



Il colore

IGEA 2006-07

7 febbraio 2007



La luce

- Radiazione elettromagnetica

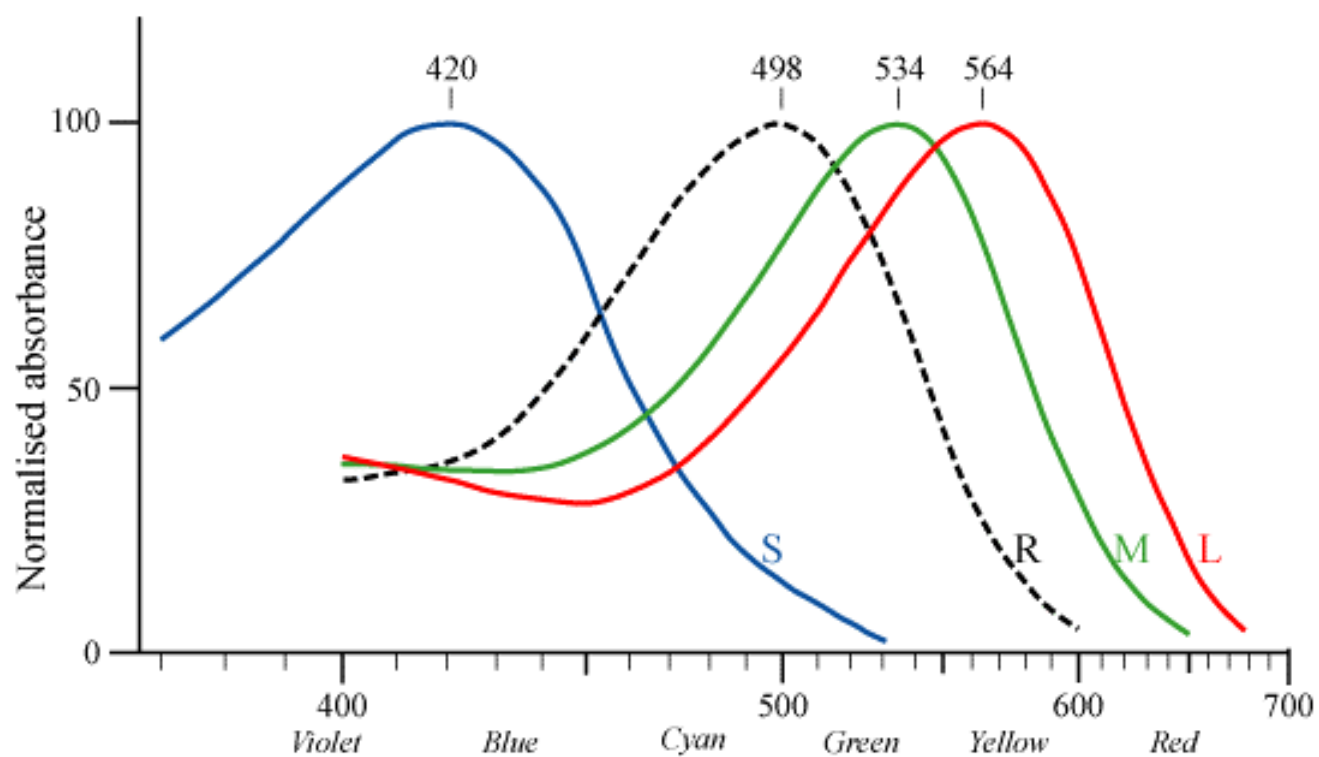


- 380 – 740 nanometri
- (790 – 480 THz)

Percezione della luce /1

- Organi sensoriali:
 - Bastoncelli
 - Molto sensibili (anche a un solo fotone: visione notturna!)
 - Sensibili all'intero spettro del visibile (luce monocromatica) (visione notturna è monocromatica)
 - Coni
 - Meno sensibili (100 volte meno dei bastoncelli)
 - Di tre tipi, ciascuno sensibile ad un intervallo di lunghezze d'onda limitato, centrato attorno alla luce
 - Rossa
 - Verde
 - Blu

Percezione della luce /2



Modelli di colore /1

- Vogliamo **rappresentare** il colore in modo matematico
- Cioè: associare ad ogni colore alcuni numeri
 - es. una terna di interi
- Obiettivo finale:
 - Trasmissione immagini
 - Memorizzazione digitale immagini
 - Ecc.

