



Università degli Studi di Verona
Dipartimento di Informatica
A.A. 2019-2020

APPUNTI DI “ELABORAZIONE DI IMMAGINI”

Creato da *Davide Zampieri*

RINFORZO NEL DOMINIO SPAZIALE

Istogramma.

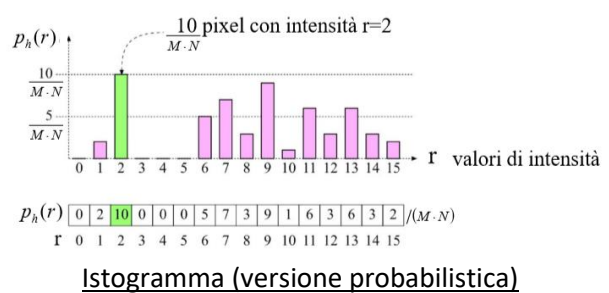
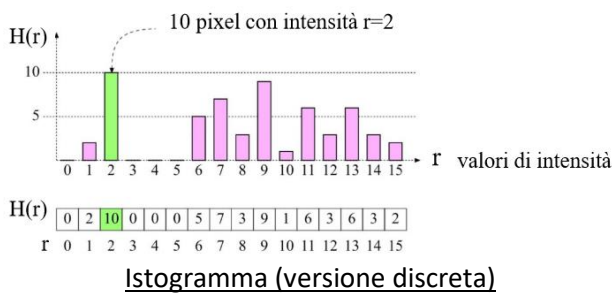
I *pixel* di una immagine sono una popolazione sulla quale possiamo calcolare tutte le *quantità statistiche* descrittive che si usano normalmente (media, mediana, varianza, deviazione standard, quartili, percentili). Particolarmente importante è la conoscenza della distribuzione delle frequenze dei toni di grigio, ovvero l'*istogramma*. Quest'ultimo è uno strumento fondamentale per l'elaborazione delle immagini in quanto vi si riporta, per ogni livello di grigio, il numero di pixel di quel colore. Per una immagine $I[M, N]$ con M righe e N colonne di pixel si ha che $H(r)$ è il numero di pixel di valore r (con $0 \leq r \leq L - 1$) dove:

$$\sum_{r=0}^{L-1} H(r) = M \cdot N$$

L'istogramma è quindi utile a comprendere in maniera immediata le *caratteristiche* dell'immagine. Inoltre, un istogramma può essere visto anche come una *distribuzione di probabilità*:

$$p_h(r) = \frac{H(r)}{M \cdot N}$$

$$\sum_r p_h(r) = 1$$



Attenzione poi al fatto che *immagini diverse* potrebbero avere *istogrammi simili*, in quanto l'istogramma non tiene conto della distribuzione spaziale dei pixel. Perciò, è chiaro che non posso ricostruire una immagine solo dall'istogramma.

Contrasto.

Un attributo di fondamentale importanza per l'elaborazione delle immagini è il *contrasto*, ovvero il rapporto (o differenza) tra il valore più alto (punto più luminoso) e il valore più basso (punto più scuro) della luminosità (o del livello di grigio per immagini a in scala di grigi). In base a come è stata *acquisita*, un'immagine può essere:

- *Sovraesposta*, se presenta i valori più alti a destra dell'istogramma (l'immagine è *chiara*).
- *Sottoesposta*, se presenta i valori più alti a sinistra dell'istogramma (l'immagine è *scura*).

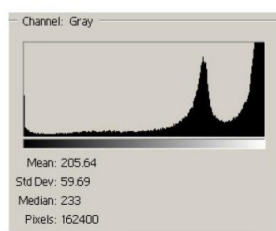


Immagine chiara/sovraesposta

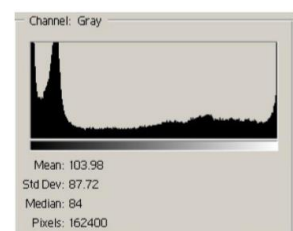


Immagine scura/sottoesposta

Elaborazione delle immagini.

Nel dominio spaziale, l'elaborazione delle immagini può essere espressa come:

$$g(x, y) = T[f(x, y)]$$

dove f è l'immagine di ingresso, g è l'immagine di uscita e T è un operatore su f definito in un intorno di (x, y) . Si identificano quindi tre principali tipi di elaborazione dell'immagine, a seconda dell'intorno di (x, y) che si prende in considerazione:

- *Puntuale*, quando l'intorno coincide con il pixel stesso.
- *Locale*, quando l'intorno è una regione (per esempio quadrata) attorno al pixel (x, y) .
- *Globale*, quando l'intorno è l'intera immagine f .

Operazioni puntuali.

Si dice *operatore puntuale*, un operatore che preso in ingresso il valore di un pixel ne restituisce uno cambiato, il quale dipende esclusivamente dal valore del pixel in ingresso. Le operazioni puntuali servono principalmente per *variare il contrasto*. Infatti, aumentare o diminuire il contrasto significa rendere più o meno evidenti le differenze strutturali dell'oggetto rappresentato. La variazione del contrasto si ottiene andando a cambiare il valore di un pixel con un altro che sia più scuro o più chiaro. Un operatore puntuale può quindi essere rappresentato da una *funzione* che preso in input un valore r lo modifica in un valore $s=T(r)$ con s e r appartenenti allo stesso campo di definizione (per le immagini a scala di grigi, tra 0 e 255). Di seguito vengono elencate le principali operazioni puntuali.

