Filtri digitali

Francesco Isgrò

3 ottobre 2023

Rumore dell'immagine

Il rumore è generalmente raggruppato in due classi

di rumore indipendenti

rumore dipendente dai dati (dovuto all'interferenza delle onde)

Rumore dell'immagine

Il rumore è generalmente raggruppato in due classi di

- rumore additivo
  - indipendenti

$$I_{ij} = \hat{I}_{ij} + n_{ij}$$

  - è spesso a media zero, cioè $\mu_n = 0$
  - descritto dalla sua varianza ÿ $^{2}_{N}$
- rumore dipendente dai dati (dovuto all'interferenza delle onde)

Rumore dell'immagine

Il rumore è generalmente raggruppato in due classi di

- rumore additivo
  - indipendenti

$$I_{ij} = \hat{I}_{ij} + n_{ij}$$

  - è spesso a media zero, cioè $\mu_n = 0$
  - descritto dalla sua varianza ÿ ^2_N
- rumore dipendente dai dati (dovuto all'interferenza delle
  - onde) modello moltiplicativo o non lineare

$$I_{ij} = \hat{I}_{ij} n_{ij}$$

  - più complicato

Il rumore è generalmente raggruppato in due classi di

rumore additivo

indipendenti

$$Iij = \hat{}Iij + nij$$

è spesso a media zero, cioè µn = 0

descritto dalla sua varianza ÿ $^{2}_{N}$

rumore dipendente dai dati (dovuto all'interferenza delle

onde) modello moltiplicativo o non lineare

$$Iij = \hat{}Iijnij$$

più complicato a meno

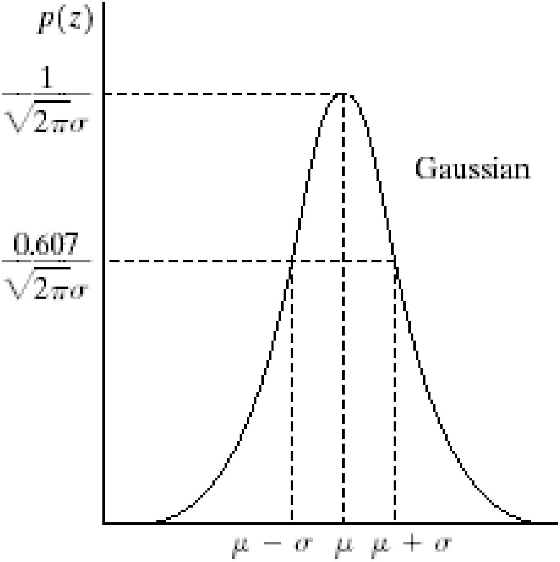
che il rumore necessario non venga assunto indipendente dai dati

Principali modelli di rumore

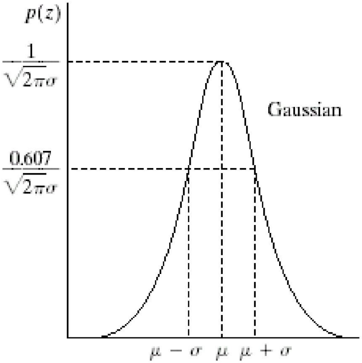
Il rumore è modellato come una variabile casuale

- Rumore gaussiano
- Rumore di Rayleigh
- Rumore gamma
- Rumore esponenziale
- Rumore uniforme
- Rumore impulsivo

