

# **ELABORAZIONE DELLE IMMAGINI**

## INFORMAZIONI GENERICHE SUL CORSO

**Docente:** Dott.ssa Binaghi

### **Elearning informatica**

Utente: STU.F52

Password: 67xa26

### **Lezioni frontali + operative**

**Programma:** ImageJ

**HIPR2** (<http://homepages.inf.ed.ac.uk/rbf/HIPR2/>): praticamente è quello che verrà spiegato a lezione, in inglese

**Testo consigliato:** Marini, Bertolo, Rizzi, *Comunicazione Visiva Digitale*, 2001

### **Modalità d'esame**

Analisi ragionata delle immagini, non memorizzazione “a macchinetta” dei concetti

Analisi di un istogramma di un'immagine e scelta del filtro migliore

## LEZIONE 1: 04 marzo 2009

### Lezione 1, slides 1 – 53

**Competenze matematiche:** operatori (come influiscono sull'immagine), funzioni, sommatoria, revisioni, moltiplicazioni

Un'immagine può essere elaborata per fini ultimi o per essere ulteriormente elaborata da algoritmi più complessi e magari automatici

**Segmentazione** di un'immagine: separazione di un soggetto dallo sfondo → selezione primitiva della percezione visiva umana → è la premessa al riconoscimento automatico (da parte del programma grafico) dell'oggetto distinto dallo sfondo

**Immagine:** una distribuzione (bi- o tridimensionale) di un'entità fisica che contiene informazioni descrittive riferite all'oggetto, alla scena che l'immagine rappresenta

Sensore che cattura l'energia irradiata su una determinata scena

**Acquisizione** → teoremi per ridurre al minimo la perdita di informazioni; in caso di evidente degrado (l'immagine non viene visualizzata distintamente, per esempio per un'eccessiva compressione del range dei valori relativi al colore) è possibile effettuare un'operazione di **miglioramento** per recuperare le informazioni senza dover acquisire di nuovo l'immagine

[Jet Propulsion Laboratory (laboratorio NASA)]

Fino alla **segmentazione** l'immagine resta tale; dalla **rappresentazione/descrizione** in poi l'immagine viene “smontata” in elementi analizzabili secondo parametri matematici da parte di un algoritmo; il riconoscimento automatico è utile su grandi moli di dati (medicina, sicurezza...)

A parità di scena, una **risoluzione** maggiore consente la rappresentazione di più elementi  
La risoluzione migliora, ad esempio, restringendo il range di colori acquisiti (da un'immagine a colori a una in scala di grigi)

**Restauro** dell'immagine per recuperare informazioni, migliorandone la rappresentazione (blurring, stretching del contrasto, aggiungere del “rumore”, colorazione dei livelli di grigio...)

## LEZIONE 2: 05 marzo 2009

### Laboratorio HIPR2; Lezioni 7 e 8 (cenni)

Se l'immagine è in scala di grigi, il valore che indica ogni pixel è unico, ed esprime l'intensità del grigio; in caso di immagini a colori devono essere indicati valori molteplici: la più usata è la scala RGB (ogni pixel è indicato da tre valori).

#### **Operatori sulle immagini – Addizione** (<http://homepages.inf.ed.ac.uk/rbf/HIPR2/pixadd.htm>)

L'addizione consente di sommare i valori di pixel di immagini differenti o di una costante per migliorare l'effetto visivo → evidenziare dei contorni, migliore nitidezza, riduzione del rumore (il segnale si rafforza, mentre il rumore, che è casuale, viene perso poco per volta)

La somma deve essere normalizzata affinché il valore del risultato si collochi tra 0 e 255. Infatti, se in un'immagine in scala di grigi la somma dei valori raggiunge il valore 255 si ha il bianco; se tale valore viene superato, si ha comunque 255 come valore massimo → **saturazione** (da evitare) → è necessaria la **normalizzazione** per risistemare i valori dei pixel entro il range 0 – 255.

#### **Operatori sulle immagini – Sottrazione** (<http://homepages.inf.ed.ac.uk/rbf/HIPR2/pixsub.htm>)

La sottrazione è un'operazione più complessa dell'addizione. Essa considera i valori possibili di un pixel dell'immagine finale compresi tra -255 (ovvero  $I_1 = 0$  e  $I_2 = 255$ ) e +255 (ovvero  $I_1 = 255$  e  $I_2 = 0$ ). Per effettuare una sottrazione si devono seguire i seguenti step:

1. Sottrarre il valore del pixel dell' $I_2$  dal corrispondente dell' $I_1$
2. Sommare il negativo del minimo dell'immagine differenza al valore del pixel considerato
3. Moltiplicare il valore così ottenuto per  $255/\text{MAX}$ , dove MAX è il valore massimo dell'immagine ottenuta al punto 2

### Lezione 2, slides 1 – 18; 24 - 38

Anni '20: trasmissione con cavi sottomarini

Anni '60: viaggio nello spazio; elaborazione immagini della Luna

Anni '70: invenzione della TAC; elaborazione immagini dell'apparecchio

Immagini classificate in base alla sorgente:

- Radiazione elettromagnetica
- Ultrasuoni (radar, GPR...)
- Elettroni
- Generazione di distribuzione di luminosità al computer (immagini grafiche)

L'uomo vede solo lo spettro tra il rosso e il violetto (e non UV o Ir) perché il “segnale” luminoso del Sole ha il picco di intensità proprio in quella porzione dello spettro. Il colore è un fenomeno psico-visuale legato all'osservatore.