- *Identità*: non fa nulla ($s=r$).
- *Negativo*: viene usata quando si vogliono evidenziare dettagli grigi immersi in zone nere ($s=L-1-r$).
- *Clamping*: limita le intensità ad un range definito $[a, b]$ e serve nel caso in cui ci siano dei pixel di rumore molto chiari o molto scuri che voglio mascherare sostituendovi un altro valore ➤

$$T(r) = \begin{cases} a & \text{se } r < a \\ r & \text{se } a \leq r \leq b \\ b & \text{se } r > b \end{cases}$$

- *Stretching/shrinking*: stira/comprime le intensità di un range $[r_{min}, r_{max}]$ ad un range definito $[a, b]$ dove r_{min} e r_{max} sono il più piccolo e il più grande livello di grigio del range che voglio trattare ➤

$$s = \left[\frac{r - r_{min}}{r_{max} - r_{min}} \right] [b - a] + a$$

- *Trasformazione logaritmica*: viene usata quando si vogliono mappare fasce strette di valori dell'immagine originale in fasce più ampie, aumentandone così il range del contrasto e rendendo l'interpretazione umana più informativa ➤

$$s = c \log(1 + r) \quad c = \frac{L-1}{\log(L)}$$

- *Trasformazione logaritmica inversa*: permette di aumentare il range di una fascia determinata di livelli di grigio chiari ➤

$$s = (e^r)^{\frac{1}{c}} - 1 \quad c = \frac{L-1}{\log(L)}$$

- *Trasformazione di potenza*: viene usata quando si vuole aumentare il range dei pixel chiari per ottenere un generale oscuramento dell'immagine (analogamente alla trasformazione logaritmica inversa) oppure quando si vuole aumentare il range dei pixel scuri per ottenere un generale schiarimento dell'immagine (analogamente alla trasformazione logaritmica) ➤

$$s = cr^\gamma$$

- **Binarizzazione:** si ottiene scegliendo una soglia e mettendo a nero tutti i pixel il cui valore è minore ad essa e a bianco tutti gli altri, così facendo si produce un'immagine che ha solo due livelli (nero e bianco), che serve di solito per discriminare un oggetto dalla scena (utile nelle immagini biomedicali e nella videosorveglianza).
- **Equalizzazione:** operazione puntuale non lineare (vedi prossimo paragrafo).

Equalizzazione.

Si parla di *immagine equalizzata* quando il contributo di ogni differente tonalità di grigio è simile, ovvero l'*istogramma* tende ad essere *uniforme* (o appiattito). L'equalizzazione si ottiene usando appositi *algoritmi*. Attenzione al fatto che non sempre l'equalizzazione migliora l'immagine. L'idea è quella di vedere l'istogramma come una distribuzione, e di voler rendere tale distribuzione il più possibile simile a quella uniforme. Avere un istogramma piatto, infatti, assicura a livello percettivo una risposta del cervello migliore (in termini di numero di dettagli che si riescono a riconoscere), per cui l'immagine diventa più piacevole da osservare ↘

- Se r_k è il k -esimo livello di grigio, $k=0, \dots, L-1$ e $H(r_k)$ il corrispondente conteggio dato dall'istogramma dell'immagine di dimensioni $M \times N$, posso definire

$$p_r(r_k) = \frac{H(r_k)}{MN}$$

- L'equalizzazione dell'istogramma è la seguente funzione T , con s_k k -esimo valore di grigio in cui si mappa r_k :

$$s_k = T(r_k) = (L-1) \sum_{j=0}^k p_r(r_j) \quad \text{somma cumulativa o funzione di ripartizione probabilistica}$$

$$= \frac{(L-1)}{MN} \sum_{j=0}^k H(r_j) = \frac{\sum_{j=0}^k H(r_j)}{\frac{MN}{(L-1)}}$$

L'*algoritmo di equalizzazione* è quindi il seguente:

1. Calcolo delle L somme cumulative dei valori dell'istogramma visto come distribuzione (per ciascun valore di k).
2. Moltiplicazione per il valore massimo dei livelli di grigio $L - 1$.
3. Normalizzazione dei valori calcolati al passo 1 dividendo per il numero totale di pixel $M \cdot N$ e arrotondamento.
4. Applicazione del mapping T ottenuto.

- Esempio: img con 8 diversi livelli di grigio

Istogramma originale

ldg	0	1	2	3	4	5	6	7
# pixel	10	8	9	2	14	1	5	2

Passo 1: somma cumulativa

ldg	0	1	2	3	4	5	6	7
# pixel	10	18	27	29	43	44	49	51

Passo 2: normalizzazione mediante numero totale di pixel

ldg	0	1	2	3	4	5	6	7
# pixel	10/51	18/51	27/51	29/51	43/51	44/51	49/51	51/51

Passo 3-4: prodotto x il ldg massimo (7), arrotondamento all'intero più vicino e mappatura

ldg orig.	0	1	2	3	4	5	6	7
ldg eq.	1	2	4	4	6	6	7	7

Tutti pixel che avevano $ldg=0$ sono mappati al $ldg=1$, quelli che avevano $ldg=1$ sono mappati al $ldg=2$, etc.