Rumore gaussiano

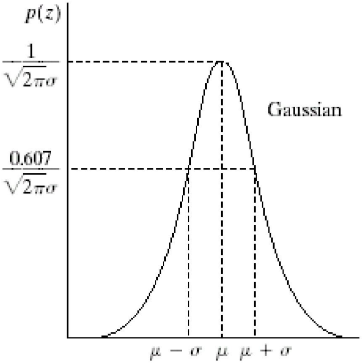


Rumore gaussiano



1 p(z) = ÿ 2ÿÿ e ÿ(zÿÿ) 2/2ÿ 2

Rumore gaussiano



1 p(z) = $\frac{\quad}{\text{ÿ 2ÿÿ}}$ e$^{\text{ÿ(zÿÿ) 2/2ÿ 2}}$

z è il livello di grigio µ

è la media ÿ è la

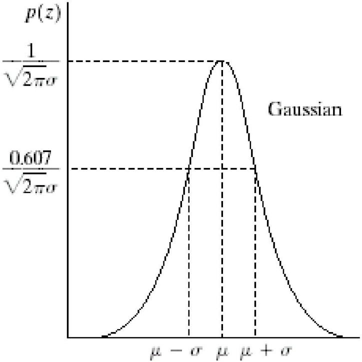
deviazione standard

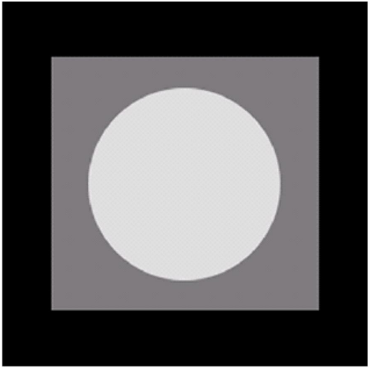
è dovuto al rumore del sensore,

alla cattiva illuminazione, alla

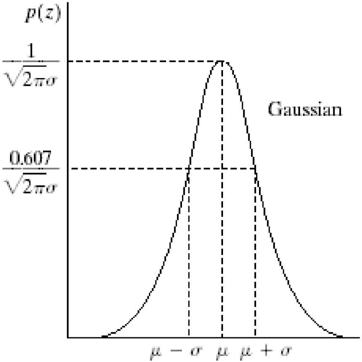
temperatura elevata

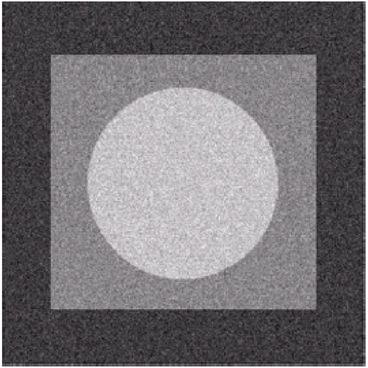
Rumore gaussiano



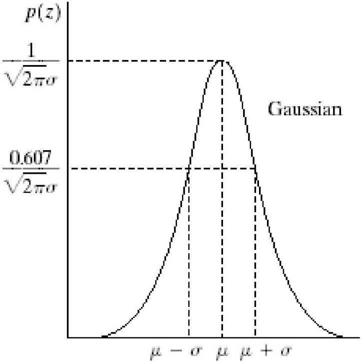


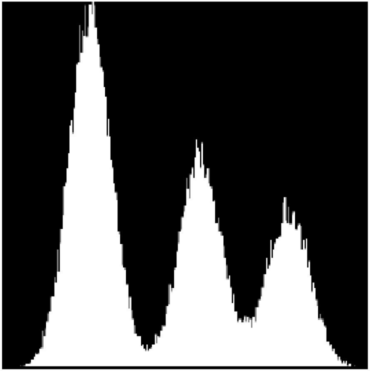
Rumore gaussiano



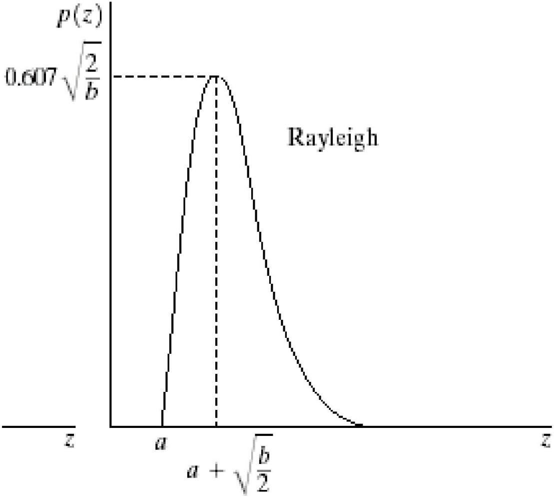


Rumore gaussiano

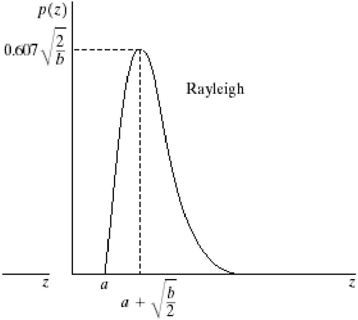




Rumore di Rayleigh

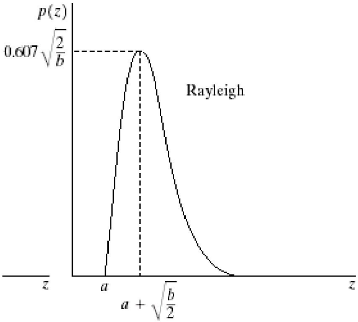


Rumore di Rayleigh



p(z) = – (z ÿ a)e ÿ(zÿa) 2/b se z ÿ a
2b0 se z < a

Rumore di Rayleigh



p(z) = ⎯ (z ÿ a)e ÿ(zÿa) 2/b se z ÿ a
2b0 se z < a

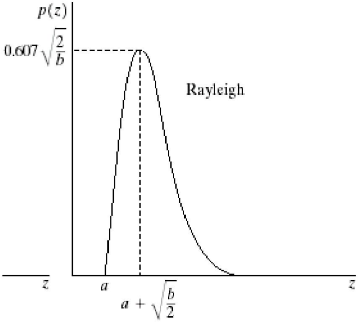
µ = a + ÿb/4 b(4ÿÿ) 4

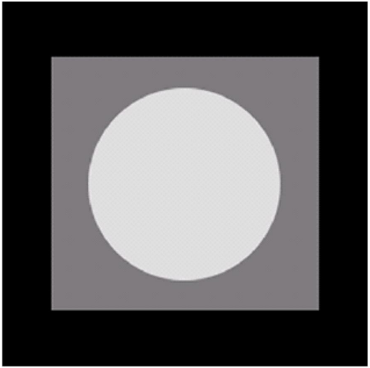
2ÿ =

usato per caratterizzare

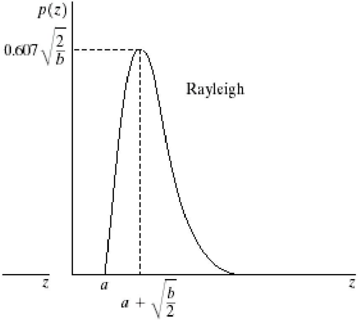
rumore nelle immagini a distanza

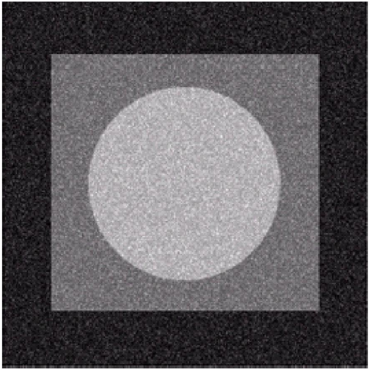
Rumore di Rayleigh



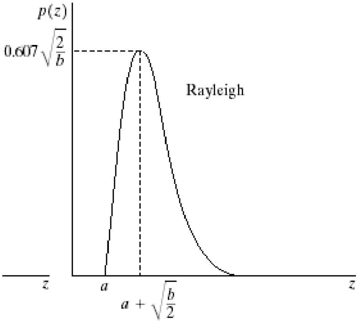


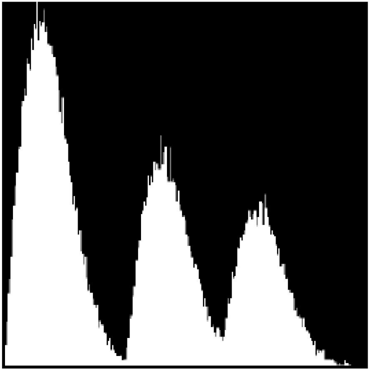
Rumore di Rayleigh



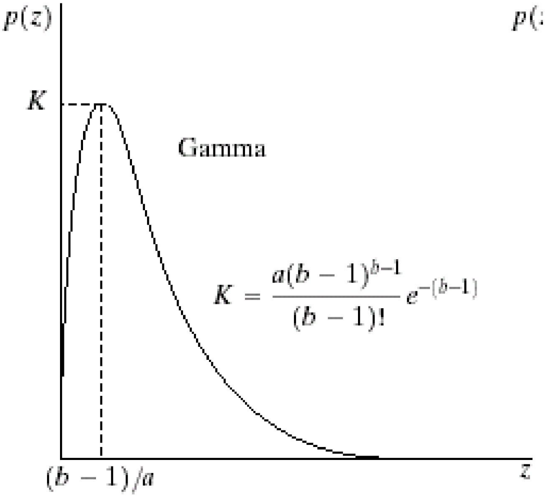


Rumore di Rayleigh

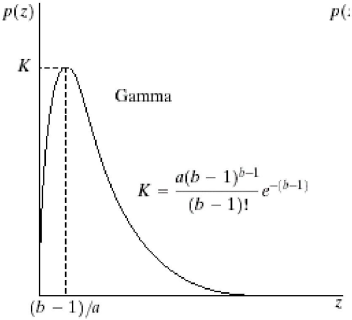




Rumore gamma



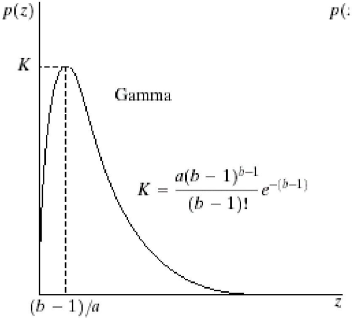
Rumore gamma



p(z) = $\frac{{}_{\text{UN}}B z^{\text{bÿ1}}}{\text{(bÿ1)! 0}}$ e$_{\text{ÿaz se z ÿ 0}}$

