

Intensità percepita

- L'intensità percepita dipende da persona a persona. *La luce che viene vista* è diversa rispetto alla *luce che effettivamente percepisce l'occhio*. Infatti, l'intensità che percepiamo è in funzione logaritmica di quella che vediamo.
- **Più luce** c'è e **meno si diventa sensibili** a variazioni di colore o intensità quindi ne serve molta di più per percepire la variazione.

In caso di immagini digitali ci sono infiniti colori.

Range dinamico

Esempio:



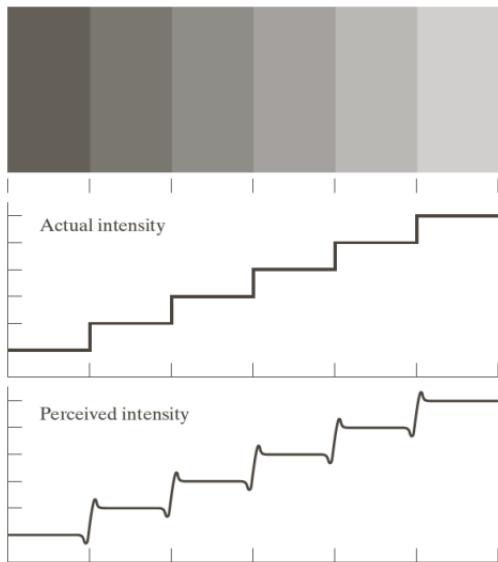
Tendiamo a concentrarci su una **parte specifica** della scena e l'occhio tende a *stabilizzarsi* solo su un'unica parte piuttosto che un'altra.

In pratica succede che:

- dove c'è poca luce la pupilla si dilata per catturare più luce e viceversa dove c'è molta luce;
- **l'occhio lascia in ombra le zone non a fuoco** permettendo una concentrazione maggiore su una zona a fuoco;
- quindi o ci si concentra su un punto con molta luce oppure su un punto con poca luce. Ci si focalizza solo un tipo di luce.

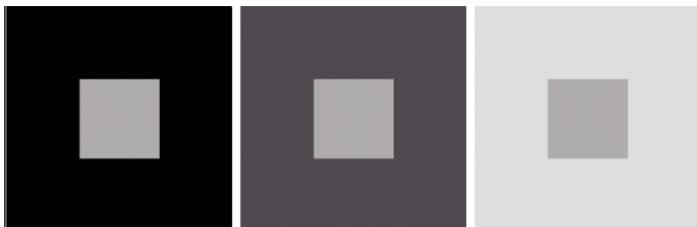
Illusioni ottiche

Le bande di Mach



- La **banda a destra è più chiara e quella a sinistra è più scura**.
- La percezione è quella mostrata nel grafico "*Perceived intensity*", dove essa è diversa rispetto alla reale transizione di colori fra i grigi ("*Actual Intensity*").
- Tendenzialmente, se si guardano le figure da sinistra verso destra, si vede il **rettangolo seguente un po' più scuro** di quanto non lo sia veramente.

Contrasto simultaneo



- I **quadrati centrali** sono dello **stesso colore**.
- Il **quadrato centrale più a destra sembra più scuro** all'impatto e dipende dallo sfondo.
- I colori uguali dei quadrati centrali sembrano completamente diversi perché l'occhio valuta anche il contrasto dello sfondo.

Rappresentazione di un'immagine

L'immagine è **bidimensionale non solida** quindi si può usare il formalismo matematico a 2 dimensioni. Un'immagine si può considerare come una funzione $f(x, y)$ che ha un certo valore in corrispondenza di x e y che rappresentano le coordinate spaziali.

Si distinguono 2 aspetti:

1. $i(x, y)$, che rappresenta **LUCE INCIDENTE**, cioè quella che arriva dall'esterno e colpisce l'oggetto;

2. $r(x, y)$, che rappresenta la **LUCE RIFLESSA** cioè la luce che l'oggetto riflette (dipende quindi dall'oggetto e le sue caratteristiche chimiche) che non è la stessa di quella incidente visto che una parte di luce viene assorbita dall'oggetto.