Operazioni locali.

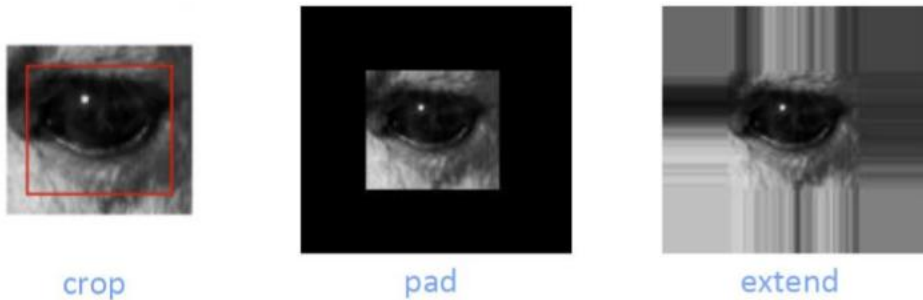
Nelle *operazioni locali*, il valore d'uscita di ogni pixel dipende da un limitato intorno del corrispondente punto in input. Tali operazioni sono usate per *migliorare la qualità* delle immagini o per *estrarre delle informazioni* dall'immagine. Le operazioni locali si possono quindi pensare come filtri spaziali dell'immagine. Un *filtraggio* è spesso ottenuto facendo la convoluzione tra l'immagine ed una matrice, chiamata maschera o kernel (in tal caso si parla di filtri lineari). Gli *intorni* presi maggiormente in considerazione per i filtri spaziali sono di grandezza $K \times K$, con K dispari (per fare in modo di considerare uniformemente i pixel attorno al punto di applicazione). I pixel al di fuori dell'intorno non prendono parte alla funzione. Le due principali categorie di operazioni locali sono:

- *Filtraggio lineare*, se T può essere visto come combinazione lineare dei valori dei pixel nel vicinato (ad esempio la media dei pixel nell'intorno).
- *Filtraggio non lineare*, se T contiene operazioni non lineari sulle variabili indipendenti (ad esempio la mediana dei pixel nell'intorno).

Filtri lineari.

Fintanto che l'intorno è definito all'interno dell'immagine, non ci sono problemi nell'applicare questo tipo di filtri. Quando invece l'intorno cade fuori dall'immagine, ci sono tre possibili soluzioni:

- *Cropping*, ovvero filtro solo dove l'intorno cade all'interno dell'immagine (in questo modo però non filtro tutta l'immagine).
- *Zero-padding*, ovvero contorno l'immagine con dei bordi di pixel neri.
- *Replicazione*, ovvero contorno l'immagine con degli artefatti (ottenendo un'immagine non realistica).



Rumore nelle immagini.

Il *rumore* è un disturbo dell'immagine, introdotto dal sistema di acquisizione o dal mezzo di propagazione, che ne degrada la qualità. Il rumore è tipicamente *modellato* come additivo e casuale:

$$\tilde{f}(n, m) = f(n, m) + \varepsilon(n, m)$$

dove f è l'immagine priva di rumore e ε è un processo aleatorio che genera delle quantità che seguono una distribuzione particolare, indipendentemente da dove il processo è collocato nell'immagine (ossia indipendentemente da n e m). Esistono due *tipi* principali di rumore, quello gaussiano additivo bianco e quello impulsivo.

Signal-to-Noise Ratio (SNR).

La quantità di rumore può essere stimata attraverso la misura del *SNR*, la cui versione più usata è la cosiddetta *mean square*:

$$SNR_{ms} = \frac{\sum_{n=1}^N \sum_{m=1}^M \tilde{f}(n, m)^2}{\sum_{n=1}^N \sum_{m=1}^M [\tilde{f}(n, m) - f(n, m)]^2}$$

Una definizione alternativa di *SNR* può essere stimata con la *deviazione standard*:

$$SNR = \frac{\sigma_s}{\sigma_n}$$

dove σ_s è la deviazione standard del segnale e σ_n è la deviazione standard dell'immagine affetta da rumore.

Rumore gaussiano additivo bianco.

È un *processo stocastico*, ovvero una variabile aleatoria che emette valori casuali nello spazio attraverso la funzione $\varepsilon(n,m)$, che si somma al segnale pulito e i cui valori (non periodici) vengono prodotti secondo una *distribuzione gaussiana* di media pari a 0, ed una particolare varianza σ^2 dove più è alta tale varianza, più è rumorosa l'immagine finale (più distanti da 0 saranno i numeri prodotti e sommati all'immagine pulita).

Rumore impulsivo.

È anche chiamato rumore *sale e pepe*, ed è causato da *alterazioni brusche nel segnale*. Viene parametrizzato da un fattore D (percentuale) che è la *densità* con cui esso si localizza sui pixel dell'immagine. Quindi maggiore è il valore di densità D , maggiore sarà il numero di pixel affetti.

Tipologie di filtraggio.