se z < 0

Rumore gamma



p(z) = UNBz bÿ1 / (bÿ1)! e ÿaz se z ÿ 0
0 se z < 0

a > 0

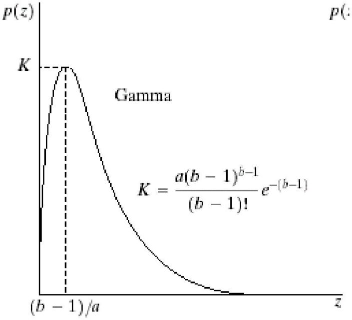
bZ

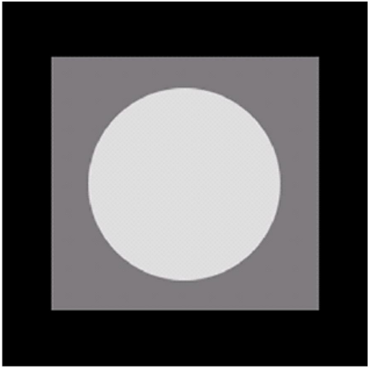
µ = b/a

2ÿ = b/a 2

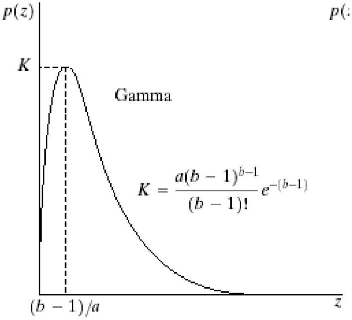
utilizzato nell'imaging laser

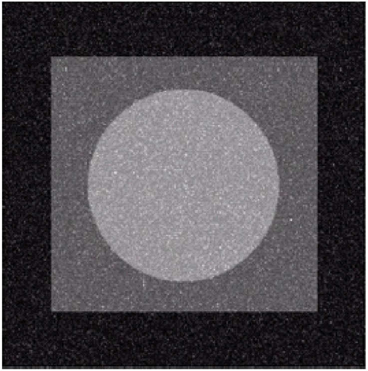
Rumore gamma



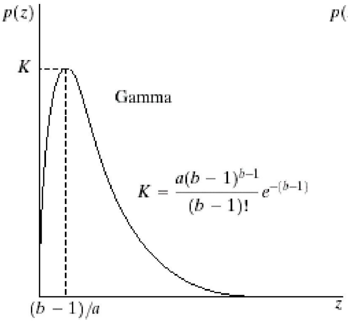


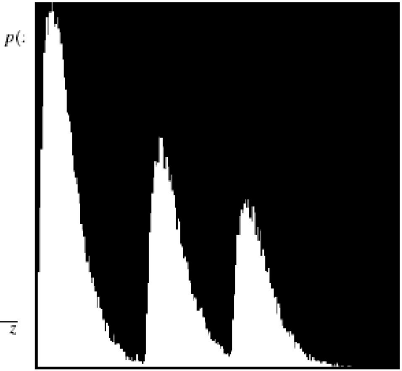
Rumore gamma



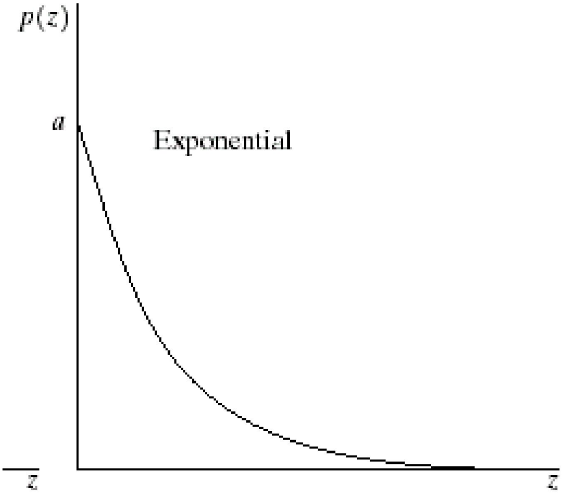


Rumore gamma

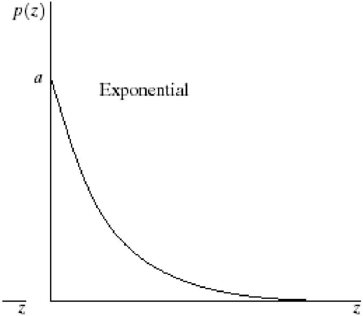




Rumore esponenziale



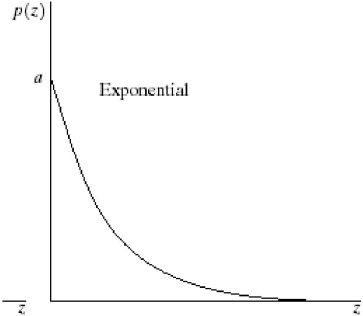
Rumore esponenziale



p(z) = aeÿaz se z ÿ 0

0 se z < 0

Rumore esponenziale



p(z) = aeÿaz se z ÿ 0
0 se z < 0

a > 0

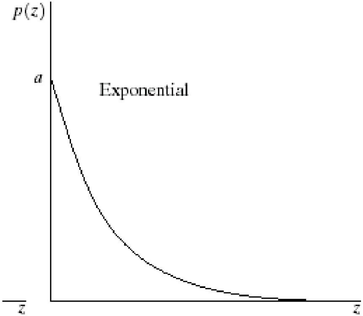
µ = 1/a

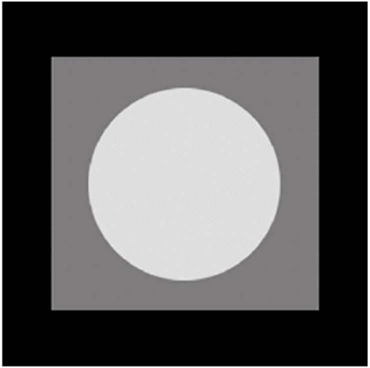
2ÿ = 1/a 2

è un caso speciale di
Gamma pdf, quando b = 1

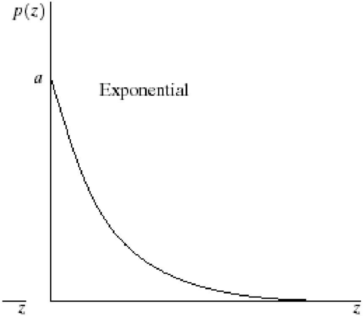
utilizzato nell'imaging laser

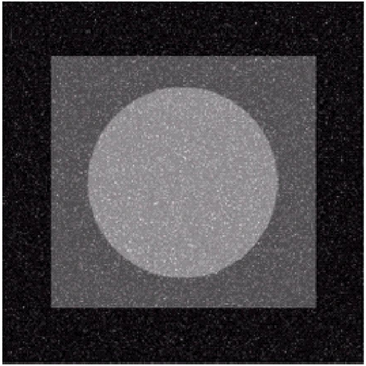
Rumore esponenziale



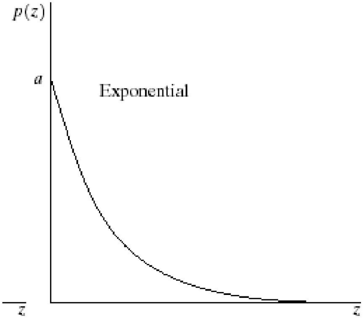


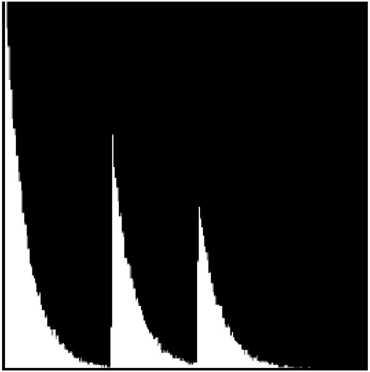
Rumore esponenziale



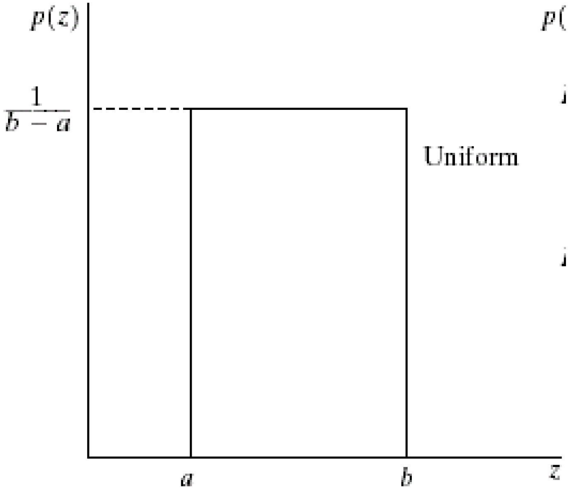


Rumore esponenziale

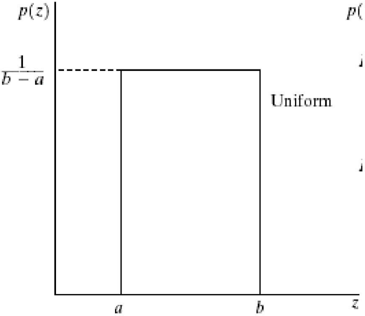




Rumore uniforme

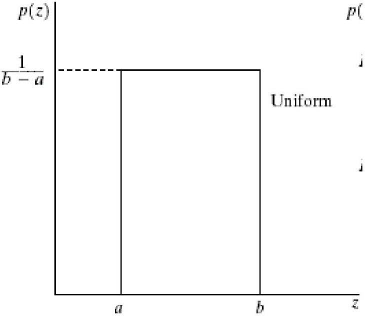


Rumore uniforme



$$p(z) = \begin{cases} \frac{1}{b-a} & \text{se a ÿ z ÿ b} \\ 0 & \text{Altrimenti} \end{cases}$$

Rumore uniforme

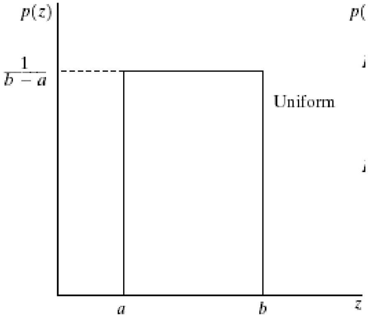


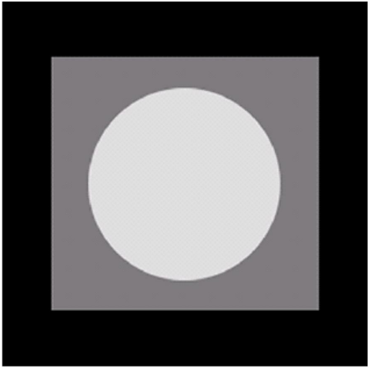
$$p(z) = \begin{cases} \frac{1}{b-a} & \text{se a ÿ z ÿ b} \\ 0 & \text{altrimenti} \end{cases}$$

$$\mu = \frac{a+b}{2}$$

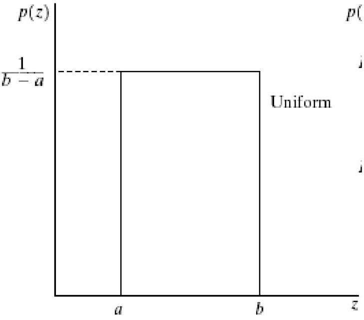
$$2ÿ = \frac{(b-a)^2}{12}$$

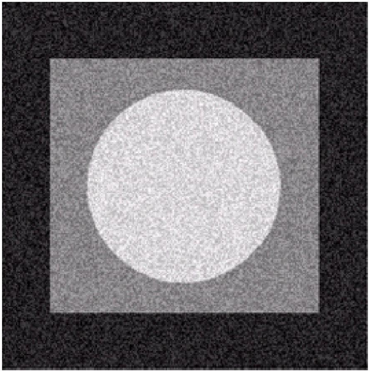
Rumore uniforme



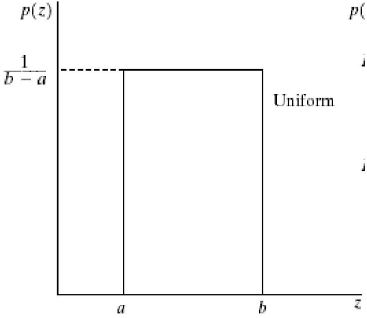


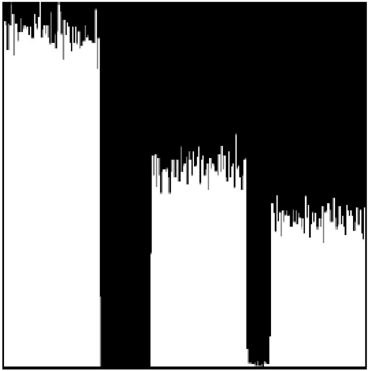
Rumore uniforme



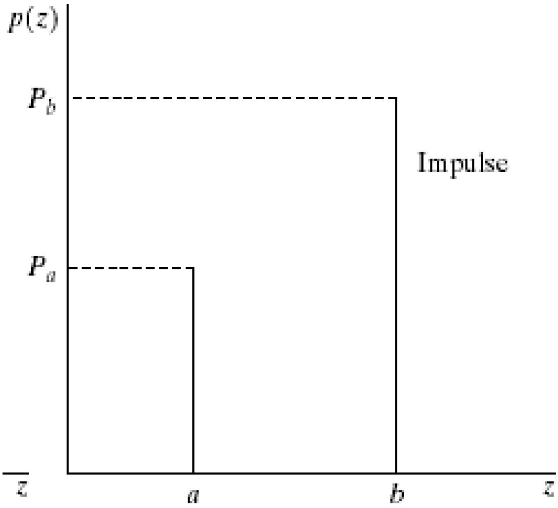


Rumore uniforme

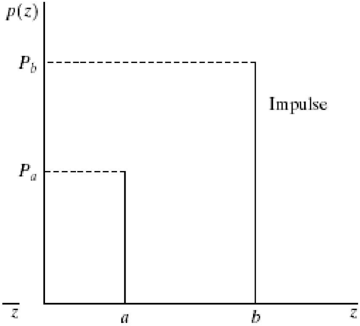




Rumore impulsivo



Rumore impulsivo



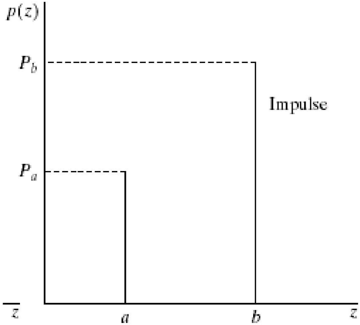
ÿ Pa se z = a

p(z) = Pb se z = b 0

ÿ altrimenti

ÿ

Rumore impulsivo



p(z) = ÿ Pa se z = a
ÿ Pb se z = b 0
ÿ altrimenti

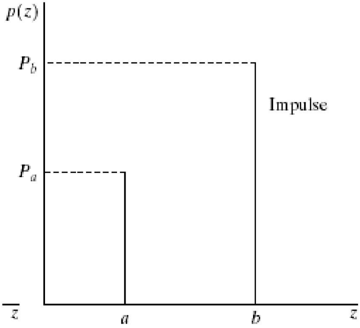
se Pa = 0 o Pb = 0
detto rumore unipolare

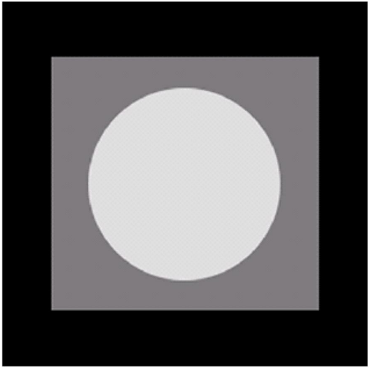
se Pa ÿ Pb detto rumore
sale e pepe

modella il rumore

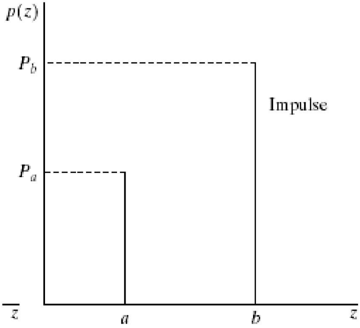
causati da errori nella
trasmissione dei dati

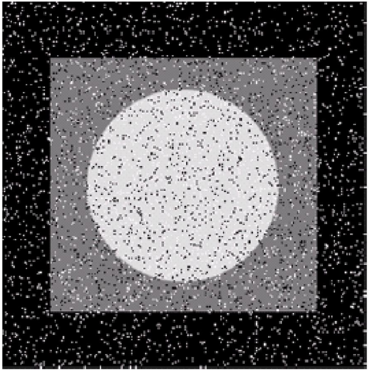
Rumore impulsivo



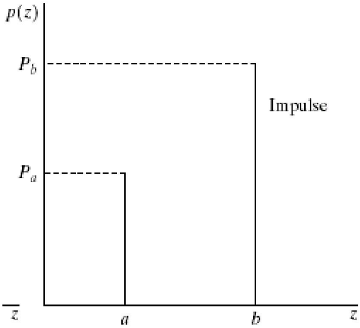


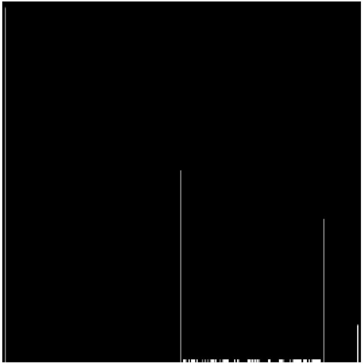
Rumore impulsivo





Rumore impulsivo





Come indovinare il rumore?

Come indovinare il rumore?

Scatta alcune immagini piatte e studia l'istogramma

Come indovinare il rumore?