Quindi, generalmente, l'immagine può essere descritta $f(x, y) = i(x, y) \cdot r(x, y)$

- i può tendere ad un valore estremamente grande (non si considerano limiti), quindi $0 < i(x, y) < \infty$
- r può avere un valore in percentuale della luce riflessa, quindi $0 < r(x, y) < 1$. Quando è 1, dipende solo dalla luce incidente perchè tutta la luce incidente viene riflessa. Quando è 0 allora l'oggetto lo vedo nero e la luce riflessa è pari a 0.

- Gli oggetti che appaiono più scuri perchè hanno coefficiente di riflessione più vicini allo zero e viceversa;
- Un oggetto che assorbe completamente la luce non si può ottenere mai in natura (coefficiente di riflessione 0) (per esempio: "il nero più nero del mondo");
- L'unità di misura dell'intensità luminosa è il **lumen**.



Alcuni valori tipici di illuminazione e riflettanza

- Il Sole può produrre più di 90000 lm/m^2 ; produce fino a 10000 lm/m^2 quando è nuvoloso;
- La Luna piena raggiunge 0.1 lm/m^2 (ricordando però che la Luna non produce luce propria ma diremo "teoricamente" che brilla di luce propria).

Discretizzazione

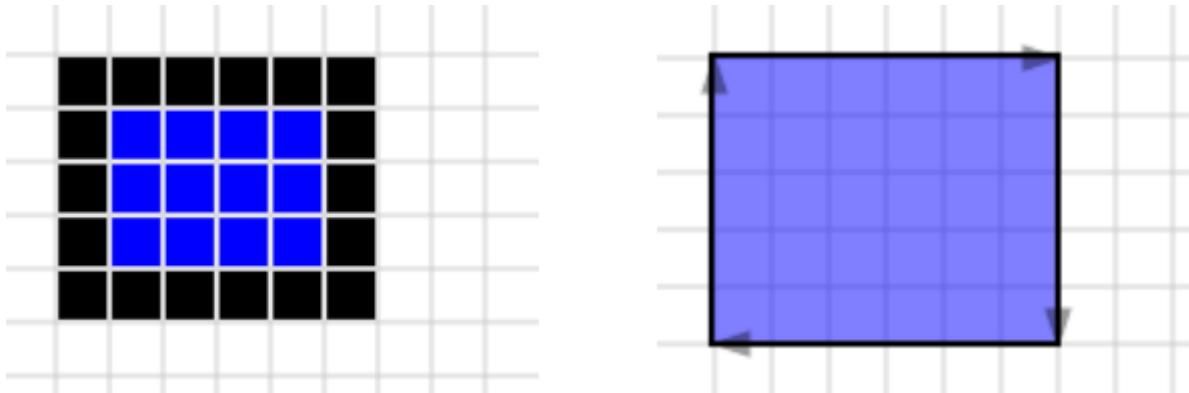
In teoria il valore di $f(x, y)$ è un numero reale, pertanto ha infiniti valori. Ma per produrre/elaborare una immagine digitale abbiamo bisogno di valori discreti visto che la memoria è limitata e il passaggio dal continuo al discreto si chiama campionamento e quantizzazione.

Quindi bisogna ridurre numero di elementi da osservare di questa funzione $f(x, y)$ e decidere fra tutti i possibili valori di $f(x, y)$ se posso considerarli tutti, e ovviamente no.

Quindi quanti valori di luminosità posso considerare validi?

- Se uso **8 bit** (scala di grigi) riesco a rappresentare molto bene un **contenuto fotorealistico**, perchè è sufficiente.
- Il piano **XY** in cui stanno le coordinate dell'immagine è detto **DOMINIO SPAZIALE** e le variabili x, y sono dette **variabili spaziali** o **coordinate spaziali**.