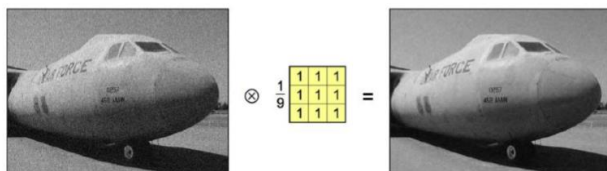
Esistono tre tipologie principali di filtraggio:

- *Smoothing*, che servono per aumentare il SNR, ossia a levare il rumore (eliminando però alcuni dettagli).
- *Sharpening*, che servono per aumentare il grado di dettaglio delle immagini (incrementando però il rumore).
- *Estrazione di caratteristiche*, che servono per estrarre rappresentazioni alternative delle immagini di partenza che ne evidenzino aspetti particolari.

Filtraggi di smoothing.

Di seguito vengono elencati i principali filtraggi di smoothing.

- *Filtro media*: è un filtraggio *lineare* che serve per levare il *rumore gaussiano* e che si attua attraverso la *convoluzione* dell'immagine con una maschera di media; in pratica, dato un intorno di applicazione, esso calcola la media dei valori ivi compresi e la sostituisce al posto del valore nel punto di applicazione; percettivamente, l'immagine risultato è più addolcita rispetto ai bordi degli oggetti perché al posto di un pixel ho sostituito una media calcolata sui vicini (strutture sottili e vicine vengono mischiate).
- *Filtro mediano*: è un filtraggio *non lineare* che serve per levare il *rumore impulsivo* e che si realizza attraverso un *algoritmo*; in pratica, la mediana è una stima robusta della media e quindi se nel calcolo della media è presente del rumore impulsivo essa ne sarà affetta mentre la mediana no.
- *Filtro gaussiano*: è simile al filtraggio di media con la differenza che usa una *media pesata* in cui i pesi più vicini al centro della maschera hanno valori più alti; in pratica, il filtro gaussiano effettua uno smoothing più lieve (preservando i contorni meglio di quanto faccia il filtro media), ma leva meno rumore.



Rumore gaussiano e filtro media



Rumore impulsivo e filtro mediano

Filtraggi di sharpening.

I filtri di sharpening sono detti anche *filtri di derivata* poiché calcolano numericamente, nell'intorno in cui sono definiti, la derivata locale (prima o seconda) dell'immagine. Questo perché la *derivata prima* è nulla in regioni di intensità costante e non nulla in presenza di variazioni di intensità, mentre la *derivata seconda* è nulla in regioni di intensità costante e in presenza di variazioni costanti di intensità (rampe) e non nulla in presenza di variazioni non costanti (all'inizio e alla fine di rampe). Di seguito vengono elencati i principali filtri di sharpening.

- *Basic highpass spatial filtering*: è un filtraggio *lineare* in cui la maschera di filtraggio (detta laplaciana) è caratterizzata da coefficienti positivi vicino al centro e negativi nella periferia esterna; in pratica, quando il filtro passa su regioni con livelli di grigio quasi stabili, l'output della maschera è 0 o molto piccolo, mentre quando il valore centrale differisce dalla periferia, l'output della maschera è alto (l'immagine di output non assomiglierà a quella originale).
- *Sharpening con Laplaciano*: è un importante filtraggio *lineare* basato sulla derivata seconda che utilizza il filtraggio precedente per creare un'immagine con gli edge amplificati riuscendo però a farla rimanere realistica (simile a quella di partenza).
- *High boost filtering*, è un filtraggio che amplifica gli edge dando però maggiore libertà al progettista (il blur può avvenire attraverso una maschera di supporto arbitrariamente grande).



Basic highpass spatial filtering



Sharpening con Laplaciano

RINFORZO NEL DOMINIO DELLE FREQUENZE

Filtraggio nel dominio delle frequenze.

Le informazioni a *basse frequenze* corrispondono a parti dell'immagine in cui abbiamo *variazioni di intensità lente*. Le informazioni ad *alte frequenze* corrispondono invece a parti in cui abbiamo *variazioni repentine*. Un *filtro passa basso* rimuove dall'immagine le informazioni ad alte frequenze e mantiene quelle a basse frequenze. Un *filtro passa alto* invece rimuove dall'immagine informazioni a basse frequenze e mantiene quelle ad alte frequenze. Filtri passa alto possono essere derivati da filtri passa basso (e viceversa).

Filtri passa basso (PB).

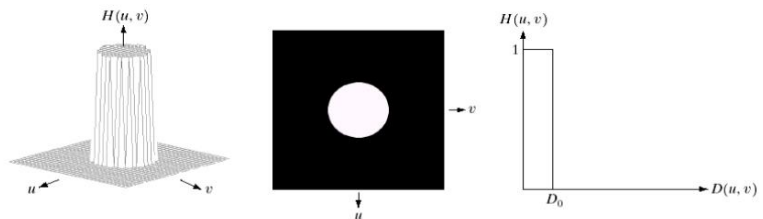
Il rumore porta ad un aumento delle alte frequenze. Riducendo le alte frequenze riduciamo il rumore, ma anche il livello di dettaglio. Un *filtro passa basso* serve quindi per attenuare le alte frequenze. Di seguito vengono elencati i principali filtri passa basso.

