



**Università degli Studi di Palermo**

***Dipartimento dell'Innovazione Industriale e Digitale***

# **Sistemi di Elaborazione delle Informazioni – 6 CFU**

Anno Accademico 2016/2017

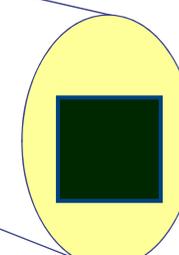
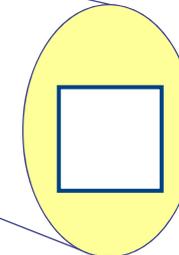
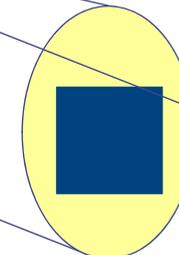
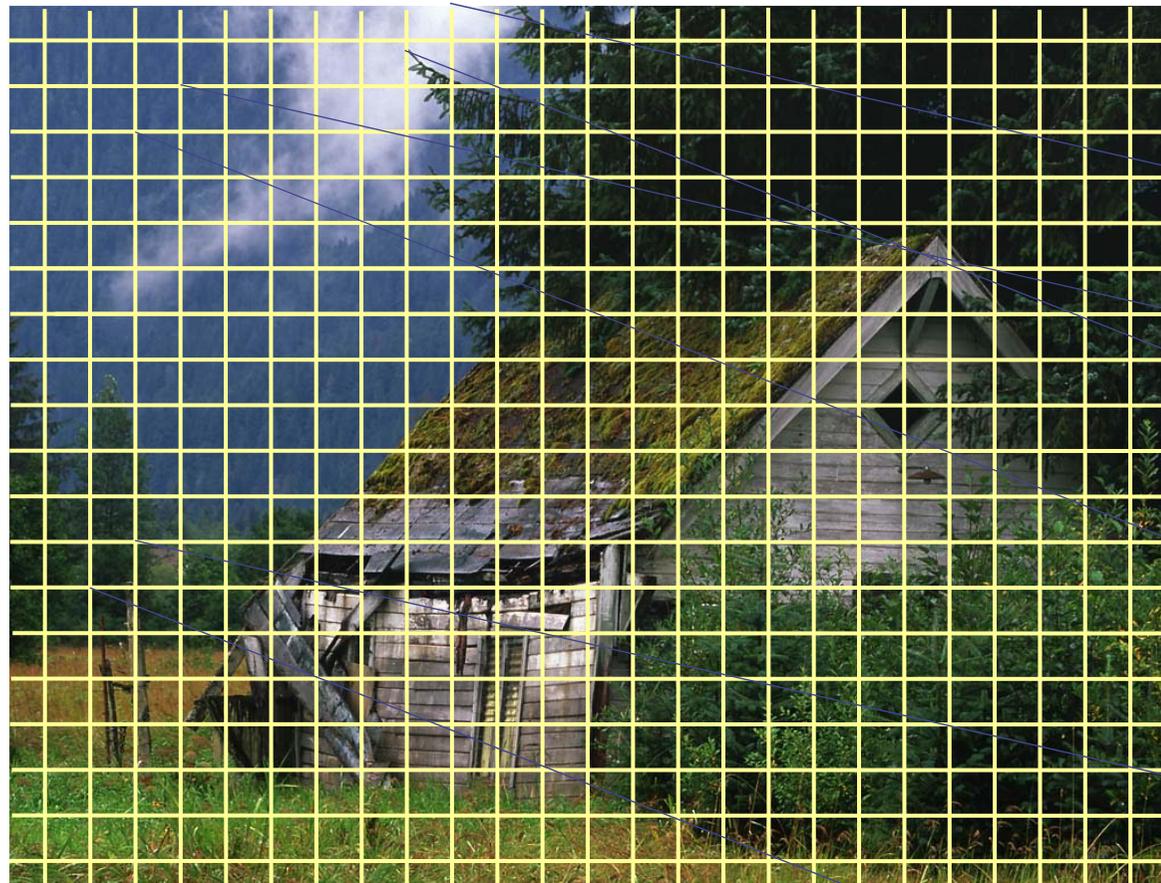
Docente: ing. Salvatore Sorce