IMMAGINI RASTER E IMMAGINI VETTORIALI

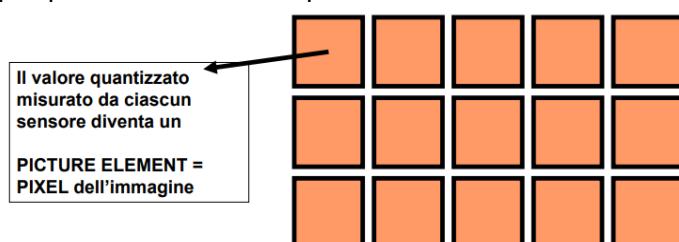


A sinistra c'è l'**immagine raster**. A destra c'è l'**immagine vettoriale**.

Immagine Raster

Per l'immagine raster uso **la matrice** che è definita da un numero finito di elementi $n \times n$.

- Fissata una coordinata posso accedere $[i, j]$, leggere, individuarne un particolare valore;
- Quindi ho un numero finito di celle che possono assumere un valore finito di valori.
- Non si ha una descrizione semantica dei contenuti.
- La singola cella di questa matrice viene detta **Piture-Element**, chiamata **PIXEL** ed essa è la più piccola unità che si può considerare nella matrice.



- Questo implica che nella descrizione dell'immagine ho una limitazione del numero di pixel che si ha a disposizione. Il grado di dettaglio dipende, quindi, dal numero di pixel disponibili.
- Il numero di pixel disponibili viene chiamato **RISOLUZIONE**.

Esempio di Raster: Paint.

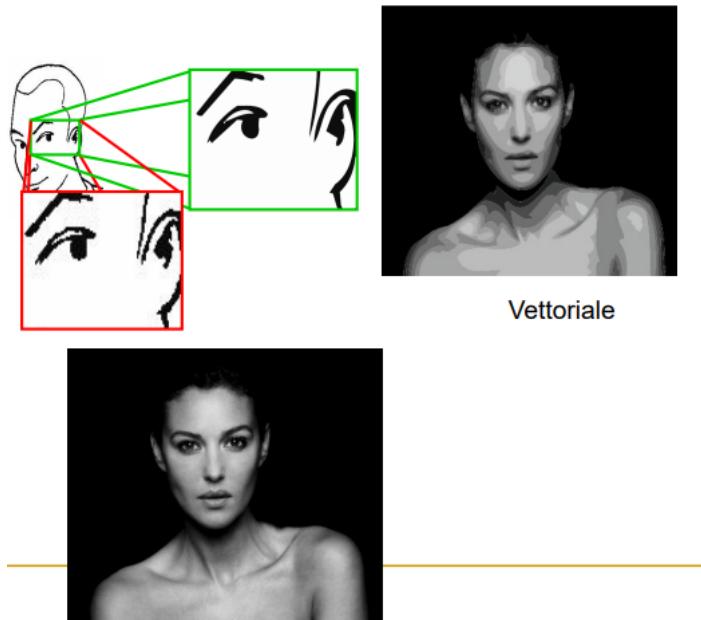
Immagine Vettoriale

- L'immagine vettoriale si descrive attraverso una lista **definendo** singolarmente **tutte le primitive geometriche** presenti nell'immagine.(bordo, centro, altezza,colore della figura, colore del bordo, ...)
- Quindi esiste una maniera precisa che serve per descrivere le figure. Pertanto le primitive sono "*fissate*".

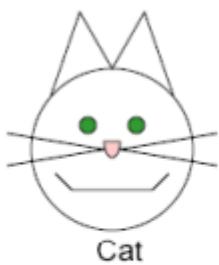
Vantaggi e svantaggi

- Un'**immagine fotorealistica**, rappresentata con immagini vettoriali, avrà primitive geometriche molto elevate e quindi verranno utilizzate **grandissime quantità di memoria**. Pertanto si utilizzano le immagini Raster;
- Uno **schizzo e/o un disegno tecnico** viene rappresentato con un'immagine vettoriale che rappresenta linee più nitide per ogni area dell'immagine.

