

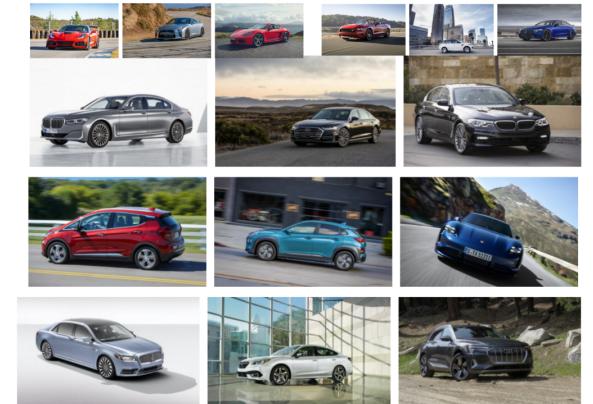


## ELABORAZIONE DELLE IMMAGINI

COLORE

## COLORE

- Il colore è un **descrittore** molto potente per **identificare** ed **estrarre** gli oggetti/dettagli dalla scena



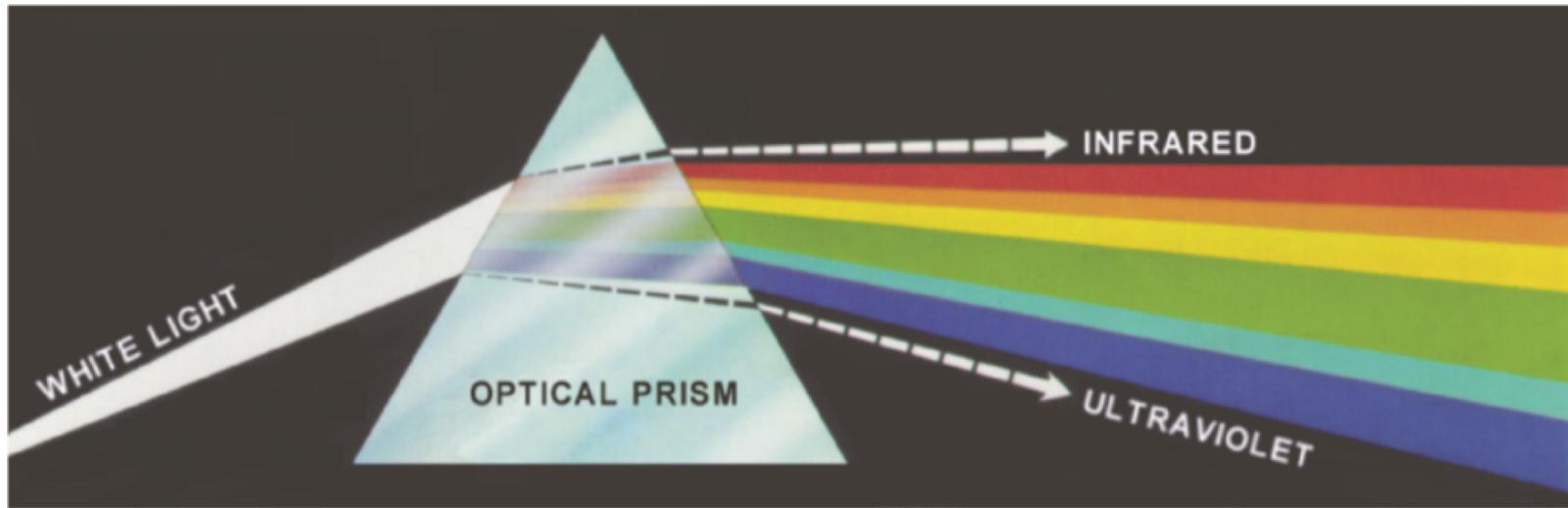
- L'**elaborazione** delle immagini a colori si divide in **due classi**
  - **Full-color** (colori acquisiti da un sensore full-color)
  - **Falsi colori** (colori assegnati a particolari valori di intesità)

↳ ad ogni livello di grigio è assegnato un colore, che ovviamente può non rispecchiare il vero colore dell'oggetto

Elim Parte II – Prof. A. Ferone

## PERCEZIONE DEL COLORE

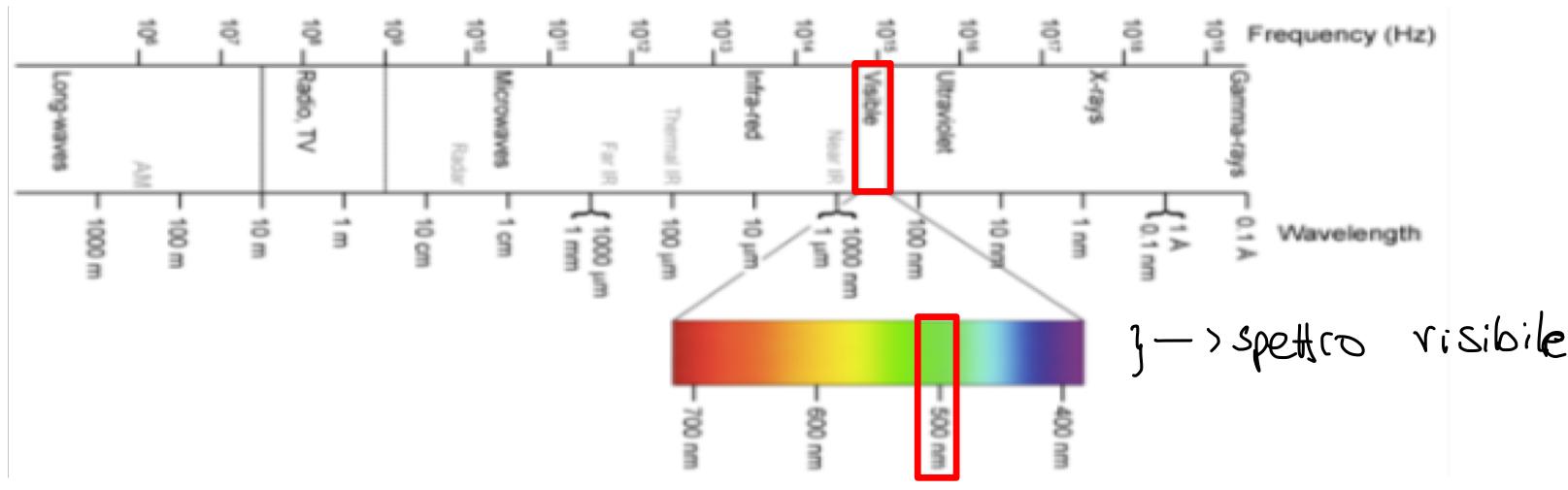
- Newton scoprì che, quando un **raggio di sole** passa attraverso un **prisma di vetro**, il raggio in uscita consiste in uno **spettro di colori** dal viola al rosso



Il prisma separa le luce nelle sue componenti (diverse frequenze)

## PERCEZIONE DEL COLORE

- La **luce visibile** è composta da una **banda** di frequenze relativamente **stretta**
- I **colori** sono determinati dalla **natura** della **luce riflessa** dagli **oggetti**
  - Un corpo che **riflette** la luce in **tutte** le lunghezze d'onda apparirà **bianco**
  - Un oggetto che riflette la luce con lunghezze d'onda tra 500nm e 570nm apparirà verde



Elim Parte II – Prof. A. Ferone

Come percepiamo il colore dell'oggetto?