- Filtro passa basso *ideale*: c'è una *brusca* transizione in corrispondenza della frequenza di taglio; tale fenomeno è detto *ringing* (o effetto di Gibbs) ed è dovuto al fatto che filtrare con un PB ideale (in frequenza) equivale a fare una convoluzione con un *sinc* (nello spazio).
- Filtro passa basso di *Butterworth*: c'è un'attenuazione *dolce* in prossimità della frequenza di taglio; la ripidità viene modellata dall'*ordine* del filtro ma la frequenza di taglio viene selezionata indipendentemente da esso.
- Filtro passa basso *Gaussiano*: c'è un'attenuazione *dolce* in prossimità della frequenza di taglio, che corrisponde alla *deviazione standard*.

Filtro passa basso "ideale"

$$H(u, v) = \begin{cases} 1 & \text{se } D(u, v) \leq D_0 \\ 0 & \text{se } D(u, v) > D_0 \end{cases}$$

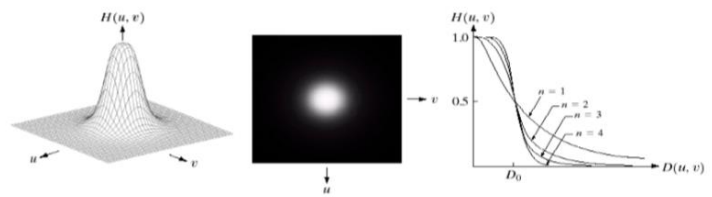
$$D(u, v) = \sqrt{(u - M/2)^2 + (v - N/2)^2}$$



Filtro passa basso di Butterworth

$$H(u, v) = \frac{1}{1 + [D(u, v)/D_0]^{2n}}$$

$$D(u, v) = \sqrt{(u - M/2)^2 + (v - N/2)^2}$$



Filtro passa basso Gaussiano

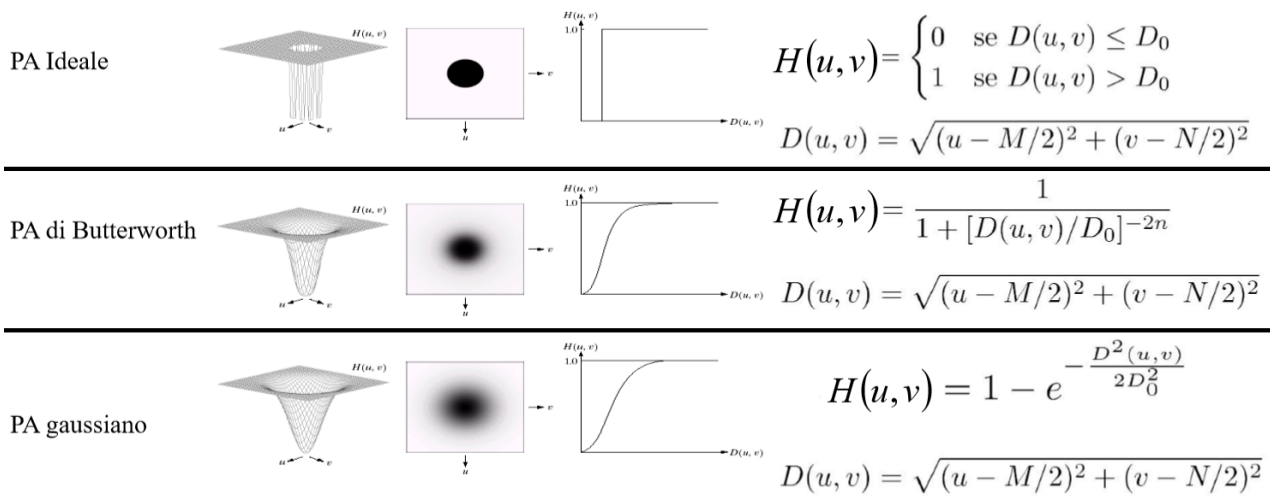
$$H(u, v) = e^{-\frac{D^2(u, v)}{2D_0^2}}$$

$$D(u, v) = \sqrt{(u - M/2)^2 + (v - N/2)^2}$$



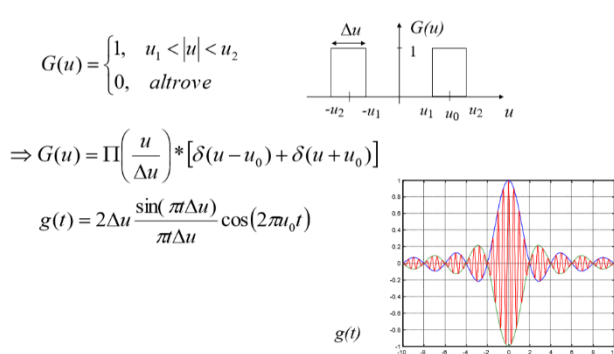
Filtri passa alto (PA).

Un *filtro passa alto* sopprime (blocca) le basse frequenze e lascia passare le alte frequenze.

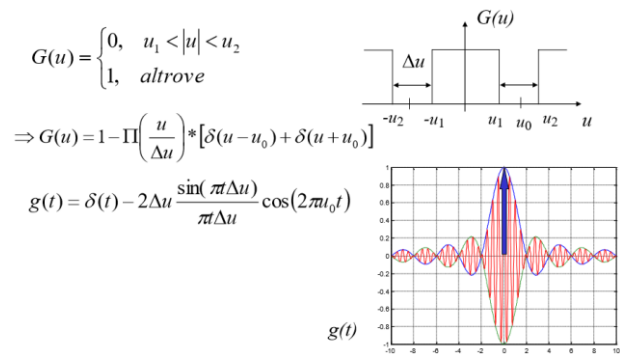


Filtri passa banda e ferma banda.