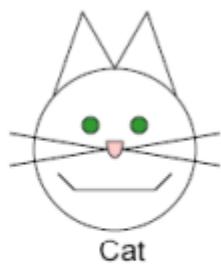
Confronto fra foto fotorealistiche



Confronto fra disegni più geometrici

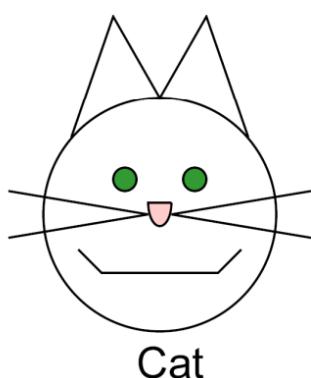
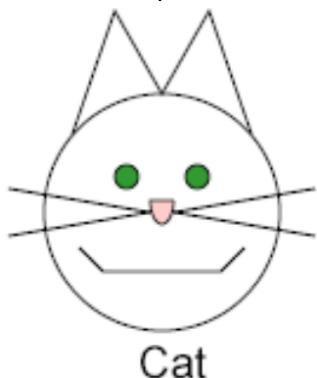


Raster
(140x170)



Vector
(140x170)

Zoomando, però, si vedono i difetti di Raster:



Vettorizzazione



	PSNR	bpp
Q=1	19,330	0,976
Q=2	20,178	1,343
Q=4	21,016	1,753
Q=8	21,975	2,251
Q=16	22,670	3,097
Q=32	23,461	4,064
Q=64	24,326	5,693
Q=128	25,180	7,660
Q=256	26,058	10,430
Q=512	26,852	14,169
Q=1024	27,596	19,281
Q=2048	28,174	26,703
Q=4096	28,768	37,641
Q=8192	29,320	53,429

bpp = bit per pixel

- Per l'immagine vettoriale soprastante servono 54 bit per descrivere l'immagine in formato vettoriale. 54 megabit
- Per l'immagine Raster soprastante ci vorrebbero solo 24 bit ottenendo un risultato ancora migliore.

Non conviene usare immagine vettoriale per rappresentare questi contenuti (fotorealistici).

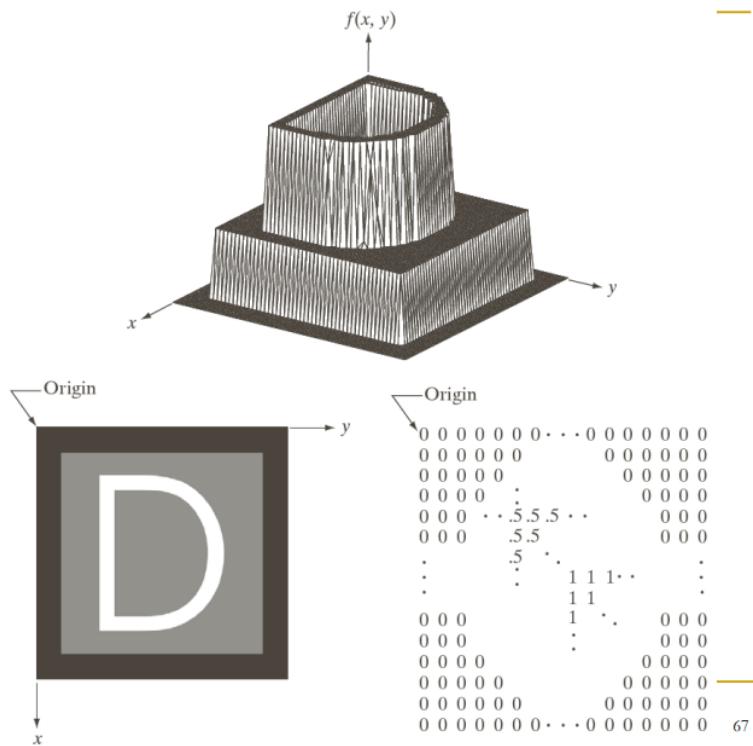
Rasterizzare

Rasterizzare vuol dire passare da un'immagine vettoriale ad una matrice in raster, così perdendo descrizione, non c'è più concetto di descrizione a livello geometrico.

