29-11-2022

ENTROPIA

- L'entropia è legata alle **statistica**.
- L'idea è: presa una sorgente di informazioni l'entropia indica il **numero di bit medio per rappresentare un simbolo** emesso da quella sorgente ed è legata la probabilità di presentazione di ogni simbolo.
- 2 simboli che hanno la stessa probabilità avranno entropia = 1, cioè bit in media per rappresentare *l'esito*.

> Non è possibile mai ottenere un istogramma piatto.
> Lo scopo è produrre un'immagine che si avvicini il più possibile al caso ideale.
> Per "*avvicinarsi*" al caso ideale ci si riferisce alle frequenze di ogni livello di grigio.

ALGORITMO DI EQUALIZZAZIONE

$M \times N \leftarrow$ Numero di pixel in un'immagine

$n_k \leftarrow$ numero di volte con cui compare il livello di grigio r_k

$P(r_k) = \dfrac{n_k}{M \times N} \leftarrow$ risultato compreso fra [0,1]

cioè la somma delle probabilità deve essere uguale a 1

$P(r_0) + P(r_1) + \ldots + P(r_{255}) = 1$

- La normalizzazione di P (cioè la divisione per MN) serve per indicare i livelli di grigio in termini relativi e non in termini assoluti.
- Dopo l'equalizzazione si avrà un livello di grigio s_k che sarà il valore in cui r_k si dovrà trasformare dopo tale procedimento:
  - $s_k = L - 1 \sum_{j=0}^{k} P(r_j)$ con:
  - $P(r_j) =$ è la probabilità P che il livello r_j si presenti;
  - L è il valore più elevato che può assumere un livello di grigio (solitamente 256).
  Quindi si avrà un algoritmo del tipo:
- $r_0 \rightarrow$ equalizzazione $\rightarrow s_0$
- $r_1 \rightarrow$ equalizzazione $\rightarrow s_1$
- e questi risultati dipenderanno dal risultato dell'equalizzazione

Esempio 1 di applicazione:

$$S_{255} = (L-1)\sum_{j=0}^{255} p(r_j) = (L-1)\cdot 1 = 255$$

Se l'immagine è a 8 bit (quindi un solo livello di grigio) allora L = 256

Allora $r_{255} \rightarrow S_{255} = 255$

$$S_k = (L-1)\sum_{j=0}^{k} \boxed{p(r_k)}$$

$\downarrow$ SOSTITUISCO $p(r_k) = \frac{n_k}{M \cdot N}$

$$S_k = L-1 \sum_{j=0}^{k} \boxed{\frac{n_k}{M \cdot N}}$$

Esempio 2 di applicazione:

Sia data un'immagine a 3 bit ($L = 8$) con 64x64 pixel ($M \times N = 4096$) con la seguente distribuzione di intensità:

r_k	n_k	$p_r(r_k) = n_k/MN$
$r_0 = 0$	790	0.19
$r_1 = 1$	1023	0.25
$r_2 = 2$	850	0.21
$r_3 = 3$	656	0.16
$r_4 = 4$	329	0.08
$r_5 = 5$	245	0.06
$r_6 = 6$	122	0.03
$r_7 = 7$	81	0.02

- Il livello $r_0 = 0$ si presenta n_k volte = 790 volte
- il livello $r_1 = 1$ si presenta n_k volte = 1023 volte

- ecc..
  - Applicando la formula si ha:

$$s_0 = T(r_0) = 7\sum_{j=0}^{0} p_r(r_j) = 7p_r(r_0) = 1.33$$

$$s_1 = T(r_1) = 7\sum_{j=0}^{1} p_r(r_j) = 7p_r(r_0) + 7p_r(r_1) = 3.08$$

$$s_2 = 4.55, s_3 = 5.67, s_4 = 6.23, s_5 = 6.65, s_6 = 6.86, s_7 = 7.00.$$

Arrotondando:

$s_0 = 1.33 \rightarrow 1$	$s_4 = 6.23 \rightarrow 6$
$s_1 = 3.08 \rightarrow 3$	$s_5 = 6.65 \rightarrow 7$
$s_2 = 4.55 \rightarrow 5$	$s_6 = 6.86 \rightarrow 7$
$s_3 = 5.67 \rightarrow 6$	$s_7 = 7.00 \rightarrow 7$

$r_0 \Rightarrow s_0$, $r_0 = 0$, $s_0 = 1{,}33$

$S_0 =$ BANALE

$r_1 \Rightarrow s_1$, $r_1 = 1$, $s_1 = 3{,}08$

$$S_1 = 7\sum_{j=0}^{1} P(r_j) = 7\cdot\left(\left(P(r_0)+P(r_1)\right)\right) = $$
$$= 7\cdot(0.19+0{,}25) = 3{,}08$$