- Quando la luce colpisce l'oggetto, una parte di essa viene assorbita dall'oggetto e un'altra riflessa
- Se l'oggetto assorbe tutta la luce → vediamo nero (perché non riflette nulla)
- Contrario → bianco

## CARATTERIZZAZIONE DEL COLORE

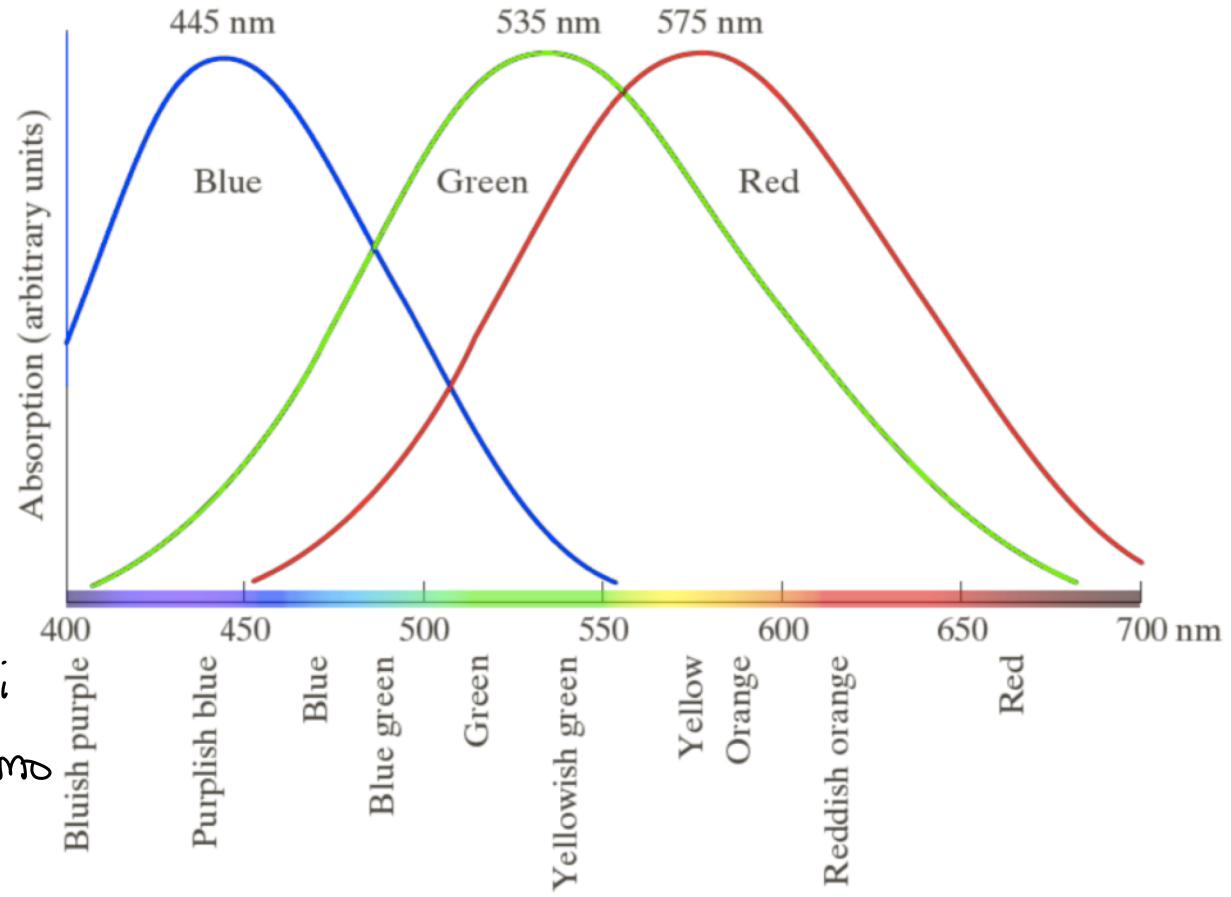
- La **luce acromatica** ha un **unico attributo**: l'**intensità** (livello di grigio)
- La **luce cromatica** è descritta da **tre quantità**
  - **Radianza**: quantità totale di energia che fuoriesce dalla fonte di luce (W)
  - **Luminanza**: misura della quantità di energia percepita (lumen)
  - **Luminosità**: descrittore soggettivo ingloba la nozione acromatica di intensità  
*sono dei fotorecettori:*
- I **colori** sono **percepiti** attraverso i **coni**, divisi in **tre categorie** percettive
  - 65% dei coni è sensibile alla **luce rossa**
  - 33% dei coni è sensibile alla **luce verde**
  - 2% dei coni è sensibile alla **luce blu** (più sensibili)

## ASSORBIMENTO DEI COLORI NEI CONI

- Le caratteristiche dell'occhio consentono di vedere i **colori** come **combinazioni** variabili dei **colori primari**
- R - **red**
- G - **green**
- B - **blue**

È dato del fatto che i coni lavorano insieme, quindi vediamo i colori che ognuno percepisce, come mescolati tra di loro