È una classe di filtri che agiscono su una *banda di frequenze* anziché discriminare solo tra frequenze alte e basse. Un *filtro passa banda* sopprime tutte le frequenze al di fuori di un intervallo di frequenze specificato. Un *filtro ferma banda* invece sopprime tutte le frequenze nell'intervallo specificato. In modo simile a quanto visto per i filtri passa basso e alto, si può derivare un filtro passa banda da un filtro ferma banda (e viceversa).



Filtro passa-banda ideale

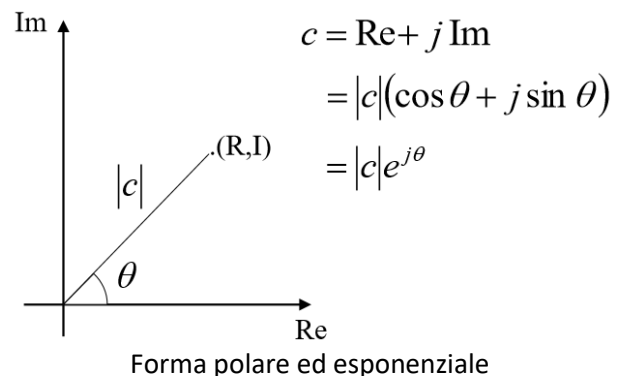
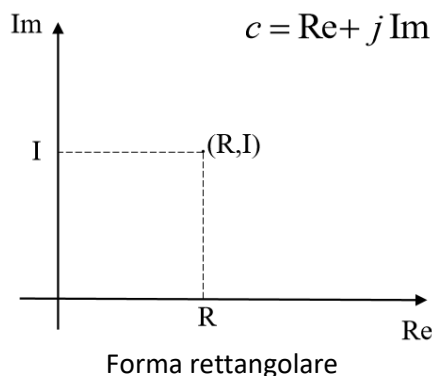


Filtro ferma-banda ideale

DOMANDE SULLA TEORIA

Quanti modi esistono per descrivere un numero complesso? Dare un esempio numerico di come un numero complesso possa essere espresso, secondo i vari modi possibili.

- *Forma rettangolare*: i numeri complessi possono essere visti geometricamente come punti su un piano (piano complesso) descritti da coordinate (R, I) .
- *Forma polare*: si ottiene rappresentando i numeri complessi in coordinate polari (*modulo, angolo*).
- *Forma esponenziale*: grazie alla formula di Eulero si riesce a scrivere la forma polare di un numero complesso in maniera alternativa.



Cos'è un fasore? Dare un esempio numerico di fasore.

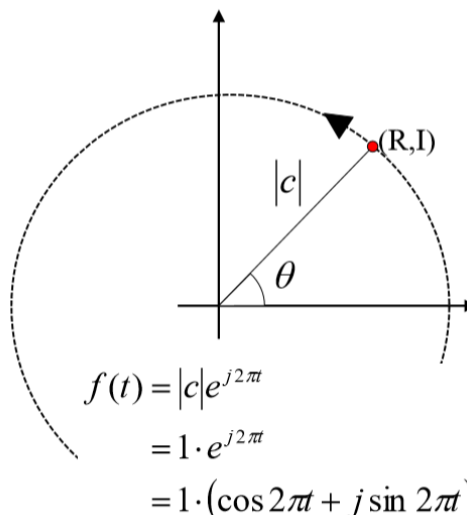
Un *fasore* è una funzione complessa di variabile reale che modella la posizione di un punto che ruota attorno all'origine con raggio determinato $|c|$ e velocità angolare costante $\theta(t)$. Con i soli numeri reali sarebbe più dispendioso, in termini di numero di funzioni da usare, rappresentare un fasore. I fasori permettono di passare dal dominio del tempo/spazio a quello dell'*analisi frequenziale*. Ciò è essenziale per la comprensione di un segnale, per la trasmissione e per il filtraggio. In definitiva, un fasore fa variare nel tempo un numero complesso (in forma polare) mantenendone il modulo fisso; inoltre, $\theta(t)$ indica l'angolo spazzato a partire da un angolo φ ad un certo istante t , ossia la velocità angolare definita in questo modo:

$$\theta(t) = \frac{2\pi}{T_0} \cdot t + \varphi$$

dove T_0 indica il tempo in cui spazza 2π radianti.

Supponiamo

- coordinate R, I
- distanza fissa dall'origine unitaria $|c|=1$
- velocità angolare costante pari a $2\pi/\text{sec}$.
ossia $\theta(t) = 2\pi t$, $T_0 = 1\text{s}$.
- a $t=0$, $\theta=0$
- $\phi = 0$



Descrivere cos'è l'istogramma di un'immagine, nella sua versione classica e probabilistica.