$r_2 \Rightarrow s_2$, $r_2 = 2$, $s_2 = 4{,}55$

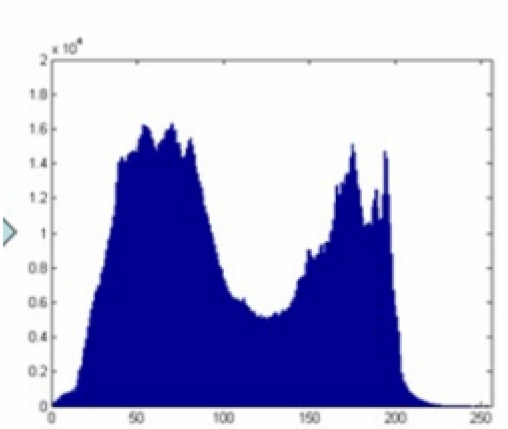
$$S_2 = 8-1\sum_{j=0}^{2} P(r_j) = 7\left(P(r_0)+P(r_1)+P(r_2)\right) =$$
$$= 4{,}55$$

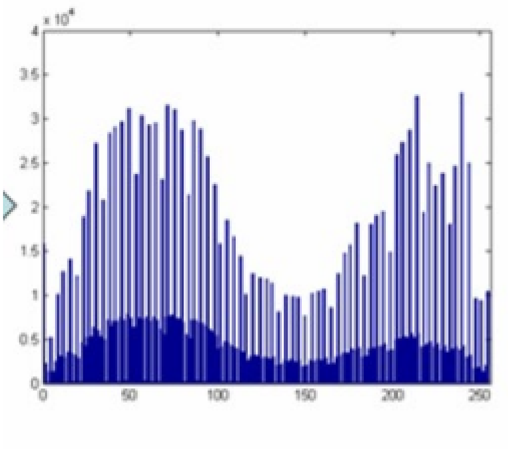
$\vdots$

$r_7 \Rightarrow s_7$, $r_7 = 7$, $s_7 = 7$

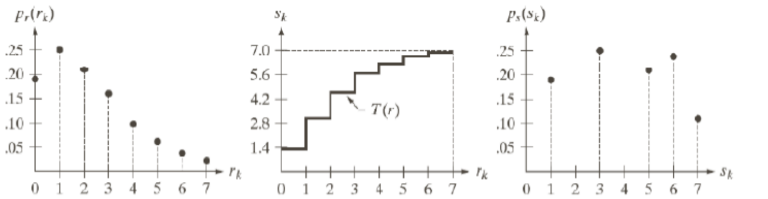
L'ultimo livello rimane tale e quale

> I valori che si ottengono vengono arrotondati all'intero più vicino.
> Si nota che gli ultimi 3 che prima erano diversi si trasformano in un unico valore.
> In output non ci sono più i pixel di valore 2 e nemmeno 4 ed è per questo motivo si notano dei buchi nell'istogramma, come le seguenti immagini:



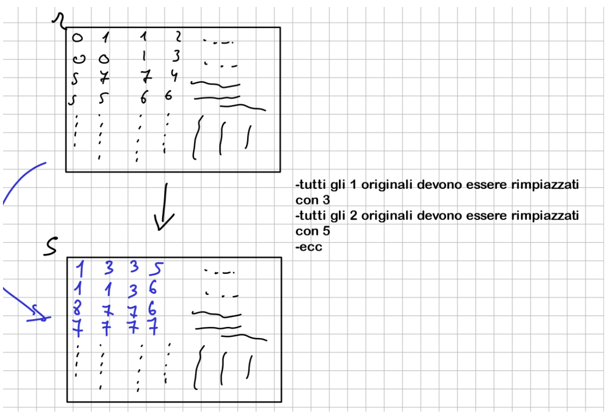


Il risultato finale dell'algoritmo è:



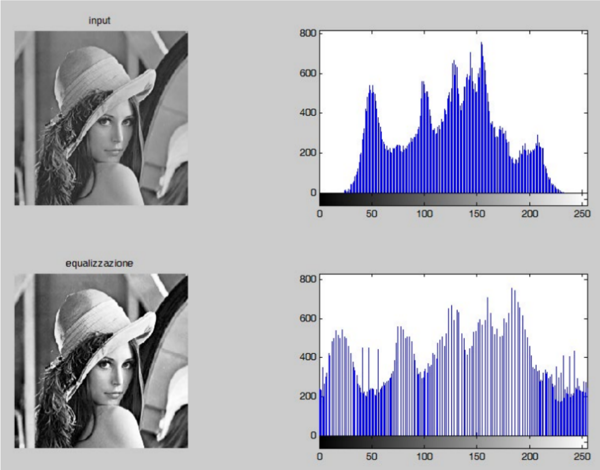
a b c

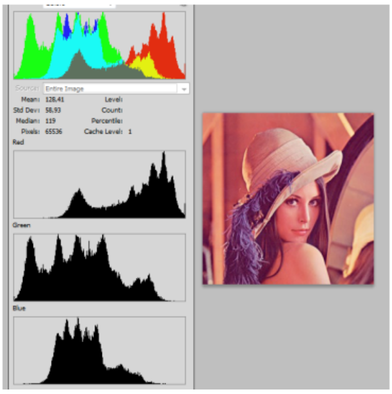
FIGURE 3.19 Illustration of histogram equalization of a 3-bit (8 intensity levels) image. (a) Original histogram. (b) Transformation function. (c) Equalized histogram.

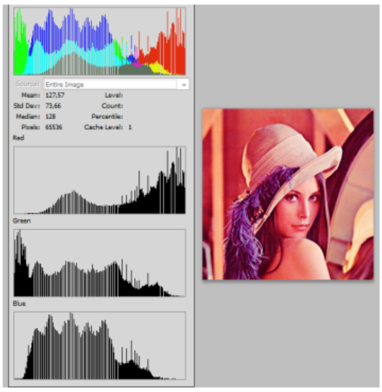


- **Si forza la probabilità iniziale** e la si fa diventare equiprobabile ogni livello, cioè ognuno di essi darà, più o meno, lo stesso contributo;
- Per tutti gli $(r_0 \dots r_k)$ si devono calcolare tutti gli $(s_0 \dots s_k)$
- Se voglio applicare l'algoritmo su un'immagine a colori (RGB), applico lo stesso algoritmo su ogni canale separatamente come se ogni canale fosse a scala di grigi.

- Il risultato dell'algoritmo **potrebbe essere sgradevole**. Esso dipende da come è fatta l'immagine iniziale e dall'osservatore stesso.







OPERAZIONI SULLE IMMAGINI

In base al tipo di dipendenza del risultato si distinguono 2 tipi di operazioni:

- se il risultato **dipende da UN PIXEL** si tratta di **OPERATORE PUNTUALE**;
- se il risultato **dipende da PIU' PIXEL**, si tratta di **OPERATORE LOCALE**.

> Per semplificare la trattazione del problema lavoreremo solo su immagini a toni di grigio visto che i 3 canali RGB si possono trattare allo stesso modo separatamente.

- [ ] Chiaramente, se si cambia spazio di colore, allora le operazioni potrebbero non avere senso, infatti **queste operazioni si riferiscono allo spazio di colore RGB**.
- [ ] A differenza degli operatori affini dove si calcolavano delle coordinate, qui **si modificano i valori relativi ai livelli di intensità di grigio**.

Un operatore generico che agisce sui livelli di grigio è rappresentato cosi:
$g(x, y) = T[f(x, y))]$ dove:

- f è **l'immagine di ingresso** all'elaborazione;
- g è **l'immagine d'uscita**;
- T è **un operatore** su f definito in un intorno di (x, y).

La **dimensione dell'intorno** di (x, y) definisce **il carattere** dell'elaborazione:

- **PUNTUALE** (*l'intorno coincide con il pixel stesso*);
- **LOCALE** (*una piccola regione quadrata centrata sul pixel*);
- **GLOBALE** (*l'intorno coincide con l'intera f*)

OPERATORI PUNTUALI

Si dice **OPERATORE PUNTUALE** un operatore che preso in input il valore di un pixel ne restituisce un altro pixel **variato** che dipende esclusivamente dal valore del pixel in ingresso.

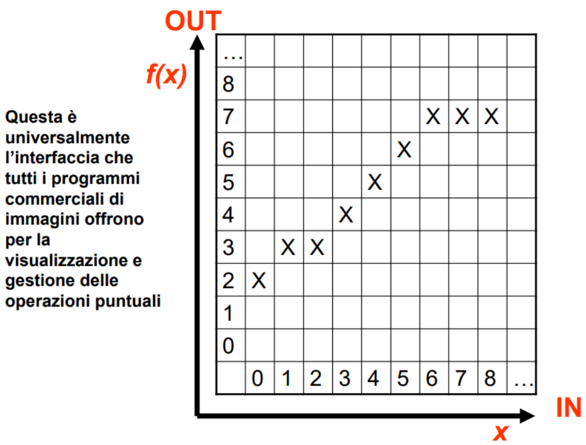
Tipiche operazioni puntuali sono:

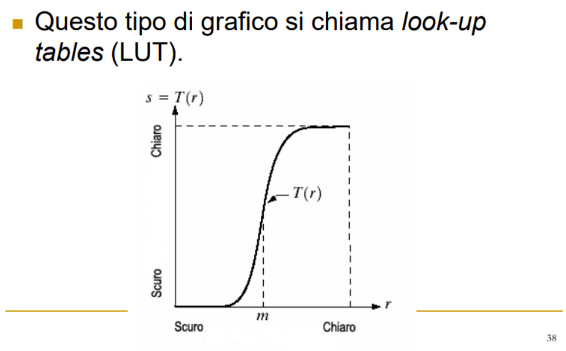
- **Aumentare/ diminuire la luminosità**. Si prende un pixel e gli si somma/sottrae una costante;
- Inversione della scala dei grigi (operazione **negativo**);
- espansione del contrasto;
- modifica (**equalizzazione** o specifica) dell'istogramma;
- presentazione in **falsi colori**. (immagine cosi come viene fuori se ho usato il Bayer Pattern, dove per ogni pixel ho la terna RGB con alcuni zeri nel mezzo)

Visto che **l'output dipende solo da un valore in ingresso**, si può costruire una **TABELLA** del tipo:

IN	0	1	2	3	4	5	6	7	...
OUT	T(0)	T(1)	T(2)	T(3)	T(4)	T(5)	T(6)	T(7)	...

Per esempio:





> Nell'immagine soprastante: questa curva aumenta il contrasto.

Negativo

L'operatore negativo è la più semplice operazione puntuale. Infatti essa consiste nell'associare al valore $f(x, y)$ del pixel il valore $255 - f(x, y)$

DATI

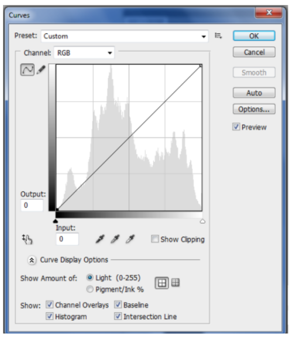
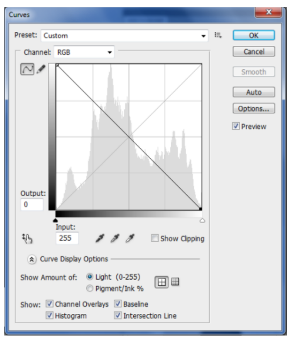
- $f(x,y)$ Valore di INPUT del pixel in posizione (x,y)
- $g(x,y)$ Valore di OUTPUT del pixel in posizione (x,y)

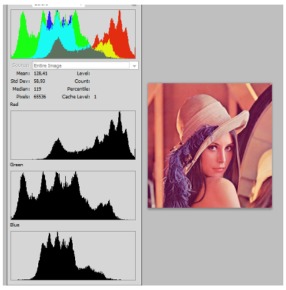
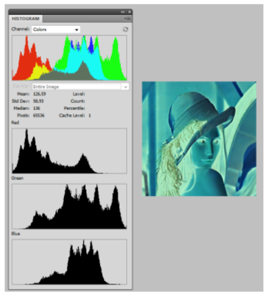
ALLORA

$$\boxed{f(x,y) \xrightarrow{\text{NEGATIVO}} g(x,y)}$$

$$g(x,y) = 255 - f(x,y)$$

"255" può variare se varia il numero di bit
che si stanno usando in quel momento

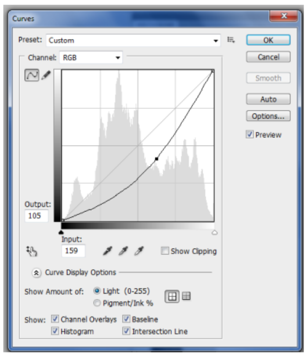
- Dopo aver applicato l'operatore NEGATIVO, la curva nella **LUT**, **in caso di un solo canale**, diventa:
    -  
- Oppure, **in caso di RGB**, diventa:

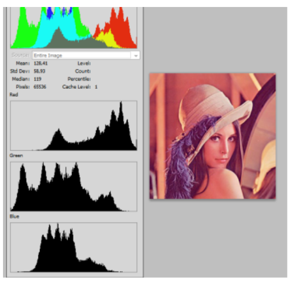
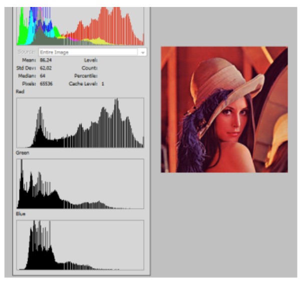
- Gli istogrammi e le curve **appaiono SPECCHIATI**.

