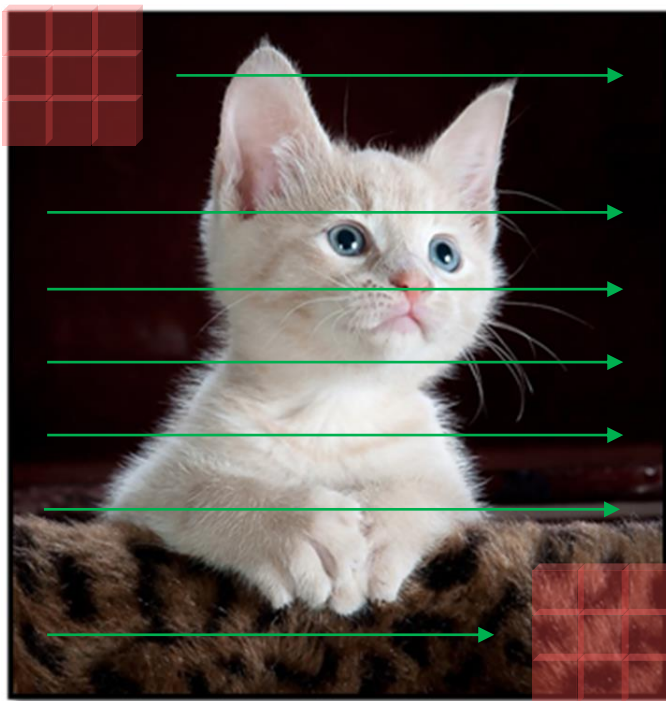


- ▶ Si moltiplica ogni valore del kernel con il valore del rispettivo pixel sottostante e si sommano tutti i valori ottenendone quindi uno solo.
- ▶ Il valore sarà sostituito a quello agganciato dall'ancora.

Convoluzione

Funzionamento

I passi indicati si ripetono per l'intera totalità di pixel dell'immagine aggiornando di volta in volta il valore del pixel al quale ci si aggancia. Lo spostamento in genere avviene da sinistra a destra, completando le riga, e scendendo dall'alto verso il basso.



L'immagine risultante avrà la stessa dimensione di quella sorgente ma sarà visivamente differente in base alla tipologia di kernel applicato.

(Rif: [Wikipedia](#), [Wikimedia](#))

Convoluzione

Kernel e casi d'uso

Esistono moltissime tipologie di kernel, distinguibili per la dimensione, per la forma e per gli specifici valori in essi contenuti.

È possibile però raggrupparli sulla base dell'effetto finale che si vuole ottenere nell'immagine o, più semplicemente, in base al loro scopo:

- Mediare, sfocare, appiattare la sorgente.
- Aumentare la nitidezza, accentuare le variazioni.
- Rilevare bordi.
- Rilevare gradienti.