↳ ci permette di vedere tutti gli altri colori



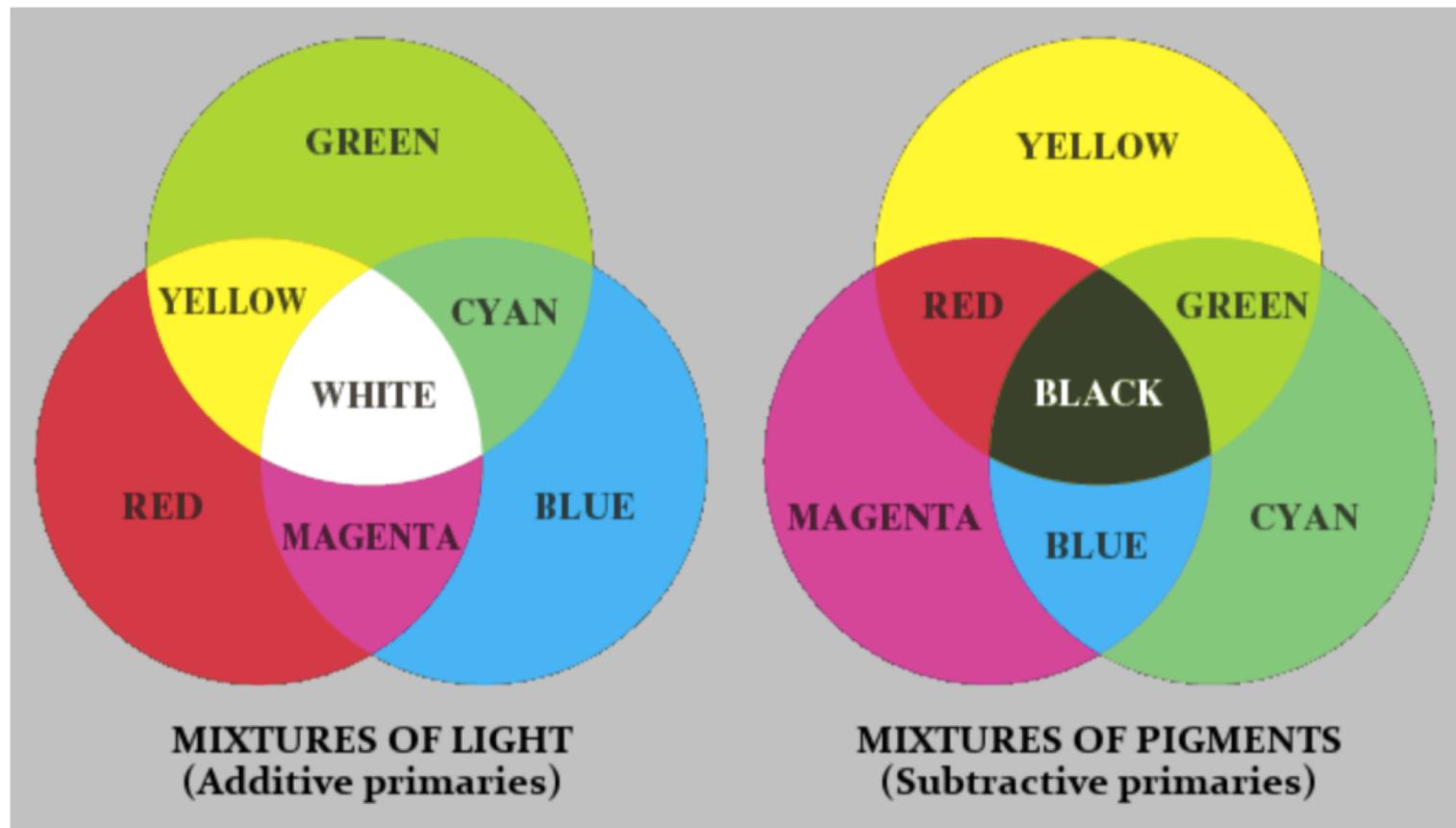
Elim Parte II – Prof. A. Ferone

## CARATTERIZZAZIONE DEL COLORE

- I **colori primari della luce** (RGB) possono essere **mescolati** per produrre i **colori secondari** della luce
  - **Magenta** (rosso + blu)
  - **Ciano** (verde + blu)
  - **Giallo** (rosso + verde) *o sottratti*
- I colori **primari dei pigmenti**, che assorbono/riflettono i colori primari della luce, corrispondono ai secondari della luce (Magenta, Ciano e Giallo)
- I colori secondari dei pigmenti corrispondono ai primari della luce (RGB)

es. Magonte assorbe il  
verde e riflette rosso e blu

# CARATTERIZZAZIONE DEL COLORE



## CARATTERIZZAZIONE DEL COLORE

- Per **distinguere** un **colore** da un altro si utilizzano **tre caratteristiche**:
  - **Luminosità** (misura la intensità) è il livello di grigio
  - **Tonalità** (Hue) associata alla **lunghezza d'onda dominante** (colore puro)
  - **Saturazione** che misura la **purezza della tonalità** (la **quantità** di bianco mescolato alla tonalità) [immagine se al rosso effunge bianco, lo sembra più] un colore saturo è più di bianco, è la tinta pura
- **Tonalità e saturazione** insieme vengono dette **cromaticità**
- Le **quantità** di **rosso**, **verde** e **blu** necessarie per formare un colore sono dette **valori tristimolo** ( $X$ ,  $Y$  e  $Z$ ) → percentuale di rosso, verde e blu presenti in un colore
- Un colore viene specificato mediante i **coefficienti tricromatici**

$$x = \frac{X}{X+Y+Z} \quad y = \frac{Y}{X+Y+Z} \quad z = \frac{Z}{X+Y+Z} \quad \rightarrow \quad x + y + z = 1$$

$$0 \leq x, y, z \leq 1$$

Elim Parte II – Prof. A. Ferone

## DIAGRAMMA DI CROMATICITÀ

- Un altro modo per **specificare** il **colore** è mediante il **diagramma di cromaticità** che mostra la **composizione** del **colore** in **funzione** di **x (rosso)** e **y (verde)**
- Dalla relazione  $x+y+z=1$ , è possibile **ricavare** il valore **z (blu)**
- Il punto in cui le **tre componenti** assumono **uguale valore** viene detto **punto di uguale energia (bianco)**
- I punti sul **bordo** del diagramma sono **completamente saturi** (senza bianco)
- **Avvicinandosi** al punto di **uguale energia**, **diminuisce** la **saturazione fino** ad arrivare a **zero** nel punto di uguale energia
- Un **segmento** lineare che unisce **due punti** individua **tutte** le possibili **variazioni** di **colore** ottenibili mediante **combinazione lineare** dei due **colori**

## DIAGRAMMA DI CROMATICITÀ

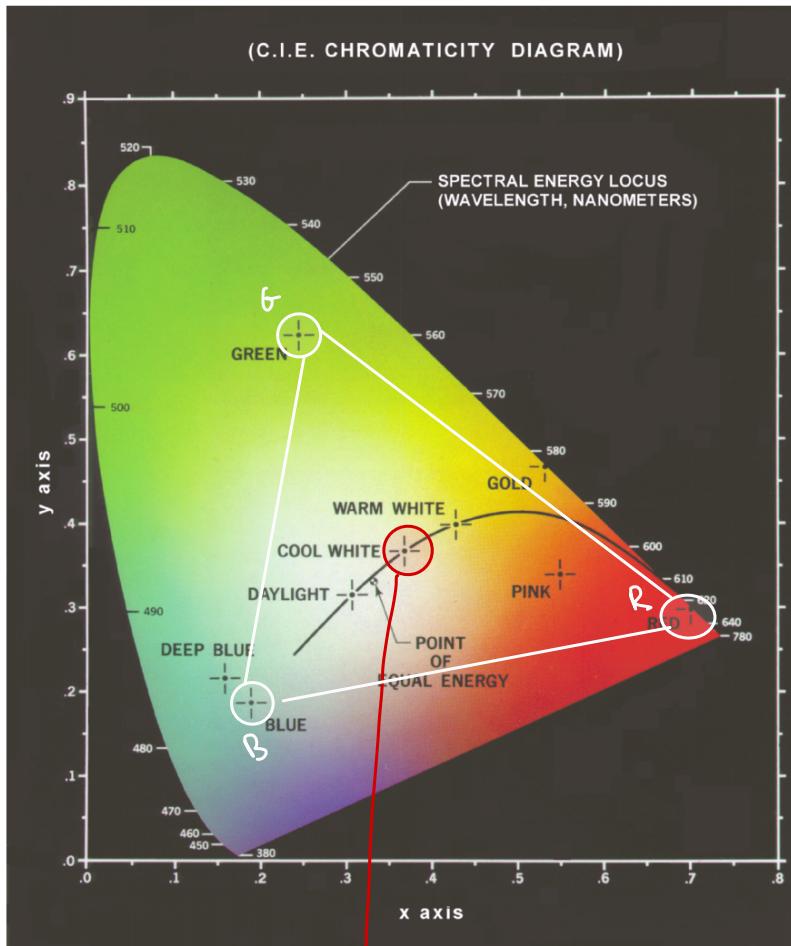
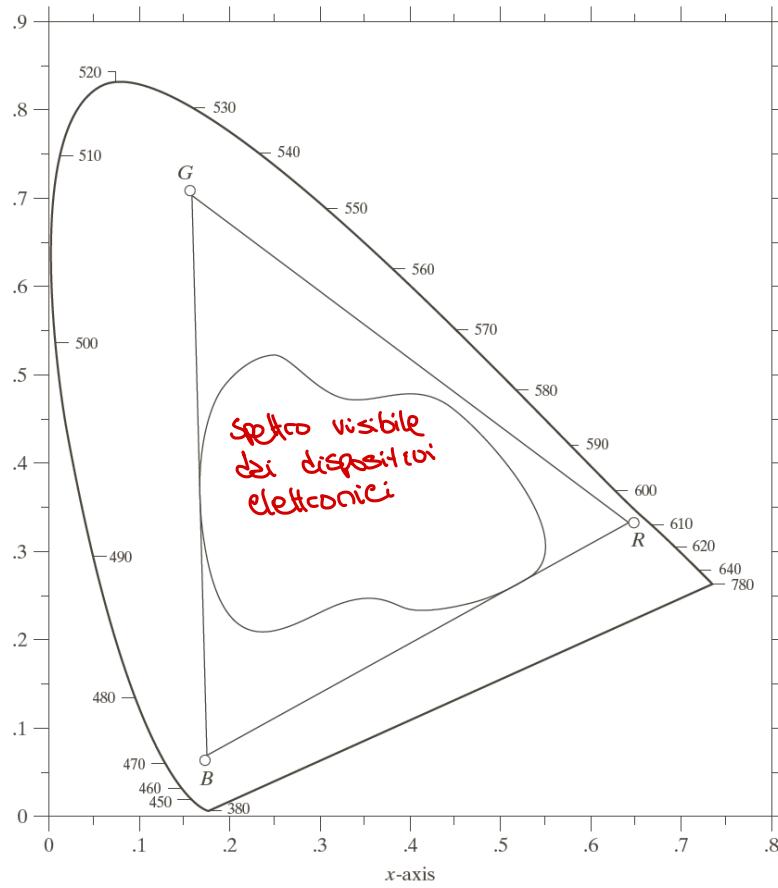