Incupimento

- In caso di operatore puntuale incupimento si deve modificare la curva della **LUT** nel seguente modo:

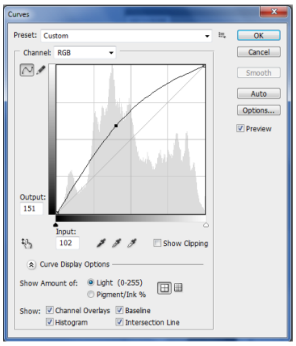


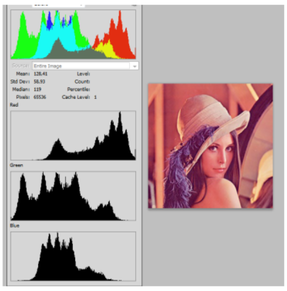
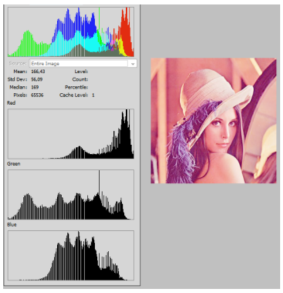
- La curva si trova **sotto la bisettrice**;
- Si tratta di una **sottrazione non lineare** e questo modifica l'**esposizione** (in questo caso la abbassa);
- In caso di **immagine RGB** ci si comporta allo **stesso modo per ogni canale**. Si ha, per esempio:

- Si nota che **l'istogramma si popola maggiormente a sinistra**.

Schiarimento

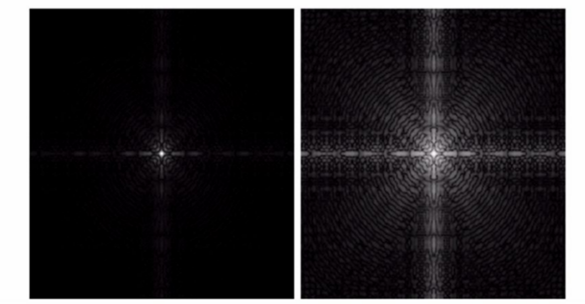
- Lo schiarimento è il contrario dell'incupimento, pertanto avrò la curva nella **LUT**:
  - 


- La curva si trova **sopra la bisettrice**;
- In caso di **immagine RGB** avrò l'applicazione dell'operatore per **ogni singolo canale**:
  -  
- Si nota che **l'istogramma è più popolato nella parte destra**.

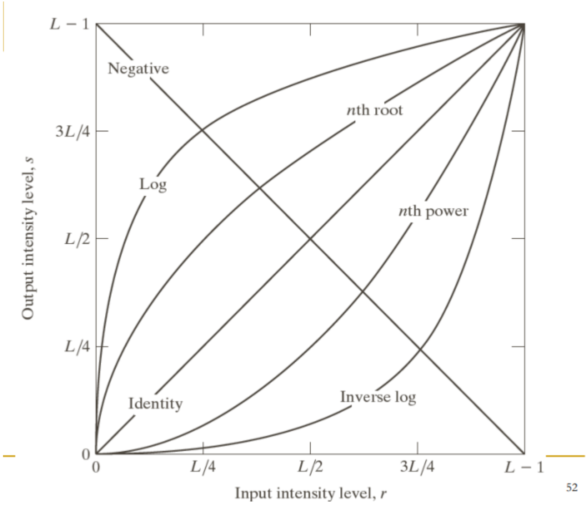
Trasformazione logaritmica

Una trasformazione logaritmica è così fatta: $g(x,y) = c\,log(1 + f(x,y))$ dove:

- c è una costante positiva che serve solo per normalizzare il risultato nel range [0,255] perchè, dopo l'applicazione del logaritmo, il range varia.
- il numero 1 serve per evitare di avere l'argomento del logaritmo uguale a zero visto che f(x,y) può anche essere 0 in caso di pixel completamente nero
  - $f(x,y) \in [0, 255]$
    $\log(1+f(x,y)) \in [0, 8]$
    $255 \cdot \frac{\log(1+f(x,y))}{\log 1+255} \in [0, \cancel{1}\ 255]$
    c
    $\boxed{c = \frac{255}{\log 1+255}}$
- Questa trasformazione è utile per osservare meglio un certo contenuto, quindi **serve per schiarire**:



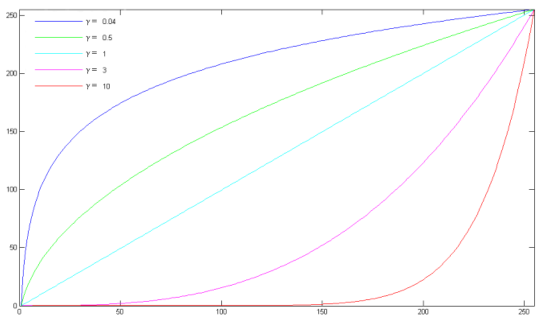
Trasformazioni raggruppate



Trasformazione di potenza

La trasformazione di potenza è così fatta:
$g(x,y) = c(f(x,y))^{\gamma}$ dove:

- se $\gamma > 1$ allora si ha un **incupimento**;
- se $\gamma < 1$ allora si ha uno **schiarimento**.
  La LUT della trasformazione di potenza varia nel seguente modo:



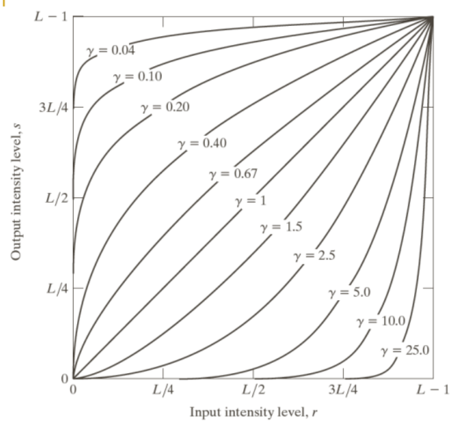


FIGURE 3.6 Plots of the equation $s = cr^{\gamma}$ for various values of γ ($c = 1$ in all cases). All curves were scaled to fit in the range shown.

Originale

Gamma=3

Gamma=10

Gamma=17

Esempio:

$$g(x,y) = c\,\left(f(x,y)\right)^{\gamma}$$

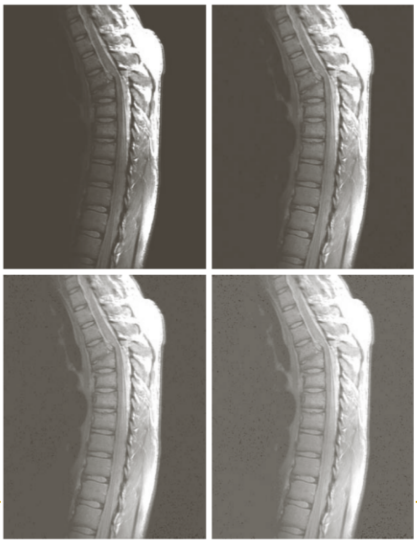
la funzione gamma è monotona crescente quindi corrispondono i minimi

$$f(x,y) \in [0, 255]$$

$$f(x,y)^{\gamma} \in [0, 255^{\gamma}]$$

$$\frac{f(x,y)^{\gamma}}{255^{\gamma}} \in [0, 1] \longrightarrow 255 \cdot \frac{f(x,y)^{\gamma}}{255^{\gamma}} \in [0, 255]$$

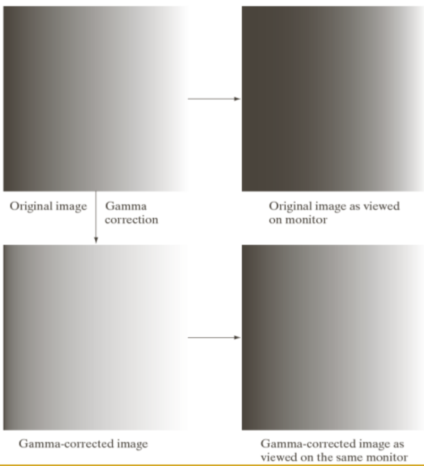
$$c = \frac{255}{255^{\gamma}} = \frac{1}{255^{\gamma-1}}$$



a b
c d

FIGURE 3.8 (a) Magnetic resonance image (MRI) of a fractured human spine. (b)–(d) Results of applying the transformation in Eq. (3.2-3) with $c = 1$ and $\gamma = 0.6, 0.4,$ and 0.3, respectively. (Original image courtesy of Dr. David R. Pickens, Department of Radiology and Radiological Sciences, Vanderbilt University Medical Center.)

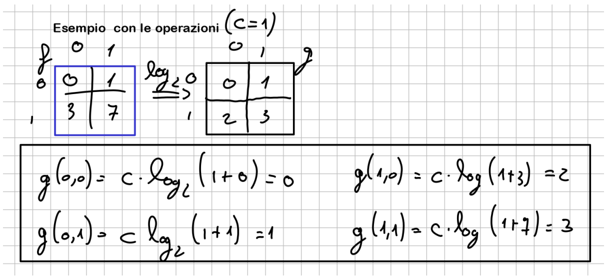
- Su un monitor CRT (con $\gamma = 2.5$ si può applicare una correzione pre-processando l'input con la corrispondente funzione inversa: $g(x,y) = f(x,y)^{\frac{1}{2.5}} = f(x,y)^{0.4}$):