Convoluzione

Esempi: sharpening



Convoluzione

Esempi: smoothing



Convoluzione

Esempi: edge detection



Convoluzione

Esempi: vertical edge detection



Convoluzione

Esempi: horizontal edge detection

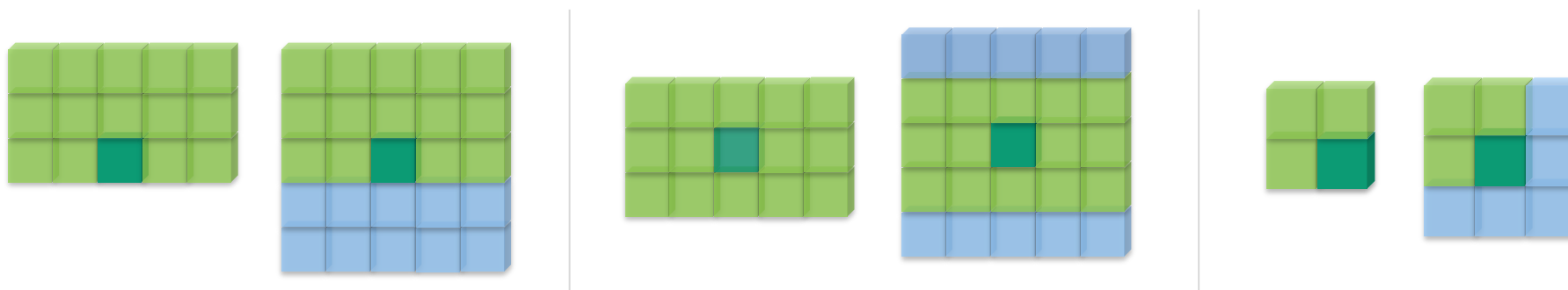


Convoluzione

Kernel non quadrati e ancore

Come detto, un kernel non deve obbligatoriamente essere quadrato e con dimensioni dispari. Allo stesso modo il centro può trovarsi in posto specifico, personalizzato.

Basti pensare che non si tratta altro che di un kernel quadrato e di dimensioni dispari del quale si escludono alcune componenti.



Ne vediamo alcuni esempi dove si possono notare il kernel (*verde*), l'ancora (*verde scuro*) e le componenti 'fantasma' (*azzurro*).

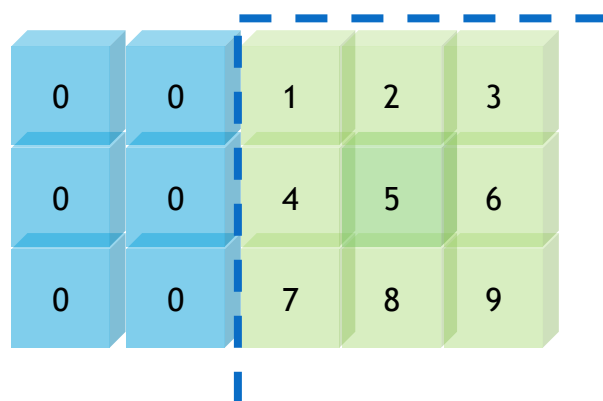
Convoluzione

Kernel al bordo

Muovendosi di pixel in pixel è certo che, ad un certo punto, il kernel si troverà ad avere parte dei valori esterni all'immagine.

L'operazione di convoluzione può essere applicata senza problemi ma, in base alla modalità di risoluzione scelta, gli elementi 'fantasma' che nell'immagine non sono presenti, dovranno assumere un valore.

Di seguito alcuni esempi:



Assegna un valore fisso

In **verde** i pixel al bordo dell'immagine sorgente, in **azzurro** la prosecuzione dei pixel immaginata dal kernel.

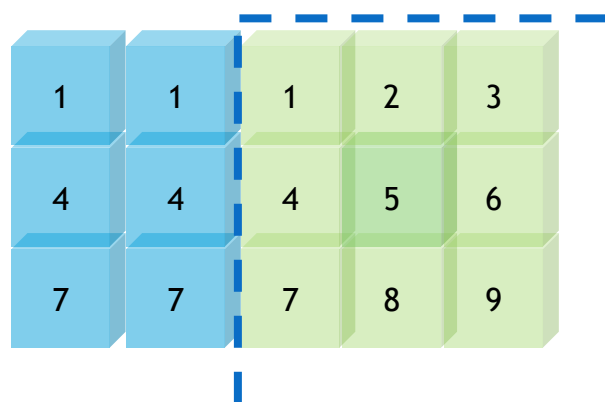
Convoluzione

Kernel al bordo

Muovendosi di pixel in pixel è certo che, ad un certo punto, il kernel si troverà ad avere parte dei valori esterni all'immagine.

L'operazione di convoluzione può essere applicata senza problemi ma, in base alla modalità di risoluzione scelta, gli elementi 'fantasma' che nell'immagine non sono presenti, dovranno assumere un valore.

Di seguito alcuni esempi:



In **verde** i pixel al bordo dell'immagine sorgente, in **azzurro** la prosecuzione dei pixel immaginata dal kernel.