Diagramma bidimensionale dati i coefficienti tricromatici



Elim Parte II – Prof. A. Ferone

+ ci avviciniamo al bianco + i colori sono - saturi e diventano uguali al bianco

## MODELLI COLORE

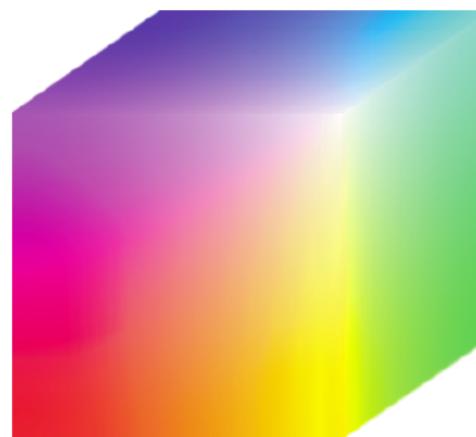
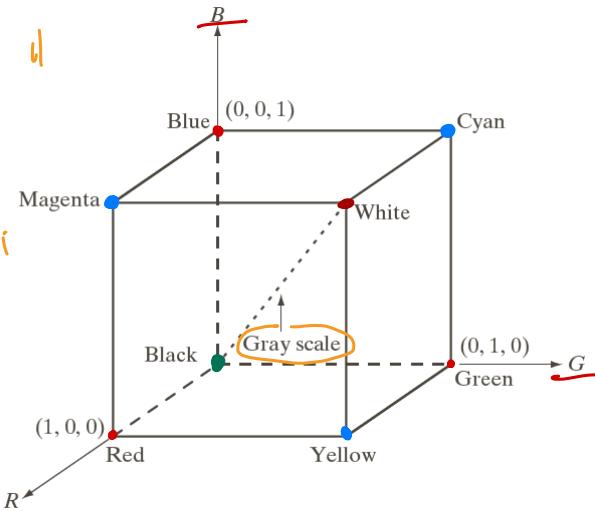
(es. coordinate cartesiane)

- Un **modello colore** o **spazio colore** è un **sistema di coordinate** ed un **sottospazio** di quel sistema in cui **ogni colore** è rappresentato da un **punto**
- I modelli colore più utilizzati sono
  - **RGB** per monitor/telecamere
  - **CMY** e **CMYK** per stampanti (sono i colori secondari) *Ciano, Magenta, Giallo* *Ciano, Magenta, Giallo, Nero*
  - **HSI** (HSV, HSL) simile al modo in cui gli uomini descrivono i colori
- Un **vantaggio** del modello HSI molto utile per **l'elaborazione** delle **immagini** è quello di **decorrelare** le informazioni dei **colori** dalla **intensità**

## RGB

- Nel modello **RGB** ogni **colore** è rappresentato dalle sue **componenti primarie**: rosso, verde e blu
- Si basa su un **sistema di coordinate cartesiane** ed il **sottospazio** di interesse è un **cubo** in cui i colori primari, quelli secondari, il bianco ed il nero occupano gli 8 vertici

diagonale che unisce il nero e il bianco è la scia di grigi  
e le stesse quantità di RGB



Elim Parte II – Prof. A. Ferone

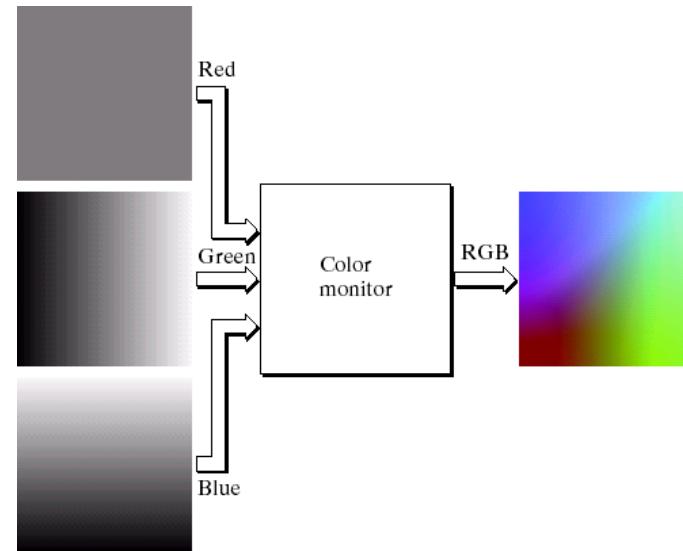
- Su 3 spigoli ho i colori primari
  - su altri 3 spigoli ho i colori secondari
  - nero
  - bianco
- ogni punto individuato nello spazio, è un colore

## RGB

- Le **immagini** rappresentate nel modello **RGB** sono **formate** da **tre immagini**, una per ogni **colore primario**
- In genere **ognuna** delle immagini rossa, verde e blu è un'immagine ad **8 bit**, per cui la **profondità** del pixele RGB è **24 bit** (3 piani x 8 bit)

ogni pixel è rapp. da 8 bit  
in totale RGB

253	144	120	251	41	147	204
67	100	32	241	23	165	30
209	118	124	27	59	201	79
210	236	105	169	19	218	156
35	178	199	197	4	14	218
115	104	34	111	19	196	
32	69	231	203	74		



Elim Parte II – Prof. A. Ferone

Ogni pixel è rapp. da 8 bit = 255

## CMY - CMYK

- Ciano, magenta e giallo sono i colori **secondari di luce o primari di pigmenti**
- Le **conversioni** da **RGB** a **CMY** e viceversa sono date dalle seguenti relazioni

$$\begin{bmatrix} C \\ M \\ Y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