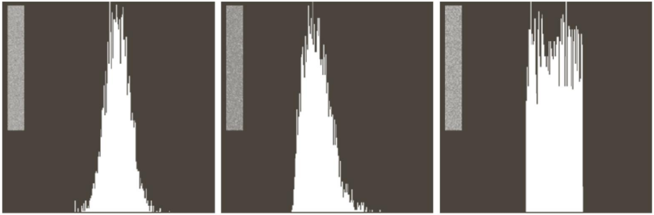
Scatta alcune immagini piatte e studia l'istogramma.

Cosa succede se ottieni solo le immagini?

Come indovinare il rumore?

Scatta alcune immagini piatte e studia l'istogramma.

Cosa succede se ottieni solo le immagini?



Filtri digitali

I filtri vengono utilizzati principalmente per:

- smussare l'immagine
- migliorando o rilevando i bordi nell'immagine

Filtri digitali

I filtri vengono utilizzati principalmente per:

smussare l'immagine migliorando

o rilevando i bordi nell'immagine

$$I_{ij} = SÌ(INij , hN)$$

hN kernel di dimensione pari alla dimensione dell'intorno N

ON un operatore che può essere

- lineare
- non lineare

Ho chiamato la risposta del filtro ij

Filtri digitali

I11	I12	I13	I14	I15	I16	I17	I18	I19	
I21	I22	I23	I24	I25	I26	I27	I28	I29	
I31	I32	I33	I34	I35	I36	I37	I38	I39	
I41	I42	I43	I44	I45	I46	I47	I48	I49	
I51	I52	I53	I54	I55	I56	I57	I58	I59	
I61	I62	I63	I64	I65	I66	I67	I68	I69	

h11	h12	h13
h21	h22	h23
h31	h32	h33

Filtra il kernel

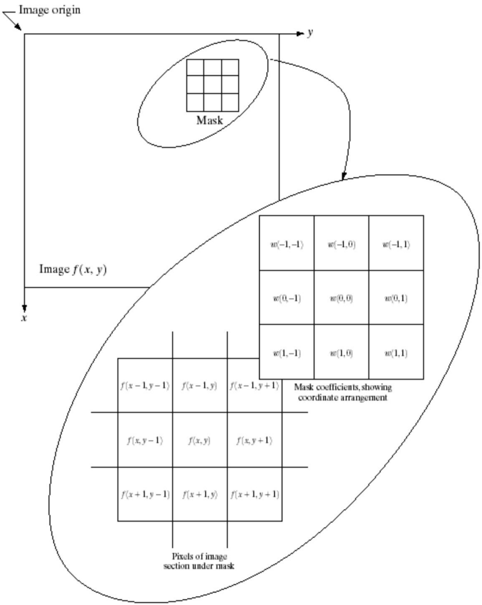
dimensione m×n

$m = 2a + 1$

$n = 2b + 1$

a volte il pixel centrale è (0, 0)

Filtri digitali



Filtri lineari

Definizione

Un filtro è lineare se

$$ON(aI1 + bI2, hN) = aON(I1, hN) + bON(I2, hN)$$

Filtri lineari

Definizione

Un filtro è lineare se

$$ON(aI1 + bI2, hN) = aON(I1, hN) + bON(I2, hN)$$

i filtri lineari sono solitamente chiamati
kernel di convoluzione

Filtri lineari

Definizione

Un filtro è lineare se

$$ON(aI1 + bI2, hN) = aON(I1, hN) + bON(I2, hN)$$

i filtri lineari sono solitamente chiamati kernel di convoluzione

Implementazione

UN B

Io ij = s=ÿa t=ÿb Ii+s,j+thsÿa,tÿb

Filtri leviganti

Cosa sono

I filtri di attenuazione riducono i cambiamenti bruschi nell'immagine. Effetti principali

Sono:

- sfocare l'immagine
- rimuovere piccoli dettagli
- colmare piccole lacune
- ridurre il rumore
- ammorbidire i bordi

Filtri di livellamento lineare

filtro medio

media ponderata

Filtro gaussiano

Filtro medio

Definizione

$$\text{Io ij} = \frac{1}{mn} \quad \overset{\text{UN}}{} \quad \overset{\text{B}}{\underset{\text{s=ÿbr=ÿa}}{}} \quad \text{Ii+r,j+s}$$

Esempio di kernel

$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$
$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$
$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$

Filtro medio

Definizione

$$I^{o}_{ij} = \frac{1}{mn} \sum^{UN} \sum^{B}_{s=ÿb\, r=ÿa} I_{i+r,j+s}$$

Esempio di kernel

$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$
$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$
$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$

Filtro medio

Filtro medio

3×3

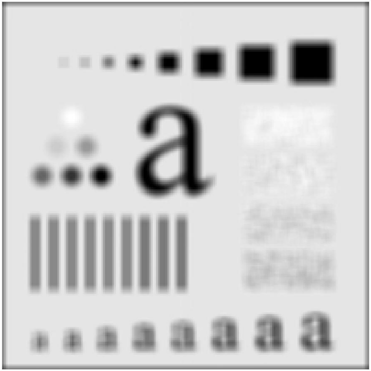
Filtro medio

5×5

Filtro medio

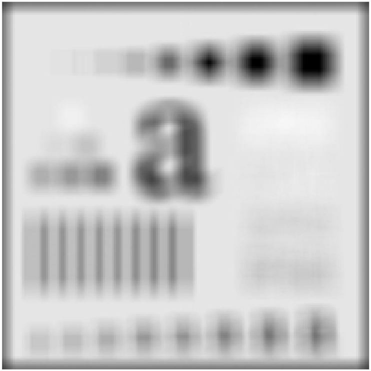
9×9

Filtro medio



15×15

Filtro medio



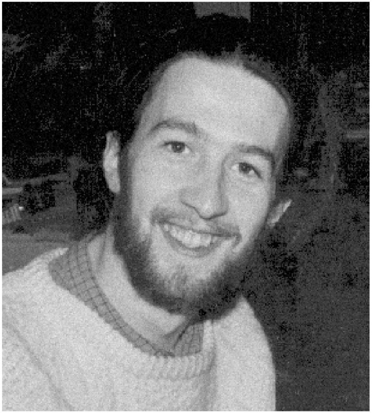
35×35

Filtro medio

Immagine originale

Filtro medio

Immagine originale



Rumore gaussiano $\mu = 0$ e ÿ = 8

Filtro medio

Immagine originale

3×3

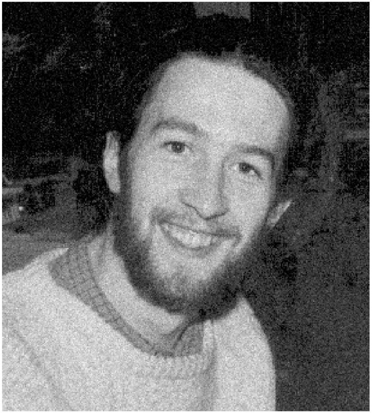
Filtro medio

Immagine originale

5×5

Filtro medio

Immagine originale



Rumore gaussiano µ = 0 e
ÿ = 13

Filtro medio

Immagine originale

3×3

Filtro medio

Immagine originale

rumore di sale e pepe

Filtro medio

Immagine originale

3×3

Filtro medio

Immagine originale

5×5

Filtro medio

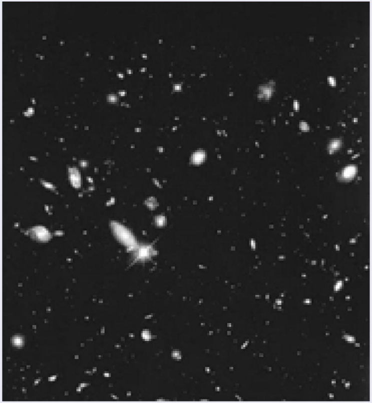
Principali problemi

Un singolo pixel con un valore molto non rappresentativo può influenzare in modo significativo il valore medio di tutti i pixel nelle sue

vicinanze. Quando l'intorno del filtro si trova a cavallo di un bordo, il filtro interpolerà nuovi valori per i pixel sul bordo e quindi sfocherà quel bordo. Questo potrebbe rappresentare un problema se nell'output sono necessari bordi taglienti.

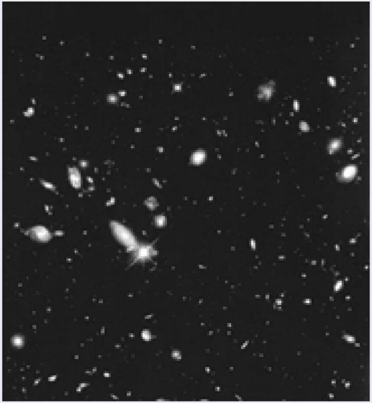
Filtro medio

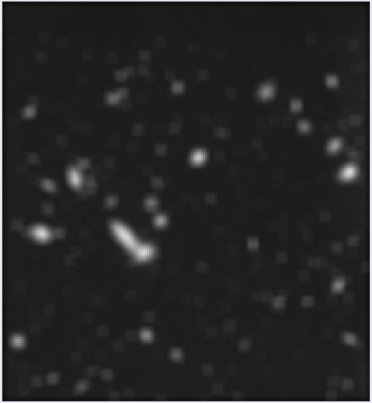
Filtra gli oggetti di interesse



Filtro medio

Filtra gli oggetti di interesse

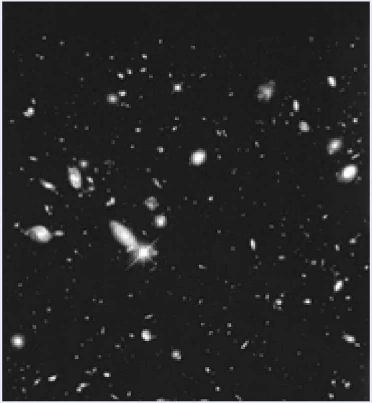




15×15

Filtro medio

Filtra gli oggetti di interesse



Sogliato

Filtro medio

E il calcolo?