**Rappresentazione delle informazioni**

III parte: Codifica binaria di immagini



## Digitalizzazione delle immagini



## Digitalizzazione delle immagini

- Rappresentazione binaria dei colori RGB
  - un colore sul monitor è specificato da tre componenti: rosso, verde e blu
  - l'intensità di ogni colore è rappresentata da una quantità (da 0 a 255)
  - Ogni intensità RGB è rappresentata da un byte (8 bit)
  - l'intensità più bassa è 0000 0000, la più alta è 1111 1111
  
- Alcuni esempi:
  - **Rosso** -> R 1111 1111 G 0000 0000 B 0000 0000 (0xFF0000)
  - **Verde** -> R 0000 0000 G 1111 1111 B 0000 0000 (0x00FF00)
  - **Blu** -> R 0000 0000 G 0000 0000 B 1111 1111 (0x0000FF)
  - -> R 1111 1111 G 1111 1111 B 1111 1111 (0xFFFFFFFF)
  - **Nero** -> R 0000 0000 G 0000 0000 B 0000 0000 (0x000000)
  - **Giallo** -> R 1111 1111 G 1111 1111 B 0000 0000 (0xFFFF00)
  - **Magenta** -> R 1111 1111 G 0000 0000 B 1111 1111 (0xFF00FF)
  - **Ciano** -> R 0000 0000 G 1111 1111 B 1111 1111 (0x00FFFF)
  - **Grigio** -> R 0100 1010 G 0100 1010 B 0100 1010 (0x4A4A4A)

## Il problema della banda passante

- La quantità di dati che è possibile trasmettere per unità di tempo
- Un aumento della banda passante significa una diminuzione della latenza (tempo di attesa affinché un'informazione sia interamente trasmessa)
- La banda passante dipende:
  - Dal mezzo trasmissivo
  - Dalle tecniche usate per la trasmissione
- Si esprime in bit/sec

## Immagini e video digitali

- Occorrerebbero 51 minuti per scaricare con un modem un'immagine a colori di 20 x 25 cm digitalizzata a 300 pixel per pollice (dpi) senza alcuna compressione
- Com'è possibile vedere immagini a tutto schermo in pochi secondi navigando sul Web?
- Un tipico schermo di computer ha meno di 100 pixel per pollice
  - un'immagine digitalizzata a 100 dpi richiede poco più di un decimo della memoria
    - ◆ richiede sempre 5 minuti e mezzo per essere spedita
- Soluzione: la compressione

## Compressione

- Cambiare la rappresentazione per usare un numero inferiore di bit per memorizzare o trasmettere informazioni
  - Esempio: i fax sono lunghe sequenze di 0 e 1 che codificano il contenuto di una pagina in bianco e nero. La codifica *run length* (Run-Length Encoding, RLE) è utilizzata per specificare la lunghezza della prima sequenza di 0, seguita dalla sequenza di 1 ecc.
  - la compressione run length è *lossless*, cioè senza perdita – la rappresentazione originale può essere ricostruita perfettamente

## Compressione JPEG

- Utilizzata per le immagini fisse
- I nostri occhi non sono sensibili alle piccole variazioni di tonalità, ma sono sensibili alle piccole variazioni di luminosità
  - fornisce una descrizione meno accurata del colore
  - la compressione ottiene un rapporto 20:1 senza variazioni percepibili ai nostri occhi



Un dettaglio da un'immagine compressa mediante JPEG. (a) compressione 14:1; (b) compressione 140:1.