*sottraggo da 1 i colori primari*

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} C \\ M \\ Y \end{bmatrix}$$

*inverso*

- **Uguali quantità** dei **pigmenti** primari non producono il **nero puro**, per cui si aggiunge un **quarto colore nero (K)**

*è + un marrone*

## HSI

- Quando si osserva un **oggetto colorato**, si è soliti descriverlo mediante **tre caratteristiche: tonalità, saturazione e luminosità**.
- La **tonalità** descrive il **colore puro**, la **saturazione** descrive la **quantità di bianco** contenuta nel colore mentre la **luminoistà** ingloba la nozione di **intensità**
- Il modello colore **Hue-Saturation-Intensity** divide queste tre componenti
- È un modello **ideale** per l'**elaborazione delle immagini**

## HSI - RGB

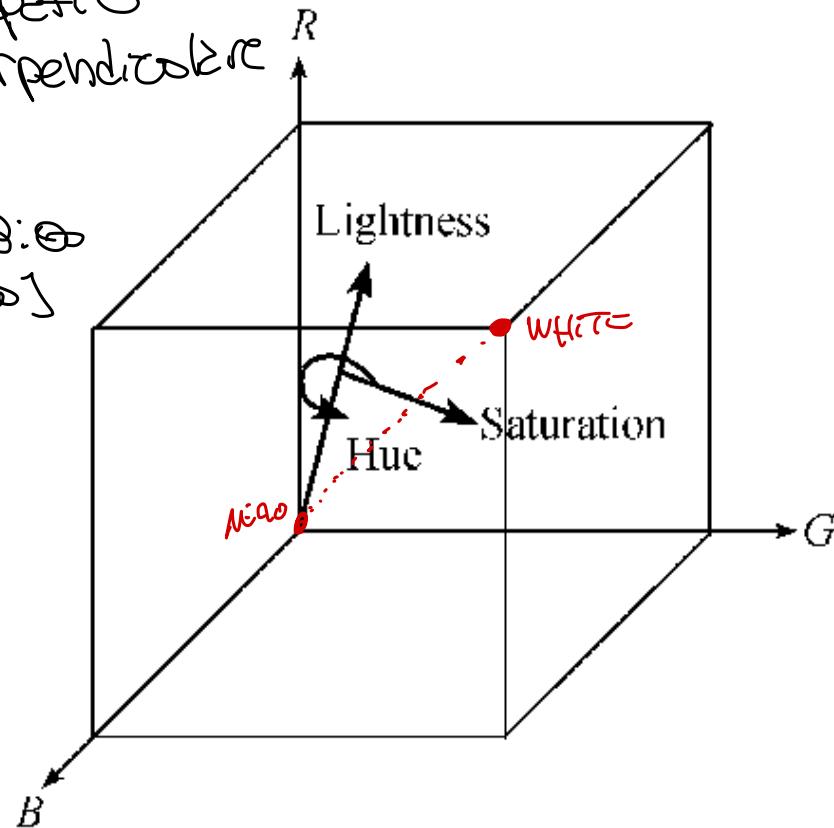
- Per determinare la componente di **intensità** di un **punto** colore **RGB**, bisogna far passare un **piano perpendicolare** all'asse di **intensità** che **contenga il punto** colore
- Il punto di **intersezione** tra l'**asse** di intensità ed il **piano** darà il valore di **intensità**
- La **saturazione** è data dalla **distanza del punto** colore dall'**asse** di intensità
- Unendo il **bianco**, il **nero** ed un **punto colore**, tutti i punti del **triangolo** ottenuto sono caratterizzati dalla **stessa tonalità**

Dobbiamo passare dal modello RGB al modello HSI

HSI

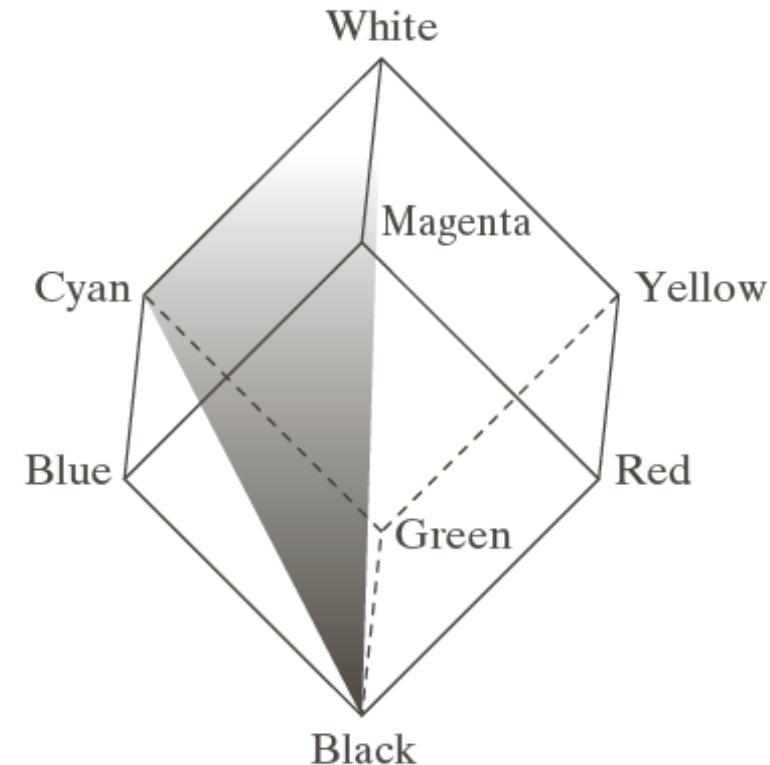
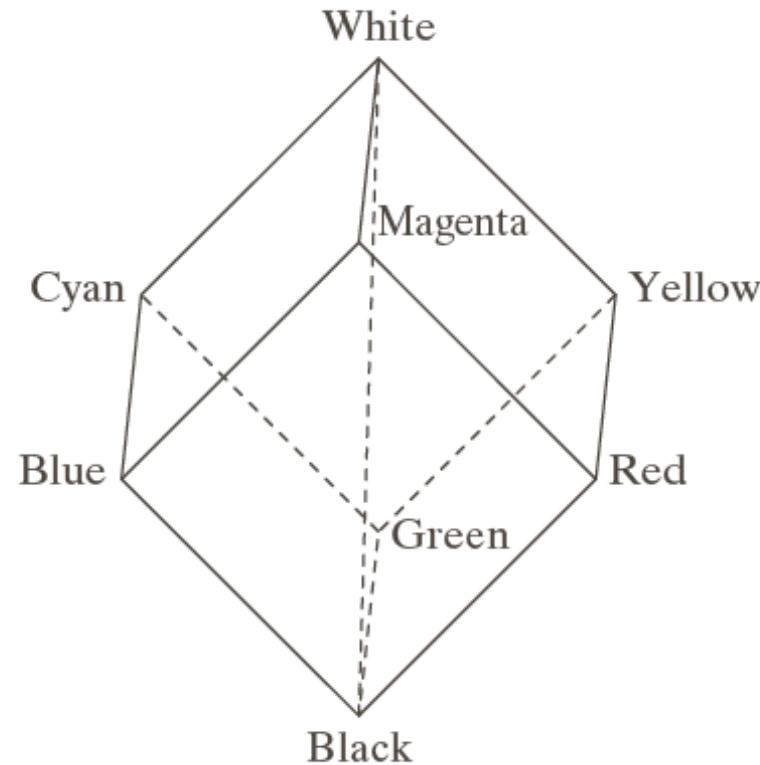
1. Dato un punto nello spazio  
si sceglie un piano perpendicolare  
alla diagonale