Prosegue con il valore di bordo

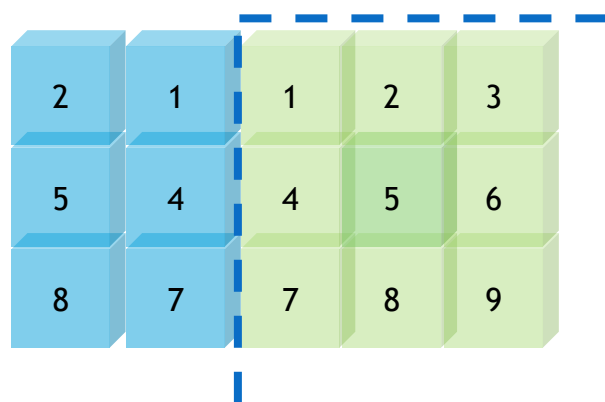
Convoluzione

Kernel al bordo

Muovendosi di pixel in pixel è certo che, ad un certo punto, il kernel si troverà ad avere parte dei valori esterni all'immagine.

L'operazione di convoluzione può essere applicata senza problemi ma, in base alla modalità di risoluzione scelta, gli elementi 'fantasma' che nell'immagine non sono presenti, dovranno assumere un valore.

Di seguito alcuni esempi:



In **verde** i pixel al bordo dell'immagine sorgente, in **azzurro** la prosecuzione dei pixel immaginata dal kernel.

Specchia l'immagine ed il bordo

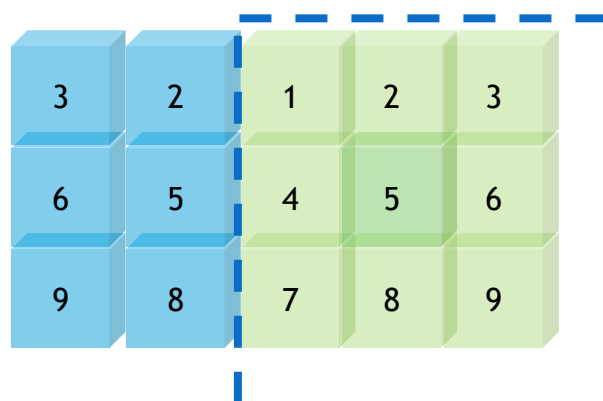
Convoluzione

Kernel al bordo

Muovendosi di pixel in pixel è certo che, ad un certo punto, il kernel si troverà ad avere parte dei valori esterni all'immagine.

L'operazione di convoluzione può essere applicata senza problemi ma, in base alla modalità di risoluzione scelta, gli elementi 'fantasma' che nell'immagine non sono presenti, dovranno assumere un valore.

Di seguito alcuni esempi:



Specchia l'immagine

In **verde** i pixel al bordo dell'immagine sorgente, in **azzurro** la prosecuzione dei pixel immaginata dal kernel.

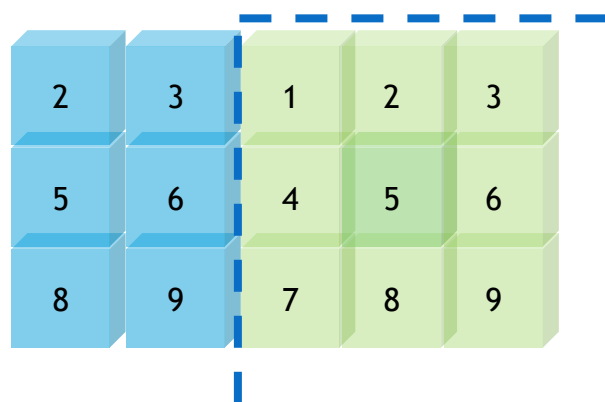
Convoluzione

Kernel al bordo

Muovendosi di pixel in pixel è certo che, ad un certo punto, il kernel si troverà ad avere parte dei valori esterni all'immagine.

L'operazione di convoluzione può essere applicata senza problemi ma, in base alla modalità di risoluzione scelta, gli elementi 'fantasma' che nell'immagine non sono presenti, dovranno assumere un valore.

Di seguito alcuni esempi:



Affianca l'immagine stessa

In **verde** i pixel al bordo dell'immagine sorgente, in **azzurro** la prosecuzione dei pixel immaginata dal kernel.

Proviamo?