Un istogramma nella sua *versione discreta* (a scala di grigi) è una funzione H tale che $H(r)$ è il numero di pixel con tonalità di colore r (con $0 \leq r \leq L - 1$ dove L è il valore massimo dei livelli di grigio). Per una generica immagine $I[M, N]$ con M righe e N colonne di pixel deve valere che:

$$\sum_{r=0}^{L-1} H(r) = M \cdot N$$

ovvero la sommatoria dei pixel per tutte le tonalità è uguale a $M \cdot N$, cioè il numero di pixel totali dell'immagine. Può essere inoltre visto come una *distribuzione di probabilità*:

$$P(r_k) = \frac{H(r_k)}{M \cdot N}$$

con r_k k -esimo valore di grigio e tale che:

$$\sum_{k=0}^{L-1} P(r_k) = 1$$

Descrivere l'algoritmo che permette l'equalizzazione dell'istogramma.

La trasformazione che descrive l'*equalizzazione* dell'istogramma è la seguente:

$$s_k = T(r_k) = (L - 1) \cdot \sum_{j=0}^k P(r_j) = \frac{L - 1}{M \cdot N} \cdot \sum_{j=0}^k H(r_j)$$

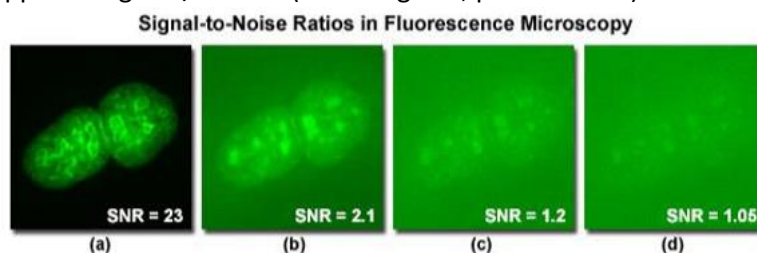
L'*algoritmo* quindi calcola le L somme cumulative per ciascun valore di k ; moltiplica per $L - 1$; normalizza la sommatoria dividendo per $M \cdot N$, ovvero il numero totale di pixel, e arrotonda il risultato; applica la mappatura T per cui il k -esimo valore viene mappato nel s_k -esimo valore corrispondente.

Dare una definizione di SNR. In generale, meglio avere SNR_{ms} alto o basso?

La quantità di rumore può essere stimata attraverso la misura del *Signal-to-Noise Ratio* (SNR). La versione di SNR più usata è la cosiddetta *mean square*:

$$SNR_{ms} = \frac{\sum_{n=1}^N \sum_{m=1}^M \tilde{f}(n, m)^2}{\sum_{n=1}^N \sum_{m=1}^M [\tilde{f}(n, m) - f(n, m)]^2}$$

Per levare il rumore, bisogna aumentare il SNR. Quindi, in generale, è meglio avere un SNR_{ms} *alto*, in quanto esso rappresenta il rapporto segnale/rumore (tanto segnale, poco rumore).



Descrivere i possibili metodi di riduzione/rimozione del rumore in un'immagine, ossia i metodi per eseguire un'operazione di filtraggio passa basso, evidenziando caratteristiche ed effetti collaterali di ogni metodo.

- Filtro passa basso *ideale*: c'è una *brusca* transizione in corrispondenza della frequenza di taglio; tale fenomeno è detto *ringing* (o effetto di Gibbs) ed è dovuto al fatto che filtrare con un PB ideale (in frequenza) equivale a fare una convoluzione con un *sinc* (nello spazio).
- Filtro passa basso di *Butterworth*: c'è un'attenuazione *dolce* in prossimità della frequenza di taglio; la ripidità viene modellata dall'*ordine* del filtro ma la frequenza di taglio viene selezionata indipendentemente da esso.
- Filtro passa basso *Gaussiano*: c'è un'attenuazione *dolce* in prossimità della frequenza di taglio, che corrisponde alla *deviazione standard*.

Descrivere che cos'è un filtro passa alto.

Un *filtro passa alto* è un sistema che permette il passaggio di frequenze al di sopra di una determinata soglia, detta *frequenza di taglio*, bloccando le basse frequenze.

Descrivere cos'è il rinforzo (enhancement) di un'immagine e descrivere la differenza tra operazione di rinforzo puntuale e locale.

Il *rinforzo delle immagini* può essere visto come:

$$g(x, y) = T[f(x, y)]$$

dove f è l'immagine di ingresso, g è l'immagine di uscita e T è un operatore su f definito in un intorno di (x, y) . A seconda dell'intorno di (x, y) che si considera, le operazioni di rinforzo possono essere di tipo:

- *Puntuale*, quando l'intorno coincide con il pixel stesso.
- *Locale*, quando l'intorno è una regione (per esempio quadrata) attorno al pixel (x, y) .

Descrivere l'operazione di trasformazione logaritmica e dire in che caso viene applicata di norma tale operazione.

La *trasformazione logaritmica* è un'operazione puntuale che mappa fasce strette di valori dell'immagine originale in fasce più ampie, aumentandone così il range del contrasto e rendendo l'interpretazione umana più informativa. Viene applicata quando si vuole aumentare il range dei pixel scuri per ottenere un generale *schiarimento dell'immagine* oppure quando si vuole aumentare il range dei pixel chiari per ottenere un generale oscuramento dell'immagine (trasformazione inversa).

Dire a che tipo di rumore è soggetta l'immagine i) e dire a quali operazioni sono state sottoposte le immagini ii) e iii) a partire dall'immagine originale i).



i)



ii)



iii)

L'immagine i) è soggetta a *rumore impulsivo* (sale e pepe), il quale è causato da alterazioni brusche nel segnale e viene parametrizzato da un fattore D che è la *densità* con cui esso si localizza sui pixel dell'immagine (maggiore è il valore di densità D , maggiore sarà il numero di pixel affetti).