2. .... senti audio 30:0  
{credo}



Elim Parte II – Prof. A. Ferone

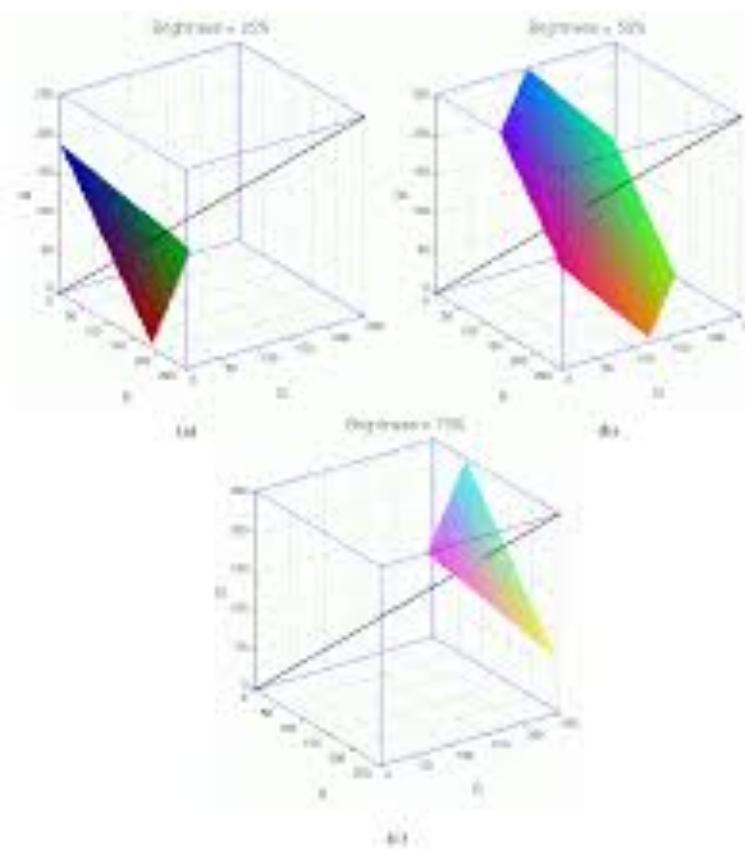
## HSI - RGB



## HSI

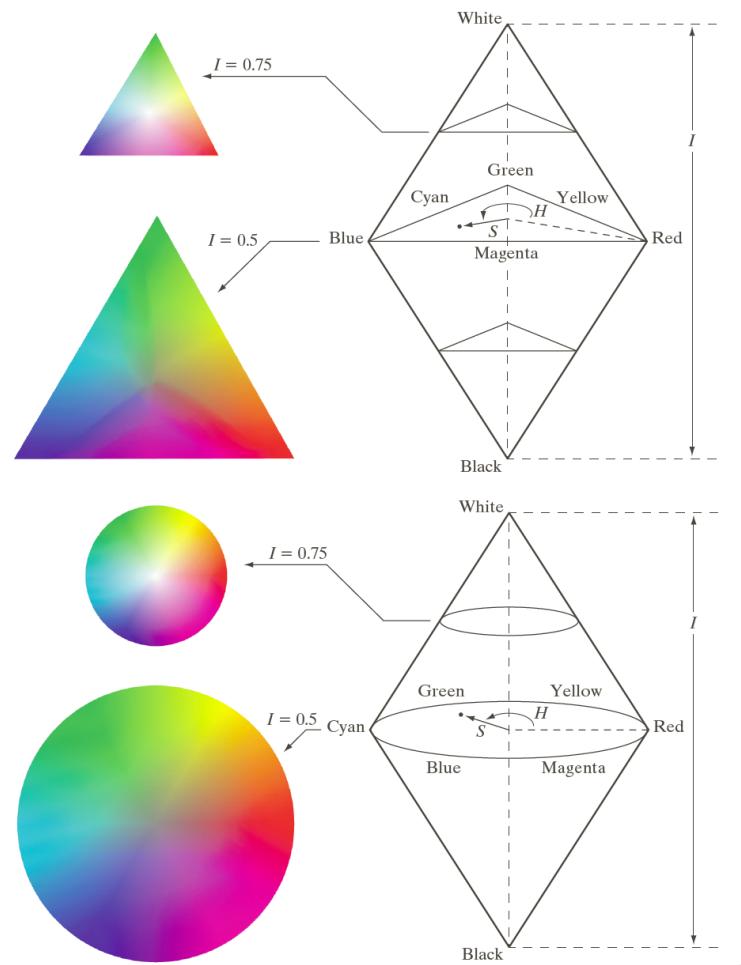
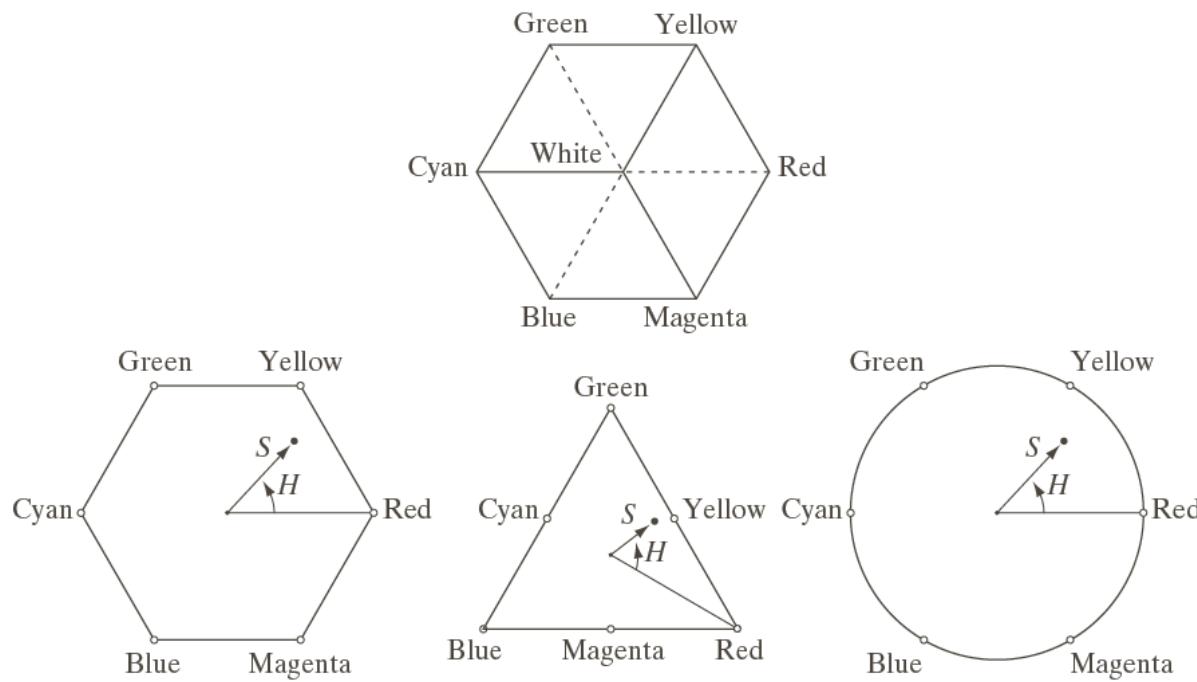
- Lo spazio **HSI** è rappresentato da un **asse di intensità verticale** e dal luogo dei **punti** colore che giacciono su **piani perpendicolari** all'asse
- I **piani** si **muovo** verso l'**alto** e verso il **basso** lungo l'asse di intensità, per cui l'**intersezione** con le facce del cubo ha forma **triangolare** o **esagonale**
- La **tonalità** del punto è determinata dall'**angolo** rispetto ad un **punto** (solitamente la **tonalità 0** corrisponde ad un **angolo di 0°** rispetto al **rosso**)
- La **tonalità** aumenta in **senso orario**