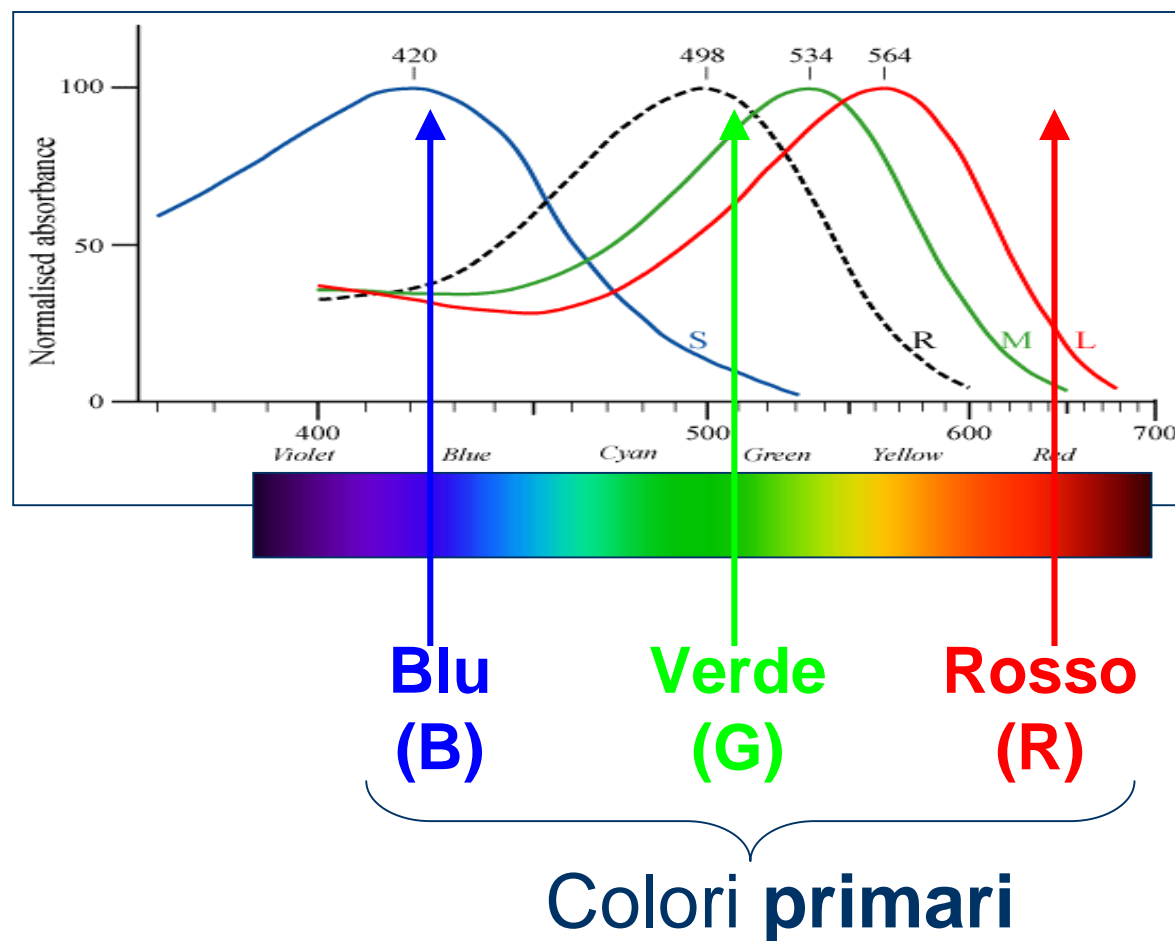
Modelli di colore /2

- Diversi modelli di colore:
 - Additivi (es. RGB)
 - Sottrattivi (es. CYMK)
 - Percettivi (es. HSV)
 - Cromaticità (usata in TV e JPEG)
 - (Tavolozze di colore)