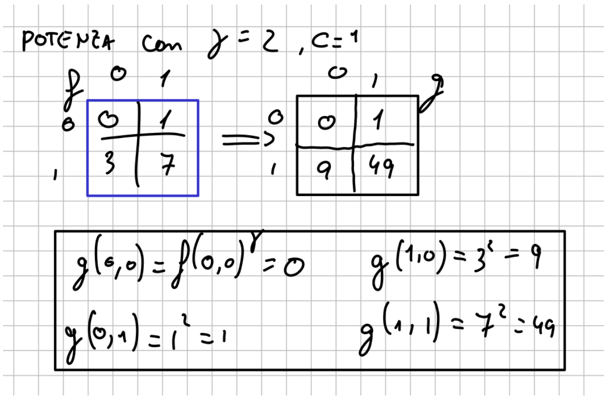


a b
c d

FIGURE 3.7 (a) Intensity ramp image. (b) Image as viewed on a simulated monitor with a gamma of 2.5. (c) Gamma-corrected image. (d) Corrected image as viewed on the same monitor. Compare (d) and (a).

Esempio con i numeri:





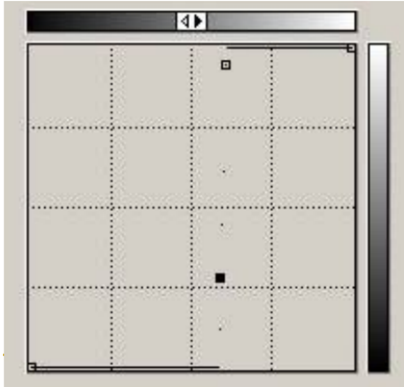
Binarizzazione

In questa operazione puntuale si passa da una situazione dove si hanno più colori ad una situazione dove si hanno solo 2 colori.

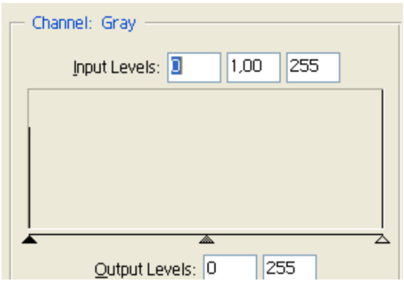
- **si sceglie un valore soglia** e tutto cioè che è **sotto la soglia diventa** 0, mentre tutto ciò che è **sopra la soglia diventa** 255.

> In altre parole: la binarizzazione si ottiene scegliendo una soglia T e mettendo a **nero** tutti i pixel il cui valore è **minore a** T e a **bianco tutti gli altri.**

La LUT apparirà nel seguente modo:

- 
- Vi saranno presenti solo 2 barre: 1 a sinistra e 1 a destra in corrispondenza del valore minimo e del valore massimo.

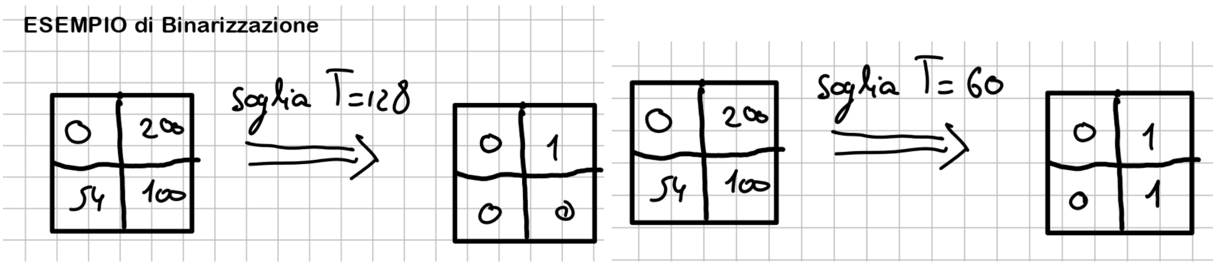
L'istogramma apparirà nel seguente modo:

- 

- In caso di immagine RGB si ha l'immagine a 8 colori quindi 2 colori per ogni canale. Di conseguenza l'immagine RGB non è binarizzata.

