

# **Appunti di Visione Artificiale**

**Riccardo Lo Iacono**

Dipartimento di Matematica & Informatica  
Università degli studi di Palermo  
Sicilia  
a.a. 2022-2023

## Indice.

<b>1</b>	<b>L'occhio umano e la visione artificiale</b>	<b>2</b>
<b>2</b>	<b>Le immagini a colore e i modelli colori</b>	<b>3</b>
2.1	Spazio colore RGB . . . . .	3
2.2	Spazio colore HSV . . . . .	3
2.3	Spazio colore YUV . . . . .	3
<b>3</b>	<b>Filtraggio di immagini digitali</b>	<b>4</b>
3.1	Filtro di convoluzione blur . . . . .	4
3.2	Filtri di denoise . . . . .	5

## – 1 – L'occhio umano e la visione artificiale.

Il concetto di visione artificiale è legato all'analisi di immagini digitali, nelle loro diverse caratteristiche. Risulta dunque ovvio che comprendere come l'occhio umano percepisca ed elabori le immagini, risulti fondamentale. Per quanto appena detto si descrive a seguire la struttura dell'occhio (*Figura 1.1*).

Questi é un'organo dalla forma pressoché sferica, si compone di diverse membrane delle quali di principale interesse è la retina. Questa è la membrana più interna sulla quale sono disposti i *fotocettori* responsabili della visione. Tali fotocettori sono divisi in *coni* e *bastoncelli*: i primi in minor numero, sono disposti attorno la fovea e sono responsabili della visione a colori, i secondi presenti sull'intera superficie della retina sono responsabili della visione a scale di grigio.

**Osservazione:** da quanto detto circa i fotocettori presenti nell'occhio umano, ne risulta che gli esseri umani sono molto più sensibili a variazioni di luce che non di colore.

**Osservazione:** per quanto concerne la visione a colori, si tenga presente che l'occhio umano è maggiormente sensibile, nell'ordine, a variazioni di verde, rosso e blu.

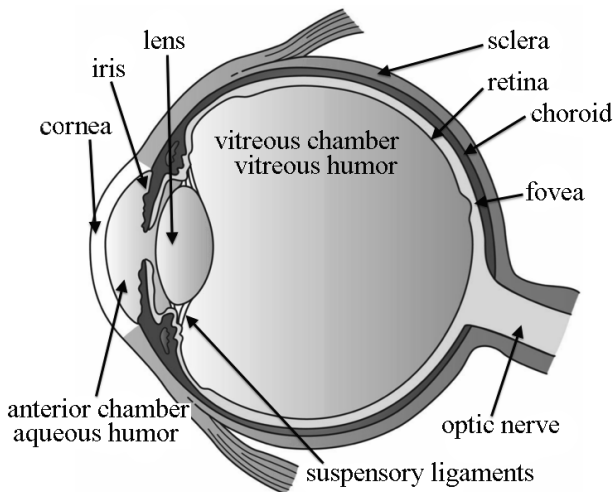


Figura 1.1: Schema dell'occhio umano

## – 2 – Le immagini a colore e i modelli colori.

Per poter essere rappresentate, le immagini a colori necessitano di un'appropriata descrizione in macchina. Tale descrizione è realizzata utilizzando opportuni *spazi colore*: delle decine che esistono, di interesse risultano RGB, YUV e HSV nel seguito descritti.

### – 2.1 – Spazio colore RGB.

Nello spazio colore RGB ciascun colore è descritto in funzione dei suoi valori di rosso, di verde e di blu. Ciascuno dei tre canali (componenti colore), ammette 256 sfumature distinte; segue da ciò la possibilità di rappresentare oltre 16 milioni di colori distinti. Generalmente tra i più noti, è di norma impiegato per la memorizzazione di immagini in macchina data la sua semplicità implementativa.

#### – 2.1.1 – Spazio colore RGB: CCD e il filtro di Bayer.

L'acquisizione di immagini digitali, è effettuata tramite l'uso di CCD. Questi, in breve, sono dispositivi che contano il numero di fotoni in una certa areola. Per quanto appena descritto però i CCD sono capaci di acquisire immagini a scale di grigio, come si può allora acquisire quelle a colore? L'idea più semplice sarebbe quella di usare un CCD per canale, dunque uno per il rosso, uno per il verde e uno per il blu. Un'implementazione di questo tipo risulta però complessa e costosa, ragione per cui nella maggior parte dei casi si sfruttano i filtri di Bayer. Si pensi a questi come delle maschere, di forma e dimensione variabile, che in un'area 2x2 della stessa presenta due elementi verdi, uno rosso e uno blu<sup>1</sup>.

### – 2.2 – Spazio colore HSV.

Nello spazio HSV ciascun colore è descritto in funzione di un angolo: per convenzione gli zero gradi sono il rosso, i centoventi il verde e i duecentoquaranta il blu. Circa il suo impiego, HSV è utilizzato per operare con le immagini via software.

### – 2.3 – Spazio colore YUV.

Nello spazio colore YUV ogni colore è separato nelle sue componenti di luminanza (livelli di grigi) e cromaticanza. L'importanza di questo spazio colore sarà maggiormente chiaro quando si parlerà del formato JPG (*Sezione ??*).