Modelli additivi /1

- Generare la luce
- Stimolare i recettori dell'occhio, “simulando” la stimolazione che si avrebbe se guardassimo il colore che vogliamo rappresentare
- Stimoliamo i coni con i tre colori ai quali sono maggiormente sensibili:

Modelli additivi /2



Modelli additivi /3

- **RGB**: Ogni colore rappresentato tramite una terna di numeri reali in $[0, 1]$
- $(0, 0, 0)$: Nero
- $(1, 0, 0)$: Rosso
- $(0.5, 0.5, 0.5)$: Grigio
- $(1, 1, 1)$: Bianco

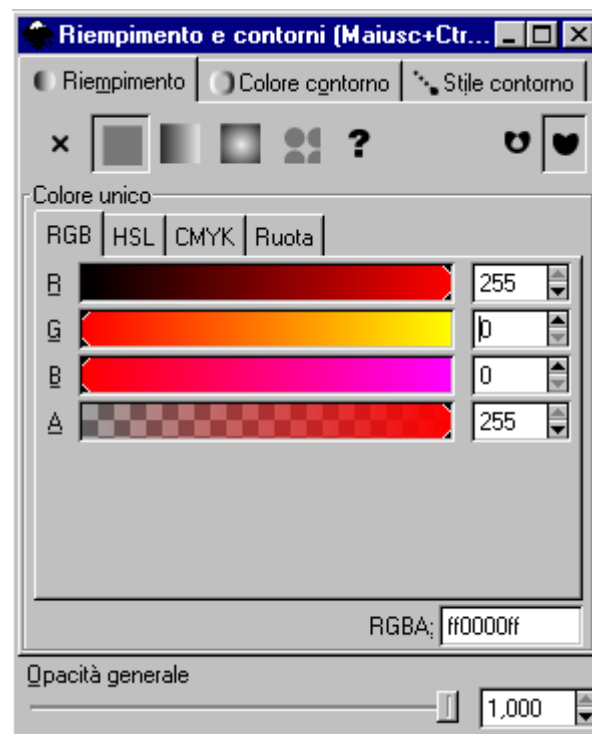
- Come rappresentare ciò nel computer?

Modelli additivi /4

- Usare numeri interi per ciascuna componente
- k bit $\rightarrow 2^k$ passi di discretizzazione
- TrueColor: 8 bit a componente (256 passi)
- (0, 0, 0): Nero
- (255, 255, 255): Bianco

- Complessivamente: $3 \cdot 8 = 24$ bit
- $2^{24} \approx 17$ milioni di colori rappresentabili
- L'occhio più raffinato ne distingue 10 milioni!

Modelli additivi /5



Modelli sottrattivi /1

- Le piante **odiano** la luce **verde!**
- Per questo non la assorbono
- La luce verde, riflessa dalle piante (e diffusa), raggiunge e stimola i recettori dell'occhio
- Le piante amano (e assorbono) le altre radiazioni luminose, che quindi non raggiungono il nostro occhio
- Oggetto **nero**: assorbe tutta la luce
- Oggetto **bianco**: riflette tutta la luce

Modelli sottrattivi /2

- Idea: sfruttare questo principio per produrre i colori su carta
- Usare dei pigmenti che assorbono solo un colore primario (RGB). Si ottengono così i colori primari **complementari**
 - Assorbo il **Rosso** → sono **Ciano** (= **Verde** + **Blu**)
 - Assorbo il **Verde** → sono **Magenta** (= **Rosso** + **Blu**)
 - Assorbo il **Blu** → sono **Giallo** (= **Rosso** + **Verde**)

