

Trasformazioni fra formati

- Esistono delle trasformazioni standard fra i diversi formati che abbiamo discusso
 - Il formato più “ricco” di informazioni (e più fedele) è il direct color
 - Può essere ridotto a indicizzato scegliendo, per esempio, i 256 colori che appaiono più frequentemente, oppure quelli “medi” fra gli altri simili (che minimizzano gli errori di approssimazione)

Trasformazioni fra formati

- Esistono delle trasformazioni standard fra i diversi formati che abbiamo discusso
 - I formati a colori possono essere ridotti a scala di grigi trasformando ogni colore in un valore di **luminanza**
 - Si moltiplica ogni componente per una costante fissa, che dipende dalla sensibilità (media) dell'occhio umano ai vari colori
 - grigio = $0.299 \times \text{rosso} + 0.587 \times \text{verde} + 0.114 \times \text{blu}$

Trasformazioni fra formati

- Infine, la scala di grigi può essere ridotta a b/n stabilendo un **valore soglia**
 - tutti i grigi più chiari della soglia diventano bianco
 - tutti i grigi più scuri della soglia diventano nero
- Ovviamente le trasformazioni inverse sono possibili, ma non si recupera l'informazione persa!

Trasformazioni fra formati

- Alcuni esempi di diversi valori soglia:



Originale
(in scala di grigi)



soglia 63 (25%)



soglia 127 (50%)



soglia 163 (64%)



soglia 192 (75%)

Esempi



- Colore diretto, **86.700** byte
 - 170x170, 24 bit colore diretto
- Colore indicizzato, **29.668** byte
 - 170x170, 24 bit colore, 256 colori
- Scala di grigi, **28.900** byte
 - 170x170, 8 bit colore, 256 grigi
- Bianco/nero, **3.613** byte
 - 170x170, 1 bit colore, b/n

Ancora sull'occupazione di memoria

- I calcoli sull'occupazione di memoria che abbiamo fatto riguardano la dimensione delle immagini **in memoria** – per esempio, mentre vengono visualizzate
- Per memorizzare le immagini su disco, o trasmetterle via rete, è conveniente usare **algoritmi di compressione** per ridurre la dimensione

Algoritmi di compressione

- In generale, possiamo dividere gli algoritmi di compressione in due grandi famiglie:
 - algoritmi **senza perdita** (*lossless*): consentono di ricostruire esattamente i dati di partenza
 - algoritmi **con perdita** (*lossy*): si perde una parte dell'informazione; non è più possibile ricostruire esattamente i dati di partenza

Algoritmi lossless

- Gli algoritmi di compressione senza perdita applicati alle immagini sono gli stessi usati nei normali programmi di compressione
 - per esempio, algoritmi di Run Length Encoding, Huffman, Lempel-Ziv-Welch, compressione aritmetica
 - usati in PKZip, gzip, compress, lha, RAR, ARJ, ecc.
 - I **formati di file** per le immagini specificano uno o più algoritmi da adottare

Algoritmi lossy

- Gli algoritmi lossy sono invece progettati esplicitamente per le applicazioni alle immagini
- I due principali sono:
 - **JPEG** (Joint Photographic Expert Group), famosissimo algoritmo basato sulla trasformata inversa del coseno e sull'eliminazione delle frequenze “alte” (spaziali e colorimetriche), comunque invisibili all'occhio umano
 - **Wavelet**, algoritmo molto efficiente ma “sperimentale” a causa dell'alto costo computazionale

Algoritmi lossy

- Quando si decide di applicare un algoritmo di compressione lossy a un'immagine, è in genere possibile specificare un **fattore di qualità**
- Ad elevati fattori di compressione (e quindi, forti riduzioni di dimensioni) corrispondono forti perdite di qualità
- In genere, **non si usano algoritmi lossy** se l'immagine deve ancora essere “lavorata”.

Esempio (file JPEG, lossy)

170x170x24
dimensione in
memoria:
86.7 Kb



originale
q=100
24.8 Kb



q=75
8.1 Kb

q=50
5.6 Kb

q=25
3.7 Kb

q=5
1.4 Kb

Esempio (file PNG, lossless)

170x170x24
dimensione in
memoria:
86.7 Kb



originale



c=0
85.1 Kb

c=2
56.9 Kb

c=5
53.9 Kb

c=9
53.3 Kb

Esempio (file PNG, lossless)

- Con gli algoritmi lossless, è a volte possibile specificare un **fattore di compressione**
- Visto che con gli algoritmo lossless non si perde **mai** in qualità, la scelta di una compressione maggiore **non** peggiora l'immagine
 - aumenta solo (in maniera impercettibile) il tempo per caricare e salvare l'immagine
- Conviene quindi usare sempre la compressione maggiore!

Principali formati grafici

- Con compressione lossless:
 - TIFF (Tagged Image File Format)
 - GIF (Graphics Interchange Format)
 - PNG (Portable Network Graphics)
 - BMP (Windows Bitmap)
 - TGA (schede grafiche Targa)
- Con compressione lossy:
 - JPEG (Joint Photographic Expert Group)

Prossimi argomenti

- Cenni sulle tecniche di compressione
 - JPEG
- Informazioni sui formati grafici più comuni
- Cenni sull'elaborazione digitale di immagini
 - Concetti e algoritmi
 - Tool di uso comune: Photoshop, GIMP

Riferimenti

- http://www.diodati.org/scritti/2001/algoritmi/algor_stam.asp contiene un conciso ma leggibile excursus delle principali tecniche di compressione di immagini
- <http://www.freeonline.org/art/a-128/> raccoglie un po' delle sigle che abbiamo incontrato, con relative (succinte) spiegazioni
- Un elenco di colori con la relativa codifica RGB è accessibile all'URL <http://riemann.unica.it/studenti/guida/colori.html> (ma se ne trovano centinaia di altri, più o meno estesi)
 - È possibile selezionare un qualunque colore RGB e vedere il suo codice a <http://www.allprofitallfree.com/color-wheel2.html>
- www.gimp.org è la home page di GIMP, il programma di grafica bitmap da cui abbiamo tratto le immagini sui selettori colore (e che useremo in laboratorio)