Formati grafici vettoriali:

- **PS** (*Postscript*): Formato sviluppato da AdobeSystems originariamente per la stampa di documenti su stampanti laser, è utilizzato anche per la memorizzazione di immagini vettoriali.
- **EPS** (*Encapsulated Postscript*): Estensione del formato PostScript che consente di incapsulare immagini bitmap (raster).
- **DCS** (*Desktop Color Separation*): Un caso speciale di EPS sviluppato originariamente da Quark per tenere separati i dati al alta risoluzione dall'anteprima a bassa risoluzione.
- **PDF** (*Portable Data Format*): Sviluppato da Adobe, è il formato più diffuso per condividere, indipendentemente dalla piattaforma, documenti di testi e immagini. PDF usa immagini raster ma sa gestire immagini vettoriali mantenendo, dopo l'esportazione, le immagini vettoriali che sono state inserite. Sa ricalcolare le primitive.
- **PICT**: Formato grafico sviluppato da Apple Computer per la piattaforma Macintosh in grado di memorizzare sia immagini vettoriali che raster.

PIXEL



i 3 puntini servono per dire che la figura è più grande rispetto a quella rappresentata.

La matrice a sinistra rappresenta una porzione dell'immagine di Lena e i vari numeri rappresentano proprio $f(x,y)$.

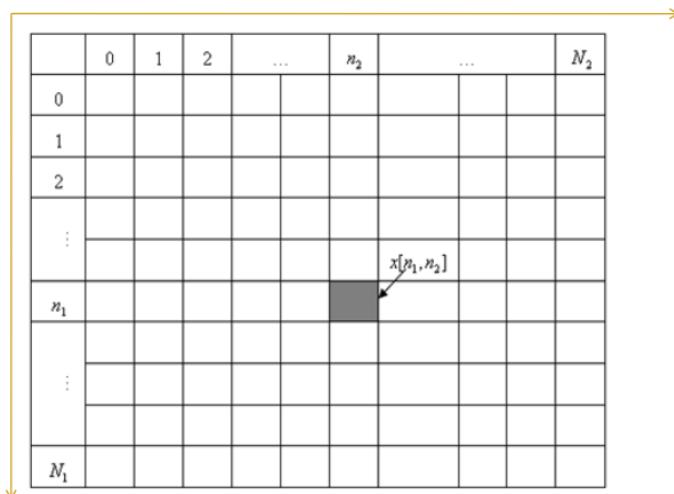
- Ogni numero presente nel pixel è associato ad una luminosità/colore;
- con **8 bit**, i valori dei pixel variano da **0-255**;
- colori scuri = valore del pixel basso
- colori chiari = valore del pixel alto

Nella seguente immagine vi è l'immagine di Lena e, accanto, la tabella che rappresenta i valori dei pixel di una porzione d'immagine.

File	Edit	Debug	Desktop	Window	Help
103	94	70	121	207	198
67	77	117	186	195	191
117	135	171	193	188	190
129	142	155	154	170	163
91	106	120	115	120	117
82	94	111	106	95	99
109	113	121	117	121	127
117	123	126	125	133	140
138	141	140	141	146	149
146	147	145	151	146	145
143	144	142	143	139	142
144	145	143	144	143	145
147	148	146	145	146	147
149	149	145	144	148	147
148	147	142	141	145	143
149	148	143	141	142	140
148	147	140	138	138	136
145	145	139	137	138	136
140	138	137	137	135	132
137	137	137	138	134	132

Convenzioni

- l'**origine** si trova nel punto (0,0) ed è in **alto a sinistra**;
- si considera sempre il **4 quadrante del piano cartesiano** ma i **valori** sono tutti **positivi**;
- l'**asse a sinistra** è ordinato **dall'alto verso il basso**; l'**asse in alto** è ordinato **da sinistra verso destra**.



Se dovessi individuare un pixel scriverei $x[n_1, n_2]$ dove n_1 è la coordinata che si riferisce all'asse x ; n_2 si riferisce all'asse y .

TIPOLOGIE DI IMMAGINI RASTER

Bianco/nero

- Serve solo **1 bit** per pixel;
- nella posizione (i, j) ci sarà 0 oppure 1.