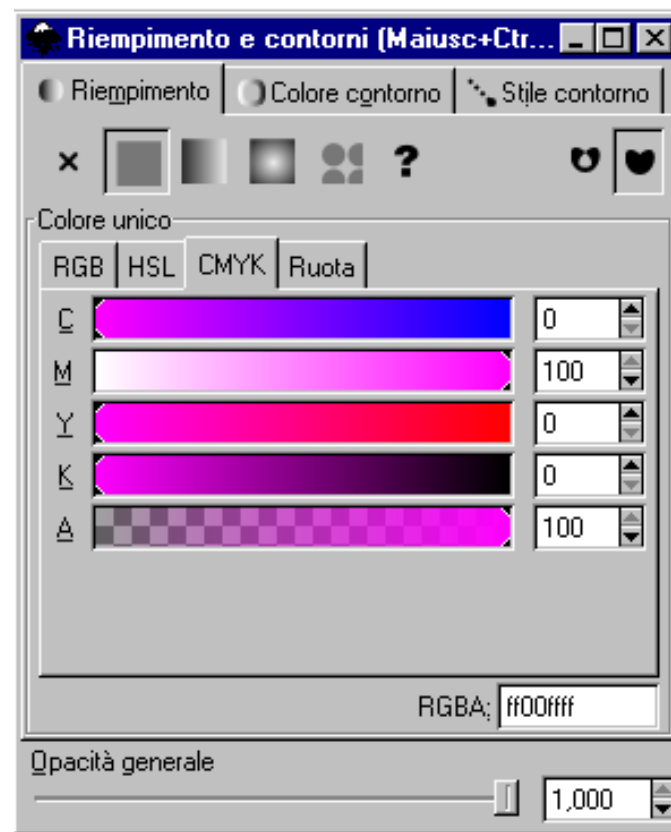
Modelli sottrattivi /3

- Mescolando colori primari complementari, si sottraggono ancora più componenti alla luce
- Esempio: mescolo **Ciano** e **Magenta**
 - il **Magenta** assorbe il **Verde**
 - il **Ciano** assorbe il **Rosso**
 - rimane soltanto il **Blu**: ottengo il **Blu**
- E' quello che fanno le stampanti a getto di inchiostro!

Modelli sottrattivi /4

- Quindi: ogni colore rappresentabile tramite tre componenti reali in $[0, 1]$, ossia
- le quantità dei tre colori primari complementari (**CMY**) che devo mescolare per ottenerlo
- posso aggiungere una componente nera (K) per aumentare la qualità della stampa
 - infatti il nero ottenuto semplicemente miscelando C, Y e K è spesso scadente
 - Modello **CMYK**

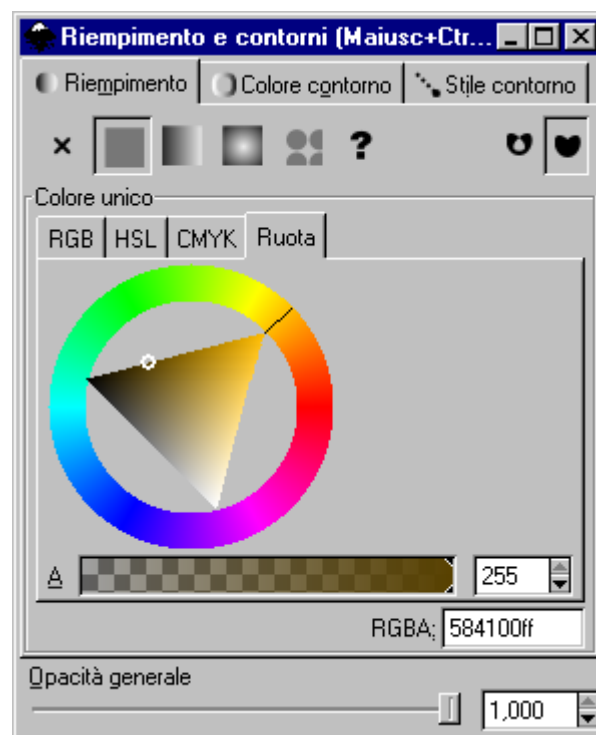
Modelli sottrattivi /5



Modelli percettivi /1

- Scopo: facilitare l'uso di programmi
- Colori disposti in modo intuitivo. Es **HSV**:
 - **Tinta (Hue)**: colore principale
 - **Saturazione**: grado di purezza del colore
 - Più saturo: solo il colore della tinta scelta
 - Meno saturo: altre componenti cromatiche presenti
 - **Valore**: presenza di luce nelle altre componenti cromatiche
 - Basso: altre componenti nere
 - Alto: altre componenti bianche