$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$
$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$
$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$

Filtro medio

E il calcolo?

$$\begin{array}{|c|c|c|}\hline \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \hline \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \hline \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \hline \end{array}$$

Molto meglio

$$\frac{1}{9}\begin{array}{|c|c|c|}\hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline 1 & 1 & 1 \\ \hline \end{array}$$

Media ponderata

$$\text{Io}_{ij} = \sum_{s=\text{ÿa}}^{\text{UN}} \sum_{t=\text{ÿb}}^{B} \text{Ii+s,j+thsÿa,tÿb}$$

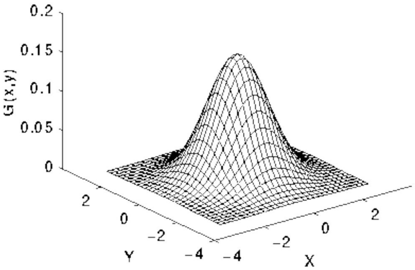
Molto meglio avere hstZ e calcolare

$$\text{Io}_{ij} = \frac{\sum_{s=\text{ÿa}}^{\text{UN}} \sum_{t=\text{ÿb}}^{B} \text{Ii+s,j+thsÿa,tÿb}}{\sum_{s=\text{ÿa}}^{\text{UN}} \sum_{t=\text{ÿb}}^{B} \text{hsÿa,tÿb}}$$

Filtro gaussiano

In questo caso i pesi vengono campionati da una superficie gaussiana.
Nel nostro caso bidimensionale

$$hst = e^{\overline{s2+t2}2ÿ2}$$



Filtro gaussiano

Come progettare un kernel gaussiano

vogliamo che la maggior parte dell'area sotto la gaussiana sia coperta

abbiamo bisogno di una relazione tra la dimensione della maschera e ÿ

prendiamo 2a + 1 = 5ÿ in modo che il 98,76% dell'area sia coperta 5ÿ ÿ 1

$$\text{un} = \frac{\quad}{2}$$

Filtro gaussiano

Come progettare un kernel gaussiano

vogliamo che la maggior parte dell'area sotto la gaussiana sia coperta

abbiamo bisogno di una relazione tra la dimensione della maschera e ÿ

prendiamo 2a + 1 = 5ÿ in modo che il 98,76% dell'area sia coperta 5ÿ ÿ 1

$$\text{un} = \frac{\qquad}{2}$$

Algoritmo

1. Calcola il kernel in virgola mobile hst 2. trova hmin =

minst hst

3. hst = giro(h ÿ1 minimo)

Filtro gaussiano

I kernel gaussiani sono separabili

UN B s 2+t 2 UN 2 seconds B 2 t

e 2ÿ2 Ii+t,j+s = e 2ÿ2 e 2ÿ2Ii +t,j+s

s=ÿa t=ÿb s=ÿa t=ÿb

Filtro gaussiano

I kernel gaussiani sono separabili

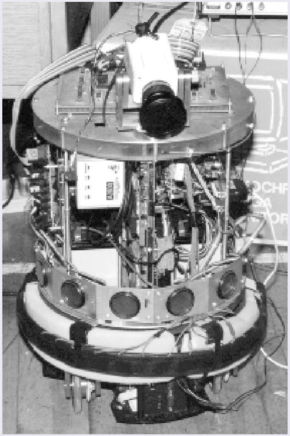
$$\sum_{s=-a}^{a}\sum_{t=-b}^{b} e^{-\frac{s^2+t^2}{2\sigma^2}} I_{i+t,j+s} = \sum_{s=-a}^{a} e^{-\frac{s^2}{2\sigma^2}} \sum_{t=-b}^{b} e^{-\frac{t^2}{2\sigma^2}} I_{i+t,j+s}$$

Implementazione

- 2 convoluzioni a cascata con maschere monodimensionali
- prima convolvono tutte le colonne
- quindi convolgi tutte le righe
- la complessità temporale è $O(n)$ e non $O(n^2)$ (n è la dimensione della maschera)

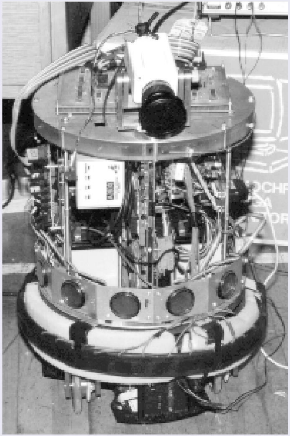
Filtro gaussiano

Effetti della levigatura



Filtro gaussiano

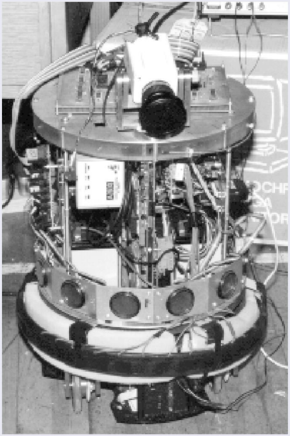
Effetti della levigatura



ÿ = 1 5 × 5

Filtro gaussiano

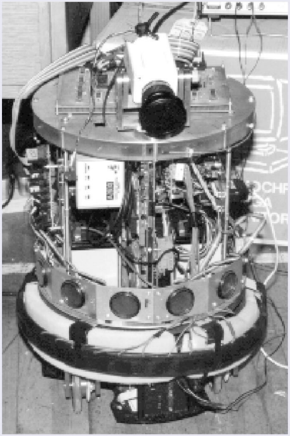
Effetti della levigatura



ÿ = 2 9 × 9

Filtro gaussiano

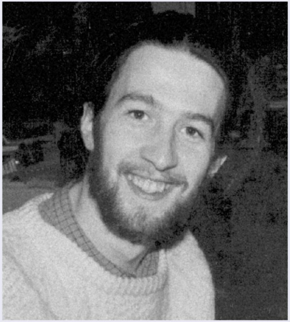
Effetti della levigatura



ÿ = 4 15 × 15

Filtro gaussiano

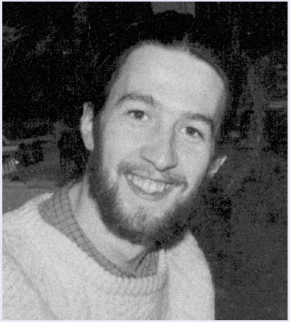
Riduzione del rumore



Rumore gaussiano µ = 0 e ÿ = 8

Filtro gaussiano

Riduzione del rumore

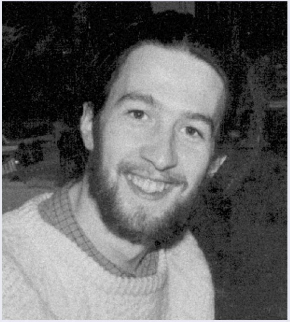


Rumore gaussiano $\mu = 0$ e ÿ = 8

5×5

Filtro gaussiano

Riduzione del rumore



Rumore gaussiano $\mu = 0$ e ÿ = 8

significa 5×5

Filtri statistici sugli ordini

Filtri spaziali non lineari la cui risposta si basa sulla classificazione dei pixel nell'area coperta dal filtro.

Filtri statistici sugli ordini

Filtri spaziali non lineari la cui risposta si basa sulla classificazione dei pixel nell'area coperta dal filtro.

Esempi

- filtro mediano
- livellamento conservativo

Filtro mediano

$$io_{ij} = \underset{(s,t)Nij}{med} \quad Ist$$

utilizzato per ridurre il rumore

preserva i dettagli meglio dei filtri medi

Filtro mediano

$$i^{o}_{ij} = \operatorname*{med}_{(s,t)N_{ij}} I_{st}$$

123	125	126	130	140
122	124	126	127	135
118	120	150	125	134
119	115	119	123	133
111	116	110	120	130

Neighbourhood values:

115, 119, 120, 123, 124,
125, 126, 127, 150

Median value: 124

Filtro mediano

$$io_{ij} = \underset{(s,t)Nij}{\text{med}} \quad Ist$$

Domanda

Come implementiamo il filtro mediano?

Filtro mediano

$$io_{ij} = \text{med}_{(s,t)Nij} \quad Ist$$

Domanda

Come implementiamo il filtro mediano?

ordinamento

Filtro mediano

$$io_{ij} = \underset{(s,t)Nij}{\mathrm{med}} \quad Ist$$

Domanda

Come implementiamo il filtro mediano?

l'ordinamento seleziona Kth più grande

Filtro mediano

```
funzione select(list[1..n], k) per i da 1 a k

  Indicemin = i
  minValue = lista[i] per j da i+1
   a n
       if lista[j] < minValue minIndex = j
            minValue = list[j]
            scambia lista[i] e list[minIndex]

lista dei resi[k]
```

Filtro mediano

```
funzione select(lista, sinistra, destra, k) seleziona
pivotIndex tra sinistra e destra pivotNewIndex =
partizione(lista, sinistra, destra, pivotIndex) if k = pivotNewIndex
return lista[k]

altrimenti se k < pivotNuovoIndice return
  select(lista, sinistra, pivotNuovoIndice-1, k) altrimenti

  return select(lista, pivotNuovoIndice+1, destra, k)
```

Filtro mediano

Domanda

Si può fare in tempo lineare?

Filtro mediano

Filtro mediano veloce di Huang

funzione filtro mediano(I, r)

Inizializza l'istogramma del kernel H per i = 1 a m

per j=1 in n per k=-r

in r

rimuovi Ii+k,jÿrÿ1 da H aggiungi Ii+k,j+r

a H

FINE

= mediana(H)

Finirò

FINE

Filtro mediano

la mediana è una media più solida della media un singolo pixel

molto non rappresentativo in un quartiere non influenzerà il valore mediano

in modo significativo il valore mediano è uno dei

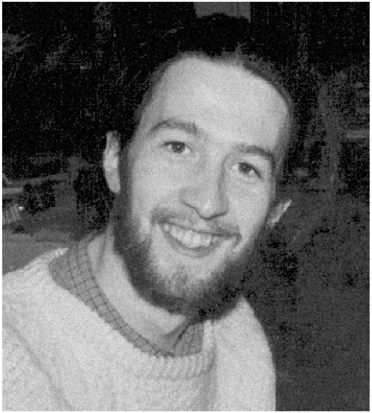
pixel nel quartiere il filtro mediano non crea nuovi valori di pixel non

realistici quando il filtro si trova a cavallo un bordo è molto più efficace nel

preservare i bordi taglienti rispetto al

filtro medio

Filtro mediano



Rumore gaussiano $\mu = 0$ ÿ $= 8$

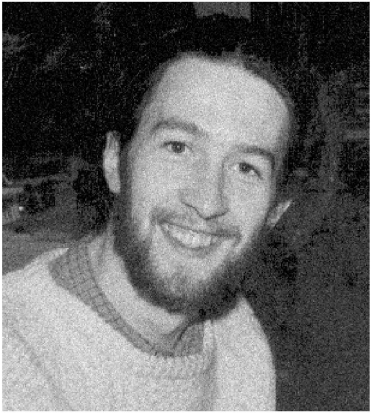
Filtro mediano

Filtro medio 3×3

Filtro mediano

Filtro mediano 3×3

Filtro mediano



Rumore gaussiano $\mu = 0$ ÿ = 13

Filtro mediano

Filtro mediano 3×3

Filtro mediano

Filtro medio 3×3

Filtro mediano

Rumore di sale e pepe

Filtro mediano

Filtro mediano 3×3

Filtro mediano

Filtro medio 3×3

Filtro mediano

Rumore di sale e pepe

Filtro mediano

Filtro mediano 3×3

Filtro mediano

Filtro mediano 7×7

Filtro mediano

Filtro mediano 3 3×3

Filtro mediano

Filtro mediano 7×7

Filtro mediano 3 3×3

Levigatura conservativa

Lo smussamento conservativo utilizza un algoritmo di filtraggio semplice e veloce che sacrifica il potere di soppressione del rumore per preservare i dettagli (ad esempio i bordi netti) in

un'immagine. più efficace con il rumore sale e pepe

meno efficace con il rumore additivo

Levigatura conservativa

I filtri tentano di rendere l'intensità del pixel coerente con quella dei suoi vicini più vicini. filtro medio: intensità

locali medie filtro mediano: selezione del rango

livellamento conservativo: garantisce che l'intensità di ciascun pixel sia limitata all'interno dell'intervallo di intensità definito dai suoi vicini