- Per binarizzare l'immagine RGB si passa a scala di grigi con la somma pesata con il valore che ha ogni canale) e poi si binrarizza normalmente

Esempio di binarizzazione con i numeri:

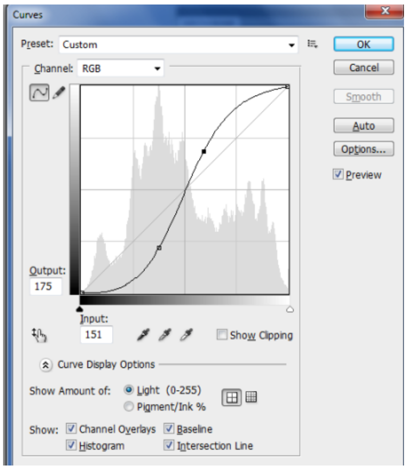


Aumento del contrasto

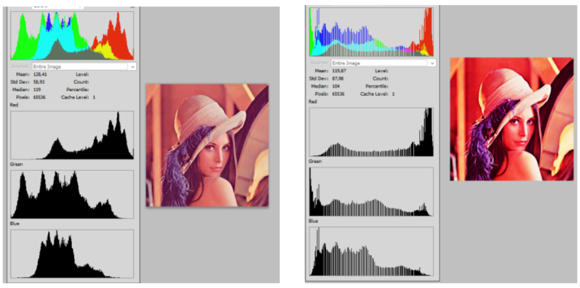
Aumentare il contrasto vuol dire **rendere più evidenti le differenze di colore** quindi *si rendono più chiari i colori chiari e più scuri i colori scuri*.

La curva della **LUT** varia nel seguente modo:

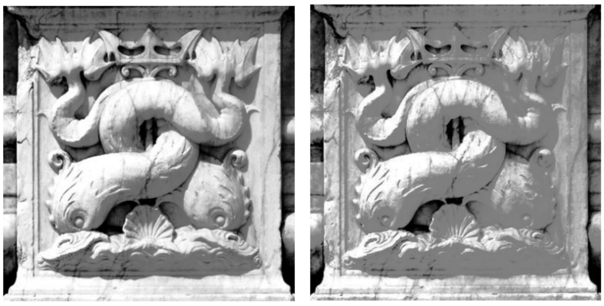
- 

In caso di immagine **RGB** la situazione si presenta come nel seguente modo:

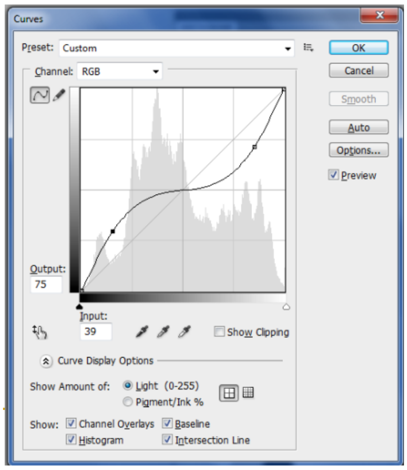
- 

Diminuzione del contrasto

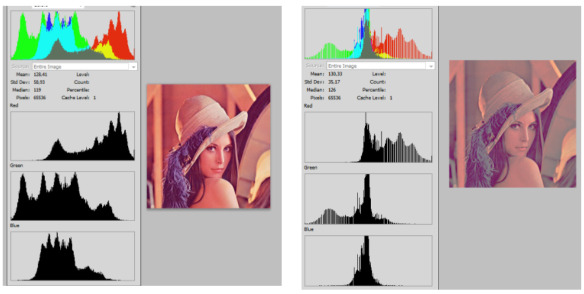
Per la diminuzione del contrasto si applica il ragionamento opposto rispetto all'aumento del contrasto. Si rendono **meno evidenti le differenze di colore** quindi rispetto all'operazione soprastante, quindi *si rendono più chiari i colori scuri e più scuri i colori chiari*.



Nella **LUT** si avranno le seguenti variazioni:

- 

In caso di immagine **RGB** si hanno le seguenti variazioni:

- 
- Si nota che nell'istogramma vi è una popolazione maggiore verso il centro di esso.

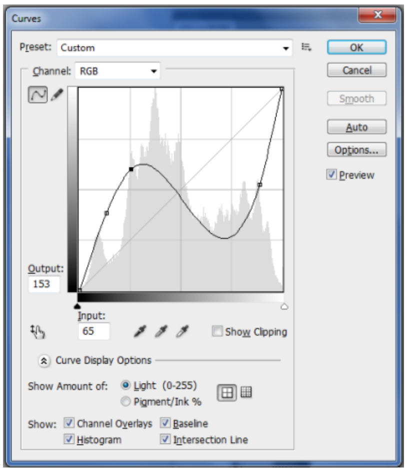
Solarizzazione (curve non monotone)

E' possibile fare delle variazioni alle **curve** della **LUT** in modo che questa **diventi non monotona**, come nella **solarizzazione**.

In altre parole: *colori chiari possono diventare più chiari o più scuri e superata una certa soglia si inverte il processo*

Nella **LUT** si avranno le seguenti variazioni:

- 

In caso di immagine **RGB** le curve variano nel seguente modo:

-