Modelli percettivi /2



La cromaticità /1

- Torniamo al modello additivo
- Se conosco la **luminanza** e due componenti cromatiche (normalizzate) (es. **verde** e **rosso**), posso ricavare la terza componente (es. **blu**)
- La TV in bianco e nero trasmetteva la **luminanza**
- Come *colorare* la TV? Aggiungendo 2 componenti cromatiche (normalizzate), la **cromaticità**

La cromaticità /2

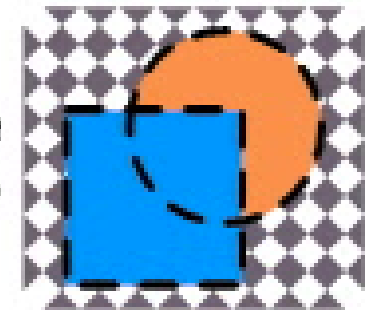
- L'occhio è più sensibile alla luminanza che alla cromaticità!
- Quindi posso risparmiare spazio rappresentando meno accuratamente la cromaticità
 - TV: poca “banda” dedicata alla cromaticità
 - JPEG: uso luminanza/cromaticità per rappresentare il file; sottocampiono la cromaticità

Il canale Alpha /1

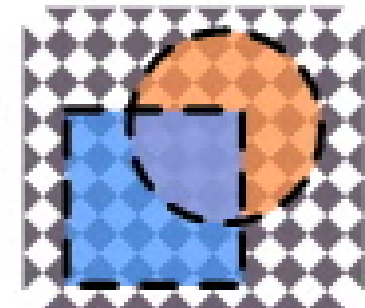
- A volte le immagini TrueColor usano 32 bit (invece di 24 bit). Perché?
- Aggiungono un'informazione sulla **trasparenza**
- **Non fa parte del modello di colore!**
- Dice come comportarsi quando si “sovrappongono” diverse immagini

A over B

Opaque
A and B



Partially
transparent
A and B

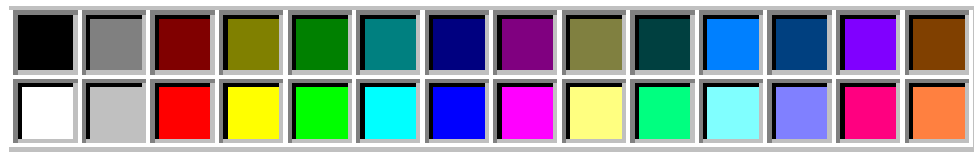


Il canale Alpha /2

- Canale Alpha: ulteriore reale in $[0, 1]$
- 0: oggetto completamente trasparente
 - invisibile!
- 1: oggetto opaco
 - copre completamente ciò che sta sotto
- TrueColor: trasparenza rappresentata con 8 bit
 - discretizzata in 256 livelli

Tavolozze di colore (palette)

- Usate per rappresentare in modo compatto immagini con pochi colori
- Definisco la tavolozza dei colori usati, esempio:



- Questa tavolozza ha 28 colori
- Sono sufficienti 5 bit per pixel per rappresentarne il colore ($2^5 = 32 > 28$)

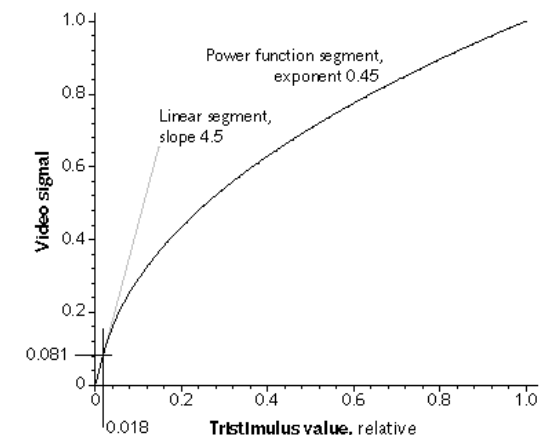
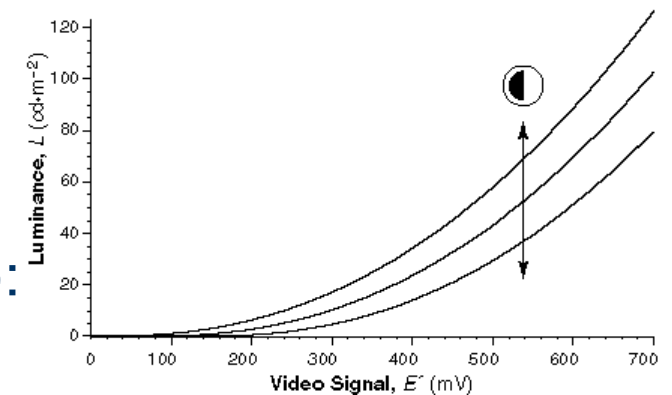
Correzione gamma /1