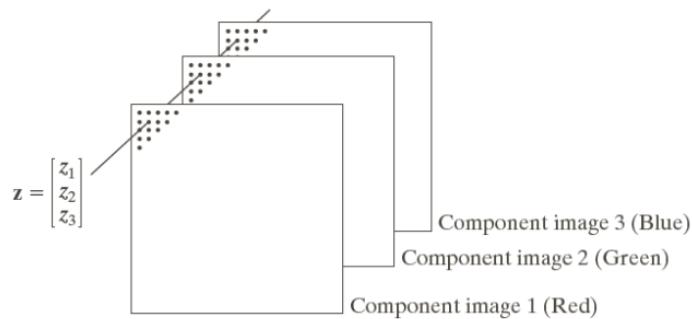
Scala di grigi

- Servono **8 bit** per pixel;
- Nella posizione (i, j) ci sarà un valore compreso fra $[0, 255]$.



A colori

- **8 bit per canale.** Poiché i canali sono 3 avrò **24 bit**;
- Nella posizione (i, j) ci sarà una terna del tipo (x, y, z) con x, y, z che assumono valori compresi tra $[0, 255]$.



In questo caso si descrive anche la lunghezza d'onda combinando le 3 lunghezze d'onda fondamentali, ovvero Red, Green e Blue (RGB). Quindi si rappresenta mediante 3 contributi di 3 lunghezze d'onda (il più usato).

Se si usassero i canali singoli (solo una lunghezza d'onda) vengono rappresentati come se fossero 3 immagini a scala di grigi:



OPERAZIONI SULLE IMMAGINI

Si tratta di un'operazione che prende una matrice e modifica i valori dei pixel secondo qualche regola (dipende dall'operazione da implementare e dal risultato al quale si tende ottenere).

Le operazioni principali sono:

- Somma di un valore negativo o positivo se voglio scurire o schiarire un'immagine quindi modificando i valori dei pixel.
- Il prodotto **puntuale** si fa fra 2 matrici che hanno esattamente le stesse dimensioni e corrisponde:

$$A = [a_{ij}]$$

$$B = [b_{ij}]$$

$$C = [c_{ij}] = [a_{ij} \cdot b_{ij}]$$

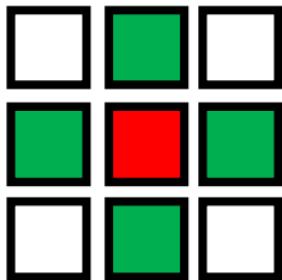
$$\begin{bmatrix} 1 & 2 \\ 3 & -1 \end{bmatrix} \cdot \begin{bmatrix} -3 & 0 \\ 1 & 4 \end{bmatrix} = \begin{bmatrix} -3 & 0 \\ 3 & -4 \end{bmatrix}$$

Un'operazione che non ha senso fare su un'immagine è il prodotto riga per colonna.

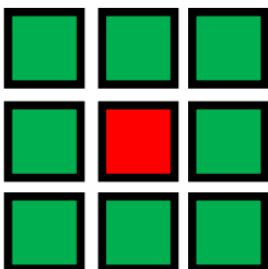
NEIGHBORHOOD

La parola *NEIGHBORHOOD* si abbrevia con N_p e tratta i **vicini** di un pixel. Essi possono essere di 2 tipi:

- Possono essere i **4 connessi** e che sono a destra/sinistra oppure sopra/sotto il pixel centrale
- In questo caso la distanza è 1;



- Possono essere **8 connessi** del pixel centrale, compresi quelli in diagonale.
- In questo caso la distanza fra il pixel centrale e uno ad esso diagonale è: $\sqrt{2} = \sqrt{1^2 + 1^2}$



OPERATORI AFFINI

Gli **operatori affini** (o trasformazioni affini) appartengono ad una classe di operatori che non agisce direttamente sui valori di intensità luminosa ma sulle coordinate.

In pratica si possono prendere dei pixel e spostarli, quindi vengono rimappati cambiandone le coordinate.