## Compressione MPEG

- La stessa idea alla base del JPEG, ma applicata ai filmati
- Come la compressione JPEG, è applicata a ogni frame
- Poiché in genere due immagini consecutive sono molto simili
  - la compressione MPEG memorizza solo "la differenza" tra fotogrammi consecutivi
  - ciò permette livelli di compressione elevatissimi

## Il riconoscimento ottico dei caratteri

- Leggere le targhe per applicare il giusto pedaggio
- Quali sono le difficoltà?
  - i computer devono catturare l'immagine della targa ma la telecamera potrebbe vedere altre immagini della strada
  - Il *frame grabber* riconosce il momento ideale per scattare l'immagine che poi viene spedita al computer
  - il computer deve capire dove si trova la targa nell'immagine
    - ◆ esamina gruppi di pixel cercando i bordi dove il colore cambia
    - ◆ cerca di identificare le *caratteristiche* peculiari
    - ◆ un *classificatore* le confronta con quelle delle lettere dell'alfabeto

## Tecnologia OCR

- Permette al computer di "leggere" caratteri stampati
  - applicazioni commerciali: le poste e le banche

## I bit possono rappresentare tutto

- Principio del mezzo universale:
  - per mezzo dei bit si può rappresentare ogni sorta di informazione discreta; i bit non hanno un significato intrinseco.
- I bit sono un mezzo universale
  - tutte le cose che possono essere rappresentate possono anche essere manipolate
- I bit non hanno preferenze
  - il significato dei bit deriva interamente dall'*interpretazione* che ne dà il computer attraverso il programma
- I bit non rappresentano necessariamente numeri
  - i bit possono essere interpretati come numeri, ma anche no

# I bit possono rappresentare tutto

23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	1	1	1	1	1	1	0	1	1	0	1	0	0	1	0	0	0	0	0	1	1	0	1

# I bit possono rappresentare tutto

23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	1	1	1	1	1	1	0	1	1	0	1	0	0	1	0	0	0	0	0	1	1	0	1

Un colore RGB:

<b>R = 254</b>	<b>G = 210</b>	<b>B = 13</b>
----------------	----------------	---------------



# I bit possono rappresentare tutto

23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	1	1	1	1	1	1	0	1	1	0	1	0	0	1	0	0	0	0	0	1	1	0	1

Un numero intero a 24 bit:

**16.699.917**

# I bit possono rappresentare tutto

23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	1	1	1	1	1	1	0	1	1	0	1	0	0	1	0	0	0	0	0	1	1	0	1

Un numero esadecimale a 6 cifre:

F	E	D	2	0	D
---	---	---	---	---	---

# I bit possono rappresentare tutto

23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	1	1	1	1	1	1	0	1	1	0	1	0	0	1	0	0	0	0	0	1	1	0	1

Due campioni di un segnale audio (L+R):

L = -19	R = +525
---------	----------

# I bit possono rappresentare tutto

23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	1	1	1	1	1	1	0	1	1	0	1	0	0	1	0	0	0	0	0	1	1	0	1

Una sequenza di caratteri ASCII:

p	ò	CR (ritorno a capo = INVIO)
---	---	-----------------------------

# I bit possono rappresentare tutto

23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	1	1	1	1	1	1	0	1	1	0	1	0	0	1	0	0	0	0	0	1	1	0	1
$S_N$	Mantissa														$S_e$	Esponente							

$$-0.111111011010010 \times 2^{13} = -1111110110100.10_2 = -8116.5_{10}$$

## Digitalizzazione delle immagini

- Quali colore rappresentano questi valori?

1100 1000 1100 1000 1100 1000

- Ogni byte corrisponde al valore decimale 200. Il colore è rappresentato da RGB (200, 200, 200).
  - In HTML, scritto in esadecimale è #C8C8C8
  - ha una quantità uguale di rosso, verde e blu, è più vicino al bianco che al nero (grigio di media intensità)
  - tutti i colori con componenti RGB uguali sono il nero, il bianco e il grigio
- Aumentare l'intensità con un'addizione binaria
  - per ottenere un grigio più chiaro, occorre modificare i valori in modo che si avvicinino al bianco