Levigatura conservativa

I filtri tentano di rendere l'intensità del pixel coerente con quella dei suoi vicini più vicini. filtro medio: intensità

locali medie filtro mediano: selezione del rango

livellamento conservativo: garantisce che l'intensità di ciascun pixel sia limitata all'interno dell'intervallo di intensità definito dai suoi vicini

Definizione

ÿ Iij se min(Nij) ÿ Iij ÿ max(Nij)

Io ij = ÿ min(Nij) se min(Nij) > Iij max(Nij) se

ÿ max(Nij) < Iij

Levigatura conservativa

Definizione

ÿ Iij se min(Nij) ÿ Iij ÿ max(Nij)

lo ij = ÿ min(Nij) se min(Nij) > Iij max(Nij) se

ÿ max(Nij) < Iij

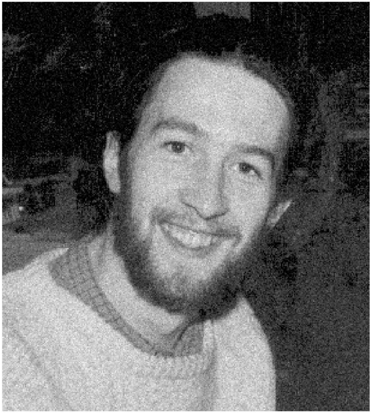
123	125	126	130	140
122	124	126	127	135
118	120	150	125	134
119	115	119	123	133
111	116	110	120	130

Neighborhood values:

115, 119, 120, 123, 124,
125, 126, 127, 150

Max: 127, Min: 115

Levigatura conservativa



Rumore gaussiano $\mu = 0$ ÿ $= 13$

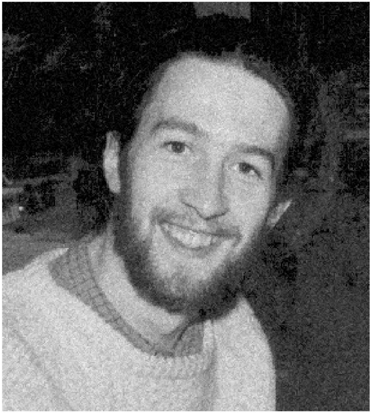
Levigatura conservativa

Filtro medio 3×3

Levigatura conservativa

Filtro mediano 3×3

Levigatura conservativa



Levigatura conservativa 3×3

Levigatura conservativa

il filtro medio gestisce meglio il rumore gaussiano: inserisce valori

di intensità intermedi il filtro mediano

non è altrettanto buono: utilizza i valori di intensità originali il filtro

mediano preserva i dettagli: utilizza i valori di intensità originali il livellamento

conservativo non è in grado di ridurre molto il rumore gaussiano poiché i

singoli valori dei pixel rumorosi non lo fanno variano molto dai loro vicini

Levigatura conservativa

Rumore di sale e pepe

Levigatura conservativa

Filtro medio 3×3

Levigatura conservativa

Filtro mediano 3×3

Levigatura conservativa

Levigatura conservativa 3×3

Levigatura conservativa

Rumore di sale e pepe

Levigatura conservativa

Levigatura conservativa 3×3

Levigatura conservativa

Filtro mediano 3×3

Filtri adattativi

prestazioni migliori

ridurre il rumore

preservare i dettagli (ad esempio, i bordi)

di maggiore complessità

filtro adattivo di riduzione del rumore locale filtro

adattivo mediano

Riduzione adattiva del rumore locale

Ciò che vogliamo

niente se non c'è rumore,

piccoli cambiamenti di intensità vicino ai bordi

agiscono come filtro medio dove l'immagine è rumorosa

Riduzione adattiva del rumore locale

Ciò che vogliamo

niente se non c'è rumore,

piccoli cambiamenti di intensità vicino ai bordi

agiscono come filtro medio dove l'immagine è rumorosa

Formalmente...

ÿÿ deviazione standard del rumore

ÿNij deviazione standard dell'immagine locale

Riduzione adattiva del rumore locale

Ciò che vogliamo

niente se non c'è rumore,

piccoli cambiamenti di intensità vicino ai bordi

agiscono come filtro medio dove l'immagine è rumorosa

Formalmente...

ÿÿ deviazione standard del rumore

ÿNij deviazione standard dell'immagine

locale se ÿÿ = 0 allora ‡ Iij ij

Riduzione adattiva del rumore locale

Ciò che vogliamo

niente se non c'è rumore,

piccoli cambiamenti di intensità vicino ai bordi

agiscono come filtro medio dove l'immagine è rumorosa

Formalmente...

ÿÿ deviazione standard del

rumore ÿNij deviazione standard

dell'immagine ij = Iij

locale se ÿÿ = 0 allora Iij se ÿÿ ij ÿNij allora I

Riduzione adattiva del rumore locale

Ciò che vogliamo

niente se non c'è rumore,

piccoli cambiamenti di intensità vicino ai bordi

agiscono come filtro medio dove l'immagine è rumorosa

Formalmente...

ÿÿ deviazione standard del

rumore ÿNij deviazione standard

dell'immagine ij = Iij

locale se ÿÿ = 0 allora Iij se ÿÿIijÿNij allora I

se ÿÿ ÿNij allora Iij = media(Nij)

Riduzione adattiva del rumore locale

Ciò che vogliamo

niente se non c'è rumore,

piccoli cambiamenti di intensità vicino ai bordi

agiscono come filtro medio dove l'immagine è rumorosa

$$I^{o}_{ij} = I_{ij} \text{ ÿ } \frac{2ÿÿ2ÿ}{N_{ij}} \; I_{ij} \text{ ÿ } \mu N_{ij}$$

Riduzione adattiva del rumore locale

Ciò che vogliamo

niente se non c'è rumore,

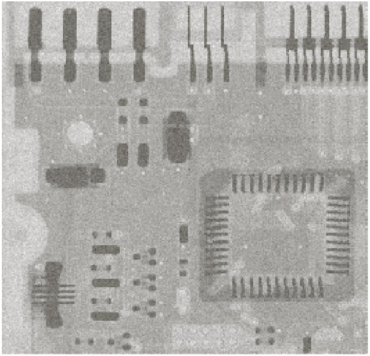
piccoli cambiamenti di intensità vicino ai bordi

agiscono come filtro medio dove l'immagine è rumorosa

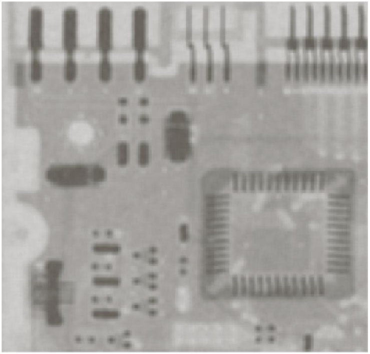
Io ij = Iij ÿ $\frac{}{2ÿÿ2ÿ \; Nij}$ Iij ÿ µNij

Si assume ÿÿ ÿ ÿNij . Altrimenti valori negativi. $\frac{2ÿÿ2ÿ}{Nij}$ è impostato su 1, impedendo

Riduzione adattiva del rumore locale

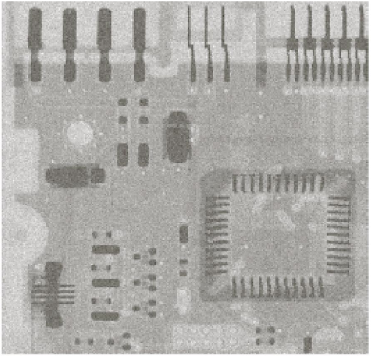


Rumore gaussiano ÿ^2 = 1000

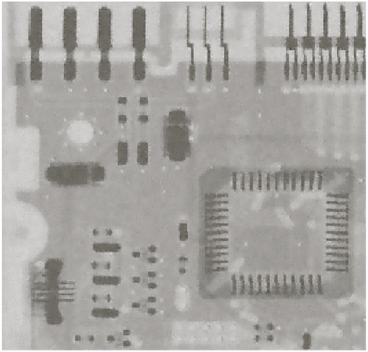


filtro medio 7 × 7

Riduzione adattiva del rumore locale

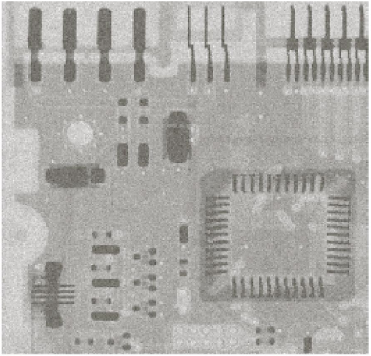


Rumore gaussiano $ÿ^2$ = 1000

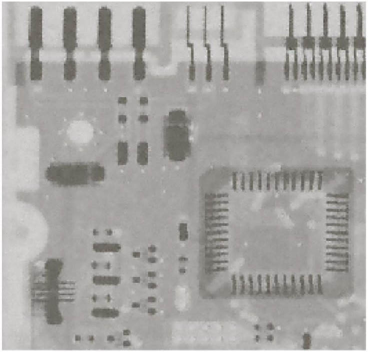


filtro adattivo 7 × 7 ÿ $^{2}_{ÿ}$ = 1000

Riduzione adattiva del rumore locale



Rumore gaussiano ÿ^2 = 1000



filtro adattivo 7 × 7 ÿ $^{2}_{ÿ}$ = 1000

non troppa correzione se ÿÿ troppo piccolo: I se ÿÿ

troppo grande perdita di dettaglio come per il filtro medio

Filtro mediano adattivo

L'obiettivo è

rimuovere il rumore

impulsivo preservare i confini dell'oggetto

Filtro mediano adattivo

L'obiettivo è

rimuovere il rumore

impulsivo preservare i confini dell'oggetto

il filtro mediano funziona bene se il rumore impulsivo non è troppo grande, il filtro mediano adattivo è in grado di gestire una corruzione maggiore e cerca di preservare i dettagli