---

<sup>1</sup>Per comprendere il motivo di tale disparità si veda l'osservazione fatta in *Sezione 1*

## – 3 – Filtraggio di immagini digitali.

Lavorando con le immagini potrebbe essere necessario dover modificare le stesse in qualche modo: accentuando i dettagli o evidenziandone i contorni; ma come fare ciò? La risposta: tramite l'applicazione di *filtri*, suddivisi in *convolutivi* e non. Prima di parlare di tali filtri è però opportuno dare una definizione di convoluzione<sup>2</sup>. Sia  $I$  un'immagine e sia  $K$  una matrice  $N \times M$ , detta kernel<sup>3</sup>, i cui valori fungono da pesi per l'operazione. Per assicurare che non vi sia perdita di informazione il kernel è normalizzato. Il processo di convoluzione sovrappone  $K$  ad  $I$ , effettua un prodotto punto-punto tra gli elementi del kernel e quelli sottostanti ad esso nell'immagine, e di questi ne calcola la somma  $t$ . Successivamente in una nuova immagine, sia questa  $I'$ , nelle coordinate puntate dall'elemento centrale del kernel, si aggiunge un pixel il cui livello di grigio è esattamente  $t$ .



Figura 3.1: Immagine base: Lena.

### – 3.1 – Filtro di convoluzione blur.

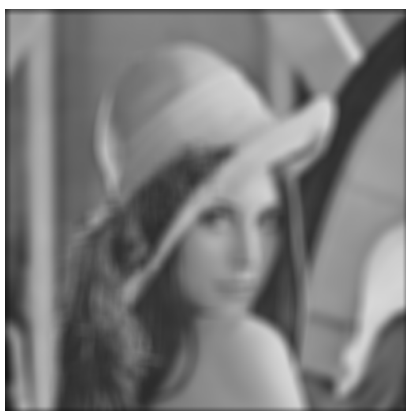


Figura 3.2: Applicazione di un filtro di blur a Figura 3.1.

Il filtro di convoluzione blur permette di applicare una sfocatura all'immagine; sfocatura che è tanto maggiore tanto maggiori sono i pesi del kernel e le sue dimensioni. Relativamente al kernel utilizzato, questi è a seguito riportati.

```
%% loading Lena.png
ker = fspecial("average", [21, 21]);
out = conv2(Lena, ker, "same");
```

Analizzando il codice: `fspecial("type", "sizes")` permette di definire un kernel con tipo e dimensioni specificate; `conv2("base", "ker")` effettua la vera e propria convoluzione. Il perché del parametro "same" sarà discusso successivamente.

<sup>2</sup>La definizione qui data è relativa l'analisi di immagini, per una definizione più rigorosa dal punto di vista matematico si veda: <https://en.wikipedia.org/wiki/Convolution>

<sup>3</sup>Per semplificare la ricerca dell'elemento centrale, in generale  $M$  ed  $N$  sono numeri dispari.

### – 3.2 – Filtri di denoise.

Capita spesso quando si opera con immagini digitali che queste siano soggette a rumore di varia natura; è dunque fondamentale, prima di operare con tali immagini, rimuovere quanto più rumore possibile, per farlo si utilizzano i filtri di denoise. Come detto il rumore si può presentare in varie forme, dal semplice rumore “sale e pepe”, si veda *Figura 3.3a*; al più fastidioso rumore gaussiano. In questa sezione si descriverà come rimuovere le due tipologie di rumore citate.

#### – 3.2.1 – Filtro mediano.

Il filtro mediano tenta di risolvere il problema del rumore “sale e pepe”, secondo la seguente logica. Poiché il rumore sale-pepe aggiunge quà e là pixel bianchi e neri per l’immagine, da cui il nome, l’idea è quella di considerare una porzione  $M \times N$  dell’immagine, una cosiddetta finestra  $W$ , procedere ad ordinare secondo tonalità di colore i pixel nella stessa e sostituire quello in posizione centrale al pixel in posizione centrale della finestra. Questo perché un pixel nero (analogamente uno bianco) non sarà eliminato se  $\frac{M*N}{2} + 1$  dei pixel nella finestra sono neri.



a: Esempio di immagine soggetta a rumore sale e pepe.



b: Applicazione di filtro mediano a *Figura 3.3a*.

**Osservazione:** l’applicazione del filtro potrebbe non rimuovere completamente tutto il rumore. A tal proposito si potrebbe riapplicare il filtro aumentato le dimensioni della finestra, o applicare il filtro unicamente ai pixel soggetti a rumore. Ulteriore problema del filtro è il fatto che finestre troppo grandi potrebbero trasformare il filtro in uno di blur.

#### – 3.2.2 – Filtro gaussiano.

Il filtro gaussiano tenta di risolvere il problema del rumore gaussiano. L’idea è quella di utilizzare una funzione gaussiana per spargere il rumore per tutta l’immagine, così da accentuarne gli effetti. Poiché di poco interesse al corso, non si riporta nessuna applicazione dello stesso.