HSI



Elim Parte II – Prof. A. Ferone

# HSI

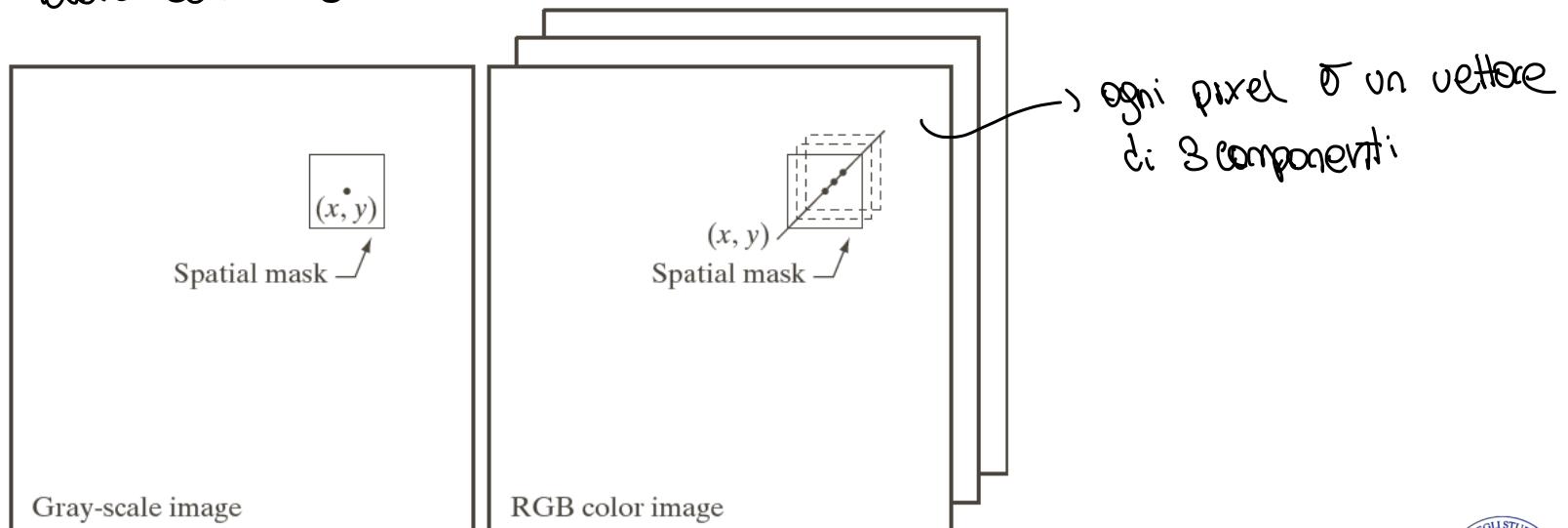


Elim Parte II – Prof. A. Ferone

## ELABORAZIONI FULL-COLOR

- **Due metodi** di elaborazioni delle immagini **full-color**

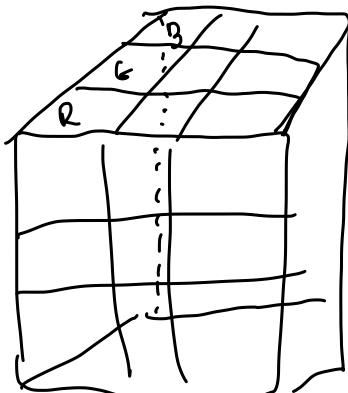
- stesso risotto* {
- Ogni **componente** viene elaborata **separatamente e combinata** con le altre. *cioè considerare le 3 bande come se fossero 3 immagini* 
  - Vengono **elaborati i pixel colore** (**tutte le componenti** sono elaborate **insieme**) *cioè lavorare con i vettori.*



Elim Parte II – Prof. A. Ferone

## SMOOTHING RGB

- Lo **smoothing** di un'immagine in scala di grigio si può realizzare con un'operazione di filtraggio spaziale ed una maschera opportuna
- Il **valore** di ogni **pixel** viene **sostituito** con la **media** dei valori dei pixel nell'**intorno** definito dalla maschera



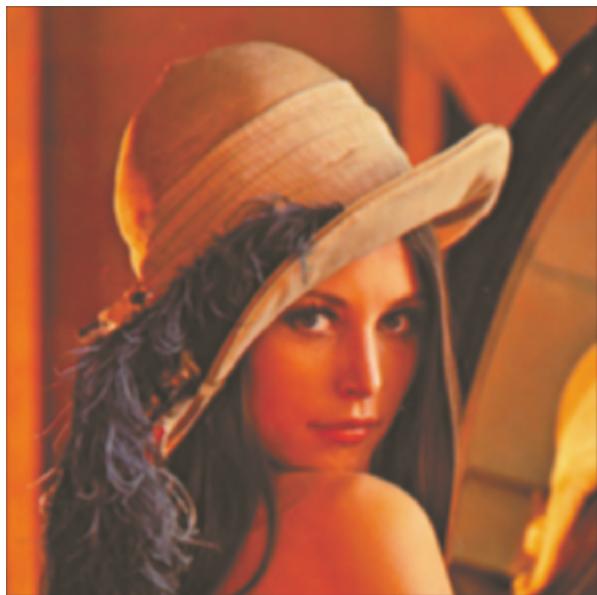
$$\bar{c}(x, y) = \frac{1}{K} \sum_{(x, y) \in S_{xy}} c(x, y)$$

$$\bar{c}(x, y) = \left[ \begin{array}{l} \frac{1}{K} \sum_{(x, y) \in S_{xy}} R(x, y) \\ \frac{1}{K} \sum_{(x, y) \in S_{xy}} G(x, y) \\ \frac{1}{K} \sum_{(x, y) \in S_{xy}} B(x, y) \end{array} \right]$$

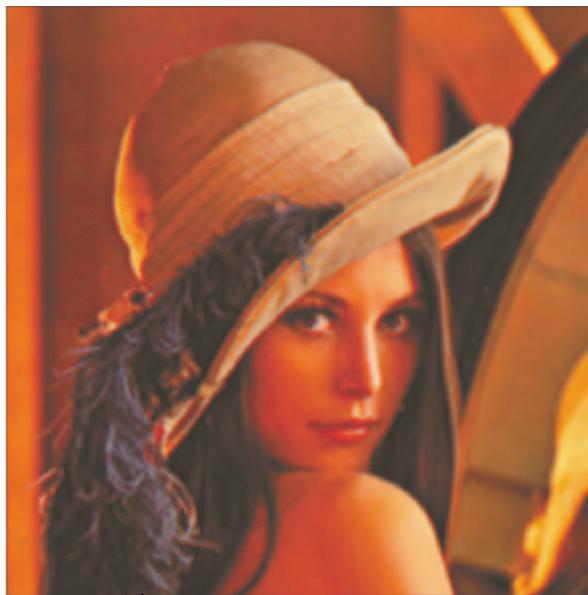
le operaz.  
sono le  
stesse.