Filtro mediano adattivo

Come funziona

Filtro mediano adattivo

Come funziona

Il problema con il rumore impulsivo di grandi dimensioni è che la mediana può essere un impulso

Filtro mediano adattivo

Come funziona

Il problema con il rumore impulsivo di grandi dimensioni è che la mediana può essere un

impulso, prima controlla se la mediana è un impulso

Filtro mediano adattivo

Come funziona

Il problema con il rumore impulsivo di grandi dimensioni è che la mediana può essere

un impulso, prima controlla se la mediana è

un impulso, se sì, allarga la finestra

Filtro mediano adattivo

Come funziona

il problema con il rumore impulsivo elevato è che la mediana può essere un

impulso controlla prima se la mediana è

un impulso se sì allarga la

finestra altrimenti controlla se può preservare le informazioni

Filtro mediano adattivo

Come funziona

il problema con il rumore impulsivo elevato è che la mediana può essere un

impulso controlla prima se la mediana è

un impulso se sì allarga la

finestra altrimenti controlla se può preservare le

informazioni controlla se il valore originale è un impulso

Filtro mediano adattivo

Come funziona

il problema con un rumore impulsivo elevato è che la mediana può essere un

impulso controlla prima se la mediana è

un impulso se sì allarga la

finestra altrimenti controlla se può preservare le

informazioni controlla se il valore originale

è un impulso altrimenti lascia il pixel così com'è

Filtro mediano adattivo

Come funziona

il problema con un rumore impulsivo elevato è che la mediana può essere un impulso

controlla prima se la mediana è un impulso

se sì allarga la finestra altrimenti

controlla se può preservare le informazioni controlla

se il valore originale è un impulso altrimenti

lascia il pixel così com'è

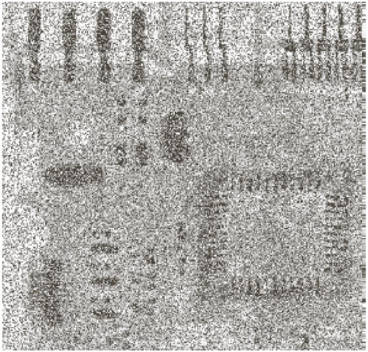
altrimenti utilizzare il valore medio

Filtro mediano adattivo

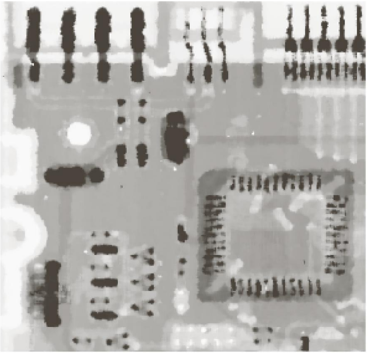
funzione filtro adattivo(I, i, j,r)
N = intorno al raggio r se r > rmax
allora restituisce medN se medN >
minN e medN < maxN allora se Iij > minN e Iij
< maxN allora restituisce Iij

altrimenti restituisce medN
altrimenti
r = r + 1

Filtro mediano adattivo

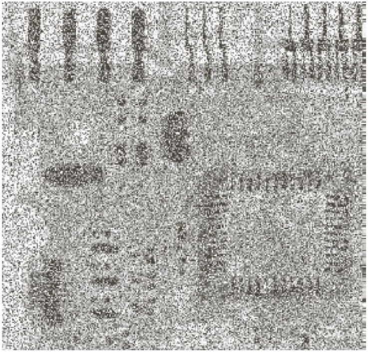


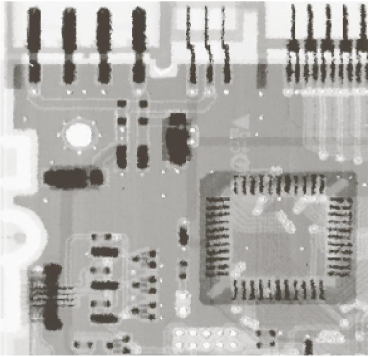
Rumore impulsivoPa = Pb = 0,25



filtro mediano 7×7

Filtro mediano adattivo





Rumore impulsivoPa = Pb = 0,25 filtro mediano adattivo rmax = 7

Filtro di nitidezza

Cosa vogliamo ottenere

Migliora i dettagli nell'immagine.

Filtro di nitidezza

Cosa vogliamo ottenere

Migliora i dettagli nell'immagine.

Ciò migliora le transizioni dei livelli di grigio delle evidenziazioni nelle immagini

Filtro di nitidezza

Cosa vogliamo ottenere

Migliora i dettagli nell'immagine.

Ciò significa migliorare le transizioni dei livelli di grigio evidenziati nelle immagini

Ingrediente principale:

Filtro di nitidezza

Cosa vogliamo ottenere

Migliora i dettagli nell'immagine.

Ciò migliora le transizioni dei livelli di grigio evidenziati nelle immagini

Ingrediente principale: differenziazione spaziale

Derivate di una funzione numerica

Ci concentriamo sulla funzione unidimensionale.

Derivate di una funzione numerica

Ci concentriamo sulla funzione unidimensionale.

$$\lim_{h\text{ÿ}0} \frac{f(x+h)\ \text{ÿ}\ f(x)}{h}$$

Derivate di una funzione numerica

Ci concentriamo sulla funzione unidimensionale.

$$\lim_{h\ddot{y}0} \frac{f(x+h)\ \ddot{y}\ f(x)}{H}$$

Cos'è una funzione digitale?

Derivate di una funzione numerica

Ci concentriamo sulla funzione unidimensionale.

$$\lim_{h\text{ÿ}0} \frac{f(x+h) \text{ ÿ } f(x)}{H}$$

Cos'è una funzione digitale?

I derivati sono definiti in termini di differenze.

Derivate di una funzione numerica

Ci concentriamo sulla funzione unidimensionale.

$$\lim_{h\text{ÿ}0} \frac{f(x+h)\ \text{ÿ}\ f(x)}{H}$$

Cos'è una funzione digitale?

I derivati sono definiti in termini di differenze.

I derivati possono essere definiti in diversi modi.

Derivate di una funzione numerica

Proprietà richieste: derivata prima

deve essere zero a intervalli costanti

deve essere diverso da zero all'inizio di un gradino di intensità o di una rampa

deve essere diverso da zero lungo le rampe

Derivate di una funzione numerica

Proprietà richieste: derivata prima

deve essere zero a intervalli costanti

deve essere diverso da zero all'inizio di un gradino di intensità o di una rampa

deve essere diverso da zero lungo le rampe

Proprietà richieste: derivata seconda

deve essere zero a intervalli costanti

deve essere diverso da zero all'inizio e alla fine di un gradino o rampa di intensità

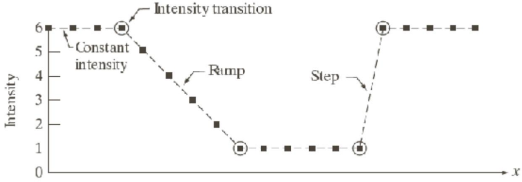
deve essere zero lungo rampe a pendenza costante

Derivate di una funzione numerica

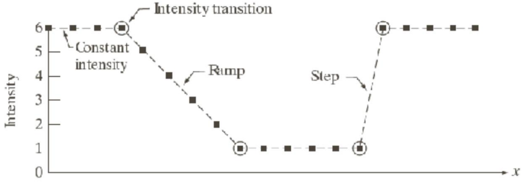
$$f(x) = f(x + 1) \text{ ÿ } f(x)$$

$$f\ (x) = f\ (x + 1) \text{ ÿ } 2f\ (x) + f\ (x \text{ ÿ } 1)$$

Derivate di una funzione numerica

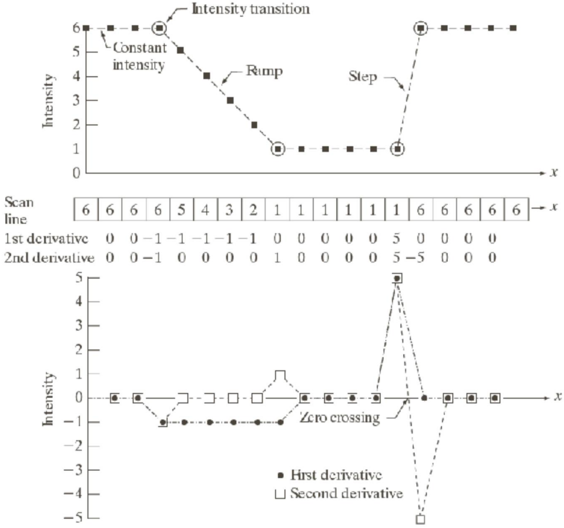


Derivate di una funzione numerica



Scan line	6	6	6	6	5	4	3	2	1	1	1	1	1	1	6	6	6	6	6
1st derivative		0	0	−1	−1	−1	−1	−1	0	0	0	0	0	5	0	0	0	0	
2nd derivative		0	0	−1	0	0	0	0	1	0	0	0	0	5	−5	0	0	0	

Derivate di una funzione numerica



Differenziazione numerica

Scriviamo l'approssimazione polinomiale di Taylor di f (x) in x + h e x ÿ h

$$f(x + h) = f(x) + hf(x) + \frac{1}{2}\,\text{ore}^{2}\, f(x){+}O(h^{3})$$

$$f\ (x\ \text{ÿ}\ h) = f\ (x)\ \text{ÿ}\ hf\ (x) + \frac{1}{\text{ore 2}}{}^{2}\, f(x){+}O(h^{3})$$

Differenziazione numerica

Scriviamo l'approssimazione polinomiale di Taylor di f (x) in x + h e x ÿ h

$$f(x + h) = f(x) + hf(x) + \frac{1}{2}\text{ora e}^{2} f(x)+O(h^{3})$$

$$f (x \text{ ÿ } h) = f (x) \text{ ÿ } hf (x) + \frac{1}{2}\,\frac{2 \text{ ore}}{2} f(x)+O(h^{3})$$

Sottraendo il secondo dal primo

$$f (x) = \frac{f (x + h) \text{ ÿ } f (x \text{ ÿ } h)}{2h} + O(h^{2})$$

Differenziazione numerica

Scriviamo l'approssimazione polinomiale di Taylor di f (x) in x + h e x-h

$$f(x + h) = f(x) + hf(x) + \frac{1}{2} \text{ore}^2 f(x)+O(h^3)$$

$$f (x ÿ h) = f (x) ÿ hf (x) + \frac{1}{2} \text{ore}^2 f(x)+O(h^3)$$

Riassumendo otteniamo

$$\frac{f (x + h) ÿ 2f (x) + f (x ÿ h) f (x) =}{h2} + O(h)$$

Differenziazione numerica

Scriviamo l'approssimazione polinomiale di Taylor di f (x) in x + h e x ÿ h

$$f(x + h) = f(x) + hf(x) + \quad \frac{1}{2} \text{2 ore} \; f(x)+O(h^{3})$$

$$f\ (x\ ÿ\ h) = f\ (x)\ ÿ\ hf\ (x) + \quad \frac{1}{2} \text{2 ore} \; f(x)+O(h^{3})$$

Riassumendo otteniamo

$$\frac{f\ (x + h)\ ÿ\ 2f\ (x) + f\ (x\ ÿ\ h)\ f\ (x) =}{h2} + O(h)$$

Utilizzando più campioni possiamo ricavare formule in cui l'errore di troncamento svanisce più rapidamente.