L'immagine ii) è stata sottoposta ad un filtraggio di smoothing detto *filtro mediano*, ovvero un filtraggio non lineare che serve per levare il rumore impulsivo e che si realizza attraverso un algoritmo.

L'immagine iii), invece, è stata sottoposta ad un filtraggio di smoothing detto *filtro media*, ovvero un filtraggio lineare che si attua attraverso la convoluzione dell'immagine con una maschera di media; in pratica, dato un intorno di applicazione, esso calcola la media dei valori ivi compresi e la sostituisce al posto del valore nel punto di applicazione; percettivamente, l'immagine risultato è più addolcita rispetto ai bordi degli oggetti perché al posto di un pixel ho sostituito una media calcolata sui vicini.

Si può notare che l'immagine filtrata con il filtro media è ancora affetta da rumore perché se nel calcolo della media è presente del rumore impulsivo essa ne rimane affetta mentre la mediana, essendo una stima robusta della media, no.

Descrivere la differenza tra filtraggio lineare e non lineare e dettagliare una tecnica di filtraggio lineare.

Le operazioni locali, in cui il valore d'uscita di ogni pixel dipende da un limitato intorno del corrispondente punto in input, si possono pensare come filtri spaziali dell'immagine usati per migliorarne la qualità o per estrarne delle informazioni. Le due principali categorie di operazioni locali sono:

- *Filtraggio lineare*, se la trasformazione può essere vista come combinazione lineare dei valori dei pixel nel vicinato (ad esempio la media dei pixel nell'intorno).
- *Filtraggio non lineare*, se la trasformazione contiene operazioni non lineari sulle variabili indipendenti (ad esempio la mediana dei pixel nell'intorno).

Una tecnica di filtraggio lineare è il cosiddetto *filtro media* che si attua attraverso la convoluzione dell'immagine con una maschera di media; in pratica, dato un intorno di applicazione, esso calcola la media dei valori ivi compresi e la sostituisce al posto del valore nel punto di applicazione; percettivamente, l'immagine risultata è più addolcita rispetto ai bordi degli oggetti perché al posto di un pixel ho sostituito una media calcolata sui vicini (strutture sottili e vicine vengono mischiate).

Si dia la definizione di istogramma di un'immagine. Si illustrino poi le operazioni principali che si possono eseguire su un istogramma, il loro effetto sulle immagini e in quali circostanze possono risultare utili.

L'*istogramma* è la distribuzione delle frequenze dei toni di grigio all'interno di un'immagine, ed è uno strumento fondamentale per l'elaborazione delle immagini in quanto vi si riporta, per ogni livello di grigio, il numero di pixel di quel colore. Per una immagine $I[M, N]$ con M righe e N colonne di pixel si ha che $H(r)$ è il numero di pixel di valore r (con $0 \leq r \leq L - 1$) dove:

$$\sum_{r=0}^{L-1} H(r) = M \cdot N$$

Le operazioni principali che si possono eseguire su un istogramma sono le cosiddette *operazioni puntuali*, ovvero quelle che, preso in ingresso il valore di un pixel, ne restituiscono uno cambiato che dipende esclusivamente dal valore del pixel in ingresso. Le operazioni puntuali servono principalmente per variare il contrasto, andando a cambiare il valore di un pixel con un altro che sia più scuro o più chiaro. Di seguito vengono elencate le principali operazioni puntuali.

- *Identità*: non fa nulla.
- *Negativo*: viene usata quando si vogliono evidenziare dettagli grigi immersi in zone nere.
- *Clamping*: limita le intensità ad un range definito e serve nel caso in cui ci siano dei pixel di rumore molto chiari o molto scuri che voglio mascherare sostituendovi un altro valore.
- *Stretching/shrinking*: stira/comprime le intensità di un range di livelli di grigio che voglio trattare ad un range definito.
- *Trasformazione logaritmica*: viene usata quando si vogliono mappare fasce strette di valori dell'immagine originale in fasce più ampie, aumentandone così il range del contrasto e rendendo l'interpretazione umana più informativa.
- *Trasformazione logaritmica inversa*: permette di aumentare il range di una fascia determinata di livelli di grigio chiari.
- *Trasformazione di potenza*: viene usata quando si vuole aumentare il range dei pixel chiari/scuri per ottenere un generale oscuramento/schiarimento dell'immagine.
- *Binarizzazione*: si ottiene scegliendo una soglia e mettendo a nero tutti i pixel il cui valore è minore ad essa e a bianco tutti gli altri, così facendo si produce un'immagine che ha solo due livelli (nero e bianco), che serve di solito per discriminare un oggetto dalla scena (utile nelle immagini biomedicali e nella videosorveglianza).
- *Equalizzazione*: operazione puntuale non lineare che si ottiene usando appositi algoritmi e consiste nel vedere l'istogramma come una distribuzione, e di voler rendere tale distribuzione il più possibile simile a quella uniforme; avere un istogramma piatto, infatti, assicura a livello percettivo una risposta del cervello migliore per cui l'immagine diventa più piacevole da osservare.