- Nei dispositivi di visualizzazione, l'intensità luminosa / prodotta **non è** proporzionale allo stimolo V_S (segnale di luminanza)
- Per esempio nella TV (e monitor CRT):

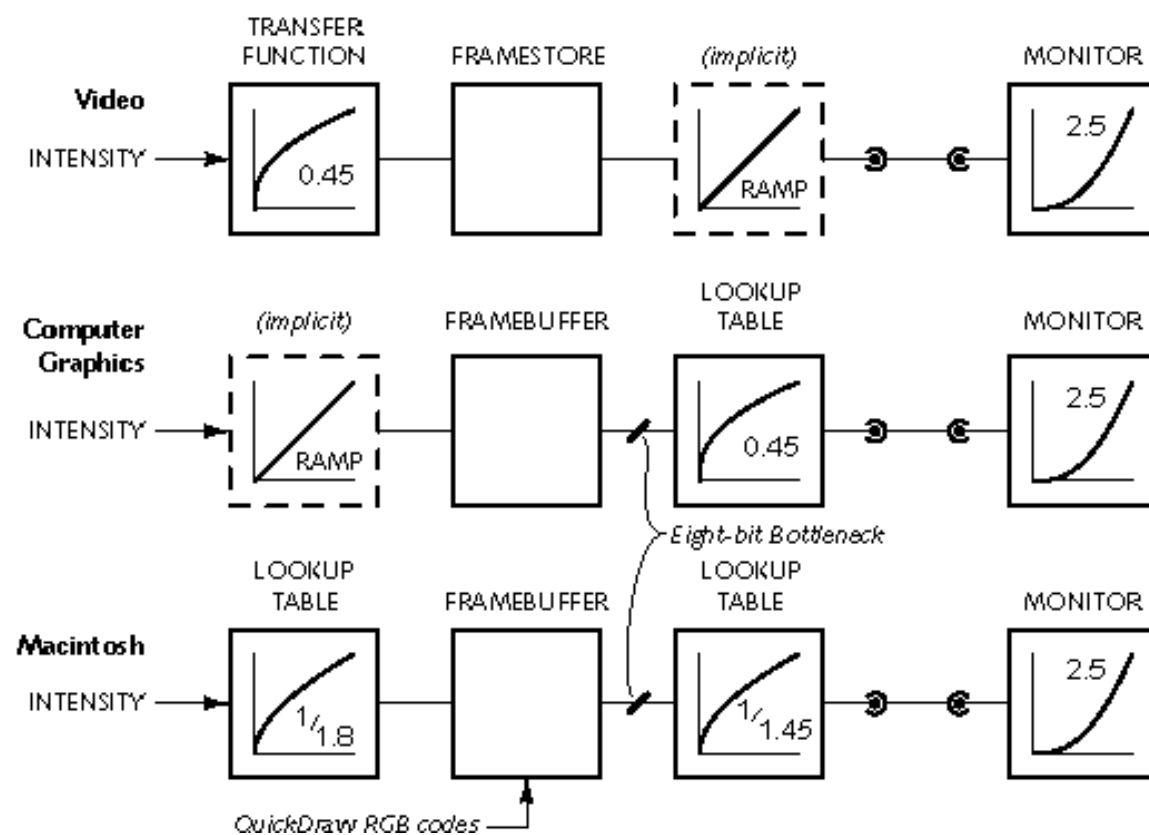
$$I \sim V_S^\gamma$$

- Quindi occorre modificare il segnale per compensare l'errore: il segnale compensato V_C (da usare come input per il monitor) è:

$$V_C \sim V_S^{(1/\gamma)}$$



Correzione gamma /2

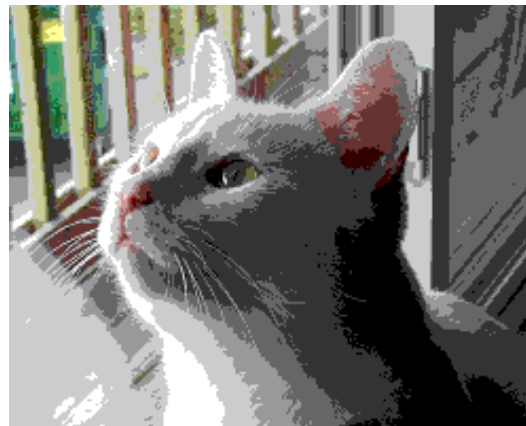


Correzione gamma /3

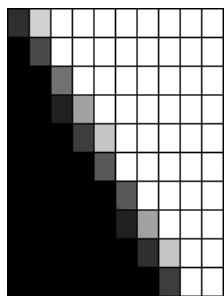
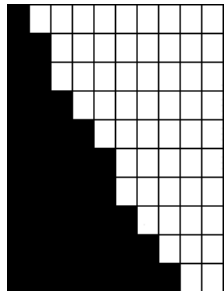
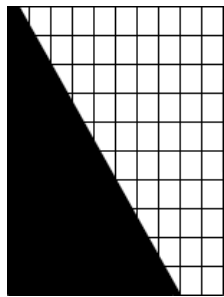
- Naturalmente occorre conoscere il valore di γ , che è propria del monitor usato (e può essere diversa per ogni colore primario)
- In altri dispositivi di output (es. monitor LCD) la relazione tra intensità e stimolo è molto più complicata, ma tali dispositivi sono “intelligenti”
 - applicano automaticamente una compensazione
 - e lasciano credere di essere un monitor CRT con un certo valore di γ

Problemi della digitalizzazione

- Pixel troppo grossi (bassa definizione)
 - Soluzione:
Anti-aliasing
- Pochi livelli di colore
 - Soluzione:
Dithering



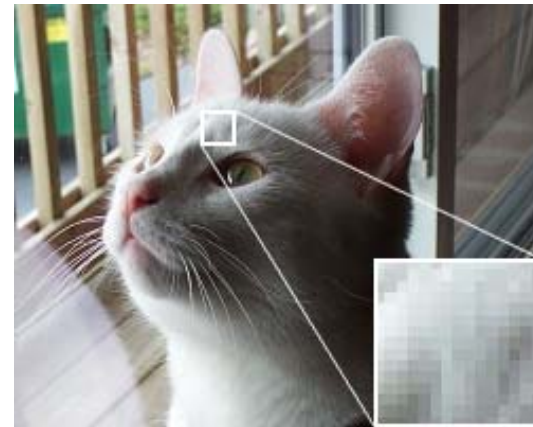
Anti-aliasing



- Ogni pixel è di un solo colore
- Ma nell'immagine originale corrisponde ad una zona potenzialmente ricca di colori
 - Così i confini appaiono “seghettati”
- Anti-aliasing: tecnica che consente di “smussare” i confini fra colori
 - Da applicare alle immagini subito prima che siano visualizzate

Dithering /1

- Digitalizzo un segnale (es. immagine) usando pochi bit per rappresentare un campione (es. il colore di un pixel)
- Pochi livelli di colore: grande **errore di quantizzazione**
 - Effetto “curve di livello”
- Soluzione: **Dithering**



Dithering /2

- Prima di digitalizzare, aggiungo del **rumore**
 - **Rumore**: segnale casuale di piccola ampiezza
- Così rendo più “frastagliata” la situazione al confine fra “curve di livello”
 - Immagine più gradevole e... realistica



Dithering /3

- Applicazione:
output su dispositivi con pochi colori
 - Stampanti
 - Giornali quotidiani
 - Laser
 - Getto di inchiostro
 - Schede grafiche “antiche”, con pochi colori
 - Display molto economici (telefoni mobili)