Differenziazione numerica

Scriviamo l'approssimazione polinomiale di Taylor di f (x) in x + h e x ÿ h

$$f(x + h) = f(x) + hf(x) + \frac{1}{2}\,{}^{2}\,\text{ore}\; f(x)+O(h^{3})$$

$$f (x ÿ h) = f (x) ÿ hf (x) + \frac{1}{2}\,\text{ore}^{2}\; f(x)+O(h^{3})$$

Riassumendo otteniamo

$$\frac{f (x + h) ÿ 2f (x) + f (x ÿ h)\; f (x) =}{h2} + O(h)$$

Utilizzando più campioni possiamo ricavare formule in cui l'errore di troncamento svanisce più rapidamente.

Queste equazioni sono chiamate approssimazioni della differenza centrale: utilizzano campioni da un intervallo simmetrico.

Differenziazione numerica

Derivate prime: approssimazioni alle differenze centrali

= f (x+h)ÿf (xÿh) f (x) + O(h 2h 2)

ÿf (x+2h)+8f (x+h)ÿ8f (xÿh)+f (xÿ2h) f (x) = + O(h 12h 4)

Derivate seconde: approssimazioni alle differenze centrali

f (x+h)ÿ2f (x)+f (xÿh) f (x) = + O(h) h2

ÿf (x+2h)+16f (x+h)ÿ30f (x)+16f (xÿh)ÿf (xÿ2h) f (x) = 2 + O(h 12h 3)

Differenziazione numerica

Usando intervalli asimmetrici otteniamo

Approssimazione in avanti

$$(x) = \frac{f\ (x + h)\ ÿ\ f\ (x)\ f}{} + O(h)h$$

Approssimazione all'indietro

$$(x) = \frac{f\ (x)\ ÿ\ f\ (x\ ÿ\ h)\ f}{H} + O(h)$$

Differenziazione numerica

Usando intervalli asimmetrici otteniamo

Approssimazione in avanti

$$(x) = \frac{f\,(x + h)\ \ddot{y}\ f\,(x)\ f}{} + O(h)h$$

Approssimazione all'indietro

$$(x) = \frac{f\,(x)\ \ddot{y}\ f\,(x\ \ddot{y}\ h)\ f}{H} + O(h)$$

Come otteniamo le formule sopra?

Calcolo delle derivate delle immagini

Calcolo delle derivate delle immagini

In un'immagine calcoliamo

$$\frac{\ddot{y}I}{\ddot{y}x} = I(x + 1, y) \ \ddot{y} \ I(x \ \ddot{y} \ 1, y)$$

$$\frac{\ddot{y}I}{\ddot{y}y} = I(x, y + 1) \ \ddot{y} \ I(x, y \ \ddot{y} \ 1)$$

Calcolo delle derivate delle immagini

In un'immagine calcoliamo

$$\frac{ÿI}{ÿx} = I(x + 1, y)\ ÿ\ I(x\ ÿ\ 1, y)$$

$$\frac{}{ÿIo\ ÿy} = I(x, y + 1)\ ÿ\ I(x, y\ ÿ\ 1)$$

Le derivate possono essere calcolate convolgendo righe e colonne con la maschera

$$[1\ 0\ ÿ\ 1]t$$

Laplaciano

Data una funzione a due variabili il laplaciano è definito come

$$\text{ÿ}2\ f = \frac{\text{ÿ}\ 2\ f\ \text{ÿ} + \text{ÿ}x\ 2}{}\ \text{ÿ}y\ \frac{^{2}\ \text{segg}}{2}$$

Per il caso delle immagini che possiamo scattare

$$\frac{2}{\text{ÿ}\ \text{ÿ}x\ 2} = I(x + 1, y)\ \text{ÿ}\ 2I(x, y) + I(x\ \text{ÿ}\ 1, y)$$

$$\frac{2\text{ÿ}}{Io\ \text{ÿ}y^{2}} = I(x, y + 1)\ \text{ÿ}\ 2I(x, y) + I(x, y\ \text{ÿ}\ 1)$$

poi

$$\text{ÿ}2\ I(x, y) = I(x{+}1, y){+}I(x\text{ÿ}1, y){+}I(x, y{+}1){+}I(x, y\ \text{ÿ}1)\text{ÿ}4I(x, y)$$

Laplaciano: implementazione

0	1	0
1	−4	1
0	1	0

1	1	1
1	−8	1
1	1	1

0	−1	0
−1	4	−1
0	−1	0

−1	−1	−1
−1	8	−1
−1	−1	−1

Laplaciano: implementazione

Qual è il risultato se l'immagine ruota?

Affilatura utilizzando il Laplaciano

L'effetto del Laplaciano è

evidenzia variazioni di intensità

deenfatizza un'area con variazioni lente di intensità

un'immagine scura con discontinuità

evidenziate E se volessimo includere anche lo sfondo, ottenendo un'immagine significativa?

Affilatura utilizzando il Laplaciano

L'effetto del Laplaciano è

- evidenzia variazioni di intensità
- deenfatizza un'area con variazioni lente di intensità
- un'immagine scura con discontinuità evidenziate

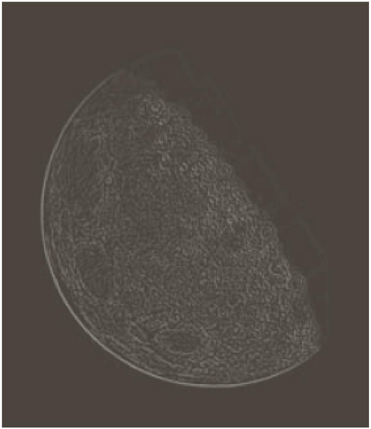
E se volessimo includere anche lo sfondo, ottenendo un'immagine significativa?

$$I(x, y) = I(x, y) + c\ddot{y}2\, I(x, y)$$

dove $c = \pm 1$ a seconda della maschera utilizzata.

Affilatura utilizzando il Laplaciano

Affilatura utilizzando il Laplaciano



Laplaciano

Affilatura utilizzando il Laplaciano

Risultato utilizzando la prima maschera

Affilatura utilizzando il Laplaciano

Risultato utilizzando la seconda maschera

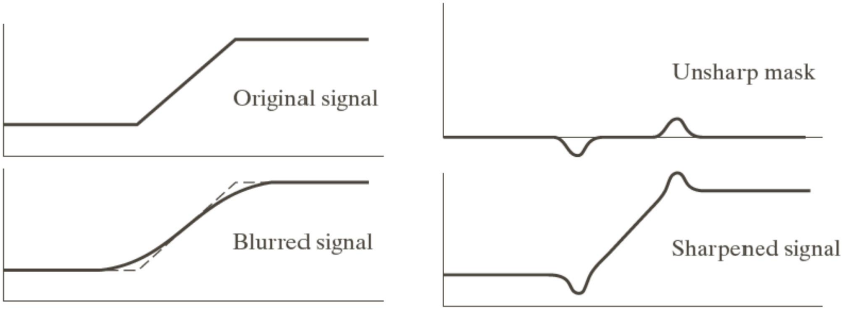
Filtro di contrasto

Un operatore di nitidezza che esalta le discontinuità sottraendo una versione poco nitida dell'immagine.

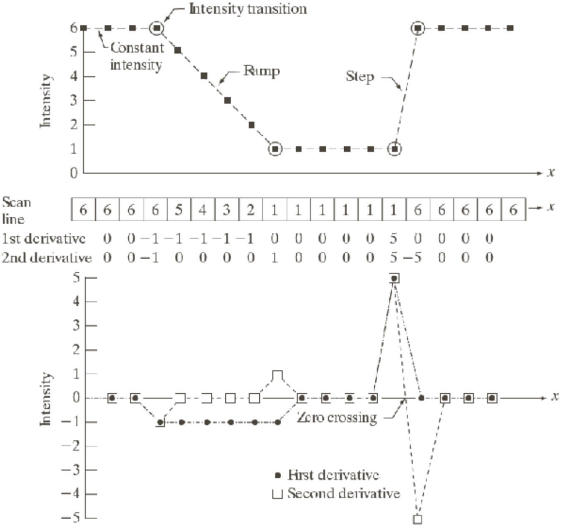
Filtro di contrasto

Un operatore di nitidezza che esalta le discontinuità sottraendo una versione poco nitida dell'immagine. Il processo è il seguente: Sfoca l'immagine originale Sottrai l'immagine sfocata dall'originale Aggiungi il risultato all'originale

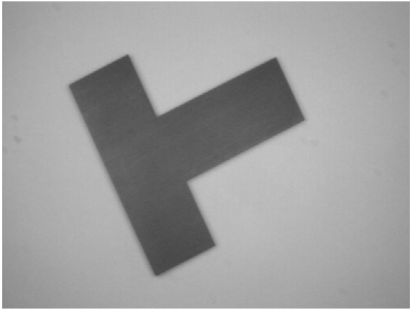
Filtro di contrasto

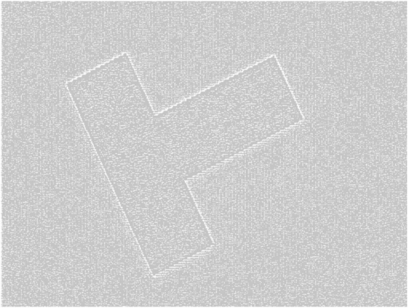


Filtro di contrasto



Filtro di contrasto





Filtro di contrasto