## SMOOTHING HSI

- Se si utilizza la rappresentazione **HSI**, è possibile eseguire lo **smoothing** solo sulla componente **intensità** lasciando inalterate la **tonalità** e la **saturazione**



RGB



HSI



RGB lavora sui colori: impiega le erezioni

quindi fare una media tra i colori di uno stesso pixel di un nuovo colore

HSI permette di lavorare solo sull'intensità separando le componenti cromatiche

Elim Parte II – Prof. A. Ferone

comuniche

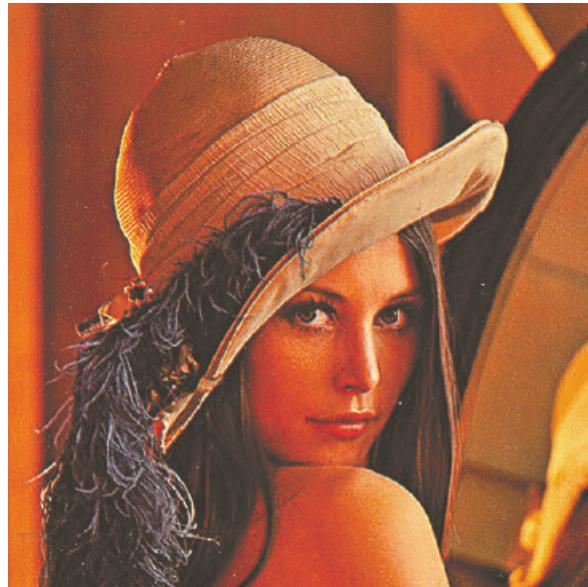
## SHARPENING RGB

- Lo **sharpening** si realizza tramite l'uso del **laplaciano**
- Il valore del **laplaciano** di un **vettore** è definito come un **vettore** le cui **componenti** sono uguali al valore **laplaciano** delle **componenti scalari** del vettore di input

$$g(x,y) = f(x,y) \pm \nabla^2 f(x,y) \quad \nabla^2 [\bar{c}(x,y)] = \begin{bmatrix} \nabla^2 R(x,y) \\ \nabla^2 G(x,y) \\ \nabla^2 B(x,y) \end{bmatrix}$$

## SHARPENING HSI

- Se si utilizza la rappresentazione **HSI**, è possibile eseguire lo **sharpening** solo sulla componente **intensità** lasciando **inalterate** la **tonalità** e la **saturazione**



è lo stesso disco

Elim Parte II – Prof. A. Ferone

# OPENCV: IMREAD

- La funzione **imread()**
  - determina automaticamente il tipo di file (BMP, JPEG, PNG, PPM, ...)
  - alloca la memoria necessaria per la struttura dati (**cv::Mat**) che conterrà i dati dell'immagine

```
cv::Mat cv::imread(  
    const string& filename,           // Input filename  
    int          flags   = cv::IMREAD_COLOR // Flags set how to interpret file  
)
```

Parameter ID	Meaning	Default
cv::IMREAD_COLOR	Always load to three-channel array.	yes
cv::IMREAD_GRAYSCALE	Always load to single-channel array.	no
cv::IMREAD_ANYCOLOR	Channels as indicated by file (up to three).	no
cv::IMREAD_ANYDEPTH	Allow loading of more than 8-bit depth.	no
cv::IMREAD_UNCHANGED	Equivalent to combining: cv::IMREAD_ANYCOLOR   cv::IMREAD_ANYDEPTH <sup>a</sup>	no

Elim Parte II – Prof. A. Ferone

## CV::IMREAD\_COLOR

- Il valore di default del secondo argomento è cv::IMREAD\_COLOR
- Le **immagini** sono caricate in **matrici a tre canali** con una profondità di **8bit per canale**
- Se l'**immagine** in **input** è a **colori**, viene allocata una **matrice** con **tre canali** in formato **BGR**
- Se l'**immagine** in **input** è in **scala di grigio**, viene allocata una **matrice** con **tre canali** (il livello di grigio viene replicato in tutti i canali)
- Passando il valore **cv::IMREAD\_ANY\_COLOR** le immagini vengono caricate in martrici con il numero di canali corretto

# ACCESSO AI CANALI

```
for(int i=0;i<image.rows;i++)  
    for(int j=0;j<image.cols;j++)  
        image.at<Vec3b>(i,j)[0]; //B  
        image.at<Vec3b>(i,j)[1]; //G  
        image.at<Vec3b>(i,j)[2]; //R
```

## SPAZI COLORE

- La funzione **cvtColor()** serve per modificare lo spazio colore con cui è rappresentata un'immagine, mantenendo lo stesso tipo
- L'array in **input** può essere a **8 bit, 16 bit unsigned o 32 bit floating point**
- L'array in **output** ha la **stessa dimensione** e la stessa **profondità** dell'array in **input**
- Il tipo di conversione è determinato dall'argomento **code**
- L'ultimo argomento è il **numero di canali** dell'immagine di output

```
void cv::cvtColor(  
    cv::InputArray src,           // Input array  
    cv::OutputArray dst,          // Result array  
    int code,                   // color mapping code  
    int dstCn = 0                // channels in output (0='automatic')  
);
```

Elim Parte II – Prof. A. Ferone

# SPAZI COLORE

Conversion code	Meaning
<code>cv::COLOR_BGR2RGB</code> <code>cv::COLOR_RGB2BGR</code>	Convert between RGB and BGR color spaces
<code>cv::COLOR_RGB2GRAY</code> <code>cv::COLOR_BGR2GRAY</code>	Convert RGB or BGR color spaces to grayscale
<code>cv::COLOR_GRAY2RGB</code> <code>cv::COLOR_GRAY2BGR</code>	Convert grayscale to RGB or BGR color spaces (optionally removing alpha channel in the process)
<code>cv::COLOR_RGB2XYZ</code> <code>cv::COLOR_BGR2XYZ</code>	Convert RGB or BGR image to CIE XYZ representation or vice versa (Rec 709 with D65 white point)
<code>cv::COLOR_RGB2HSV</code> <code>cv::COLOR_BGR2HSV</code> <code>cv::COLOR_HSV2RGB</code> <code>cv::COLOR_HSV2BGR</code>	Convert RGB or BGR image to HSV (hue saturation value) color representation or vice versa
<code>cv::COLOR_RGB2HLS</code> <code>cv::COLOR_BGR2HLS</code> <code>cv::COLOR_HLS2RGB</code> <code>cv::COLOR_HLS2BGR</code>	Convert RGB or BGR image to HLS (hue lightness saturation) color representation or vice versa

Elim Parte II – Prof. A. Ferone

## ESERCIZI

- Implementare le conversioni RGB2HSI (pag. 404) e HSI2RGB (pag. 406)
- Implementare smoothing e sharpening sulle componenti R, G e B
- Cofrontare il risultato con quello ottenuto modificando solo la componente di intensità della rappresentazione HSI

Split -> separare l'immagine in 3 componenti Mat(RGB)