L'infrarosso vicino e medio rileva il segnale infrarosso riflesso; l'infrarosso termico considera quello emesso direttamente dall'oggetto.

Radar: sensore attivo → produce un segnale e lo raccoglie

Nell'immagine a ultrasuoni è difficile effettuare un'operazione di elaborazione dell'immagine per via del forte disturbo

Nell'immagine ottenuta dall'emissione di elettroni si ha un ingrandimento notevole, ma l'oggetto viene spesso danneggiato (il materiale biologico viene sempre distrutto)

## LEZIONE 3: 10 marzo 2009

### Lezione 3, slide 1 – 26

L'immagine catturata dal sensore è relativa all'energia riflessa dalla superficie del corpo osservato. Il valore di intensità si determina dalla quantità di energia irradiata, dalla proporzione tra energia riflessa e non, sensibilità del sensore... → determinazione in scala di grigio dell'intensità energetica

**Legge di Plank** → applicata sul modello di un “corpo nero” utilizzato come standard per elaborare la legge; conoscendo la natura chimico-fisica dell'oggetto osservato e l'intensità energetica irradiata, posso calcolare l'energia riflessa e quindi l'intensità di grigio dell'immagine finale; nei corpi reali (non ideali) si aggiunge alla legge di Plank un coefficiente di riflessione

Ogni oggetto ha una propria **firma spettrale**, ovvero un determinato insieme di valori di energia luminosa riflessa

**Radianza = coefficiente di energia incidente \* coefficiente di riflessione**

Luminanza = energia elettromagnetica nello spettro del visibile

### Lezione 4, slide 1 - 52

Coni: visione completa; bastoncelli: visione acromatica

Il *contrast stretching* è necessario per migliorare il contrasto di elementi di un'immagine vicini tra loro. Infatti l'occhio umano percepisce la differenza cromatica solo se il  $\Delta I$  raggiunge un determinato valore

Le nostre percezioni prospettiche sono intermedie tra le dimensioni prospettiche e le dimensioni effettive dell'oggetto → nell'immagine del porticato ad archi, gli archi in fondo hanno un'altezza visibile di circa la metà di quelli iniziali, mentre in realtà sono alti uguali → li percepiamo come alti all'incirca i 2/3

### Presentazione di un progetto svolto: la percezione

### Lezione 5, slide 3 – 10; 15 – 21; 24 - 34

**Discretizzazione** (digitalizzazione) = Campionamento + Quantizzazione

**Campionamento**: relazione tra il pixel dell'immagine (con valore uniforme) e una griglia regolare; e gli conferisce un valore discreto che varia a seconda della scala utilizzata (1 bit: bianco e nero; 8 bit: scala di grigi...)

Nella digitalizzazione dell'immagine, il piano cartesiano che definisce la matrice viene ruotato: infatti, l'origine nella matrice deve essere in alto a sinistra, mentre nel piano cartesiano normale essa si trova in basso a sinistra.

**Dimensione = Grandezza \* Risoluzione**

## LEZIONE 4: 11 marzo 2009

### Lezione 5, slide 9 – 13; 35 - 48

**Frequenza di campionamento:** frequenza con cui viene catturata l'immagine

Per ottenere un segnale uniforme e con minima perdita di informazione, la frequenza di campionamento deve essere almeno 2 volte maggiore della frequenza massima del segnale stesso

**Teorema di Fourier:** per quanto un segnale sia complesso, esso può essere percepito come l'insieme di onde di segnale semplici combinate tra loro; questo consente di individuare molto facilmente la frequenza massima

Con un'immagine ricca di dettagli (ad alta frequenza) è possibile risparmiare sui bit dei livelli di grigio; in un'immagine a bassa frequenza, invece, è preferita una maggiore gradualità nella sfumatura

**Modifica di N (grandezza):** *zooming* (ingrandimento, aggiunta di righe/colonne, interpolazione dei livelli di grigio in modo automatico per creare livelli intermedi) o *shrinking* (riduzione, interpolazione dei toni di grigio in modo automatico per uniformare i livelli intermedi)

Due tipi di interpolazione: *nearest neighbor* (→ effetto scacchiera) o *bilinear* (→ smoothing dell'immagine)

**Criterio di vicinanza:** due tipologie di individuazione dei punti *vicini* al punto di coordinate  $(x, y)$

1. A 4 punti: le coordinate dei punti vicini sono  $(x+1, y)$ ,  $(x-1, y)$ ,  $(x, y+1)$ ,  $(x, y-1)$
2. A 8 punti: le coordinate dei punti vicini sono  $(x+1, y)$ ,  $(x-1, y)$ ,  $(x, y+1)$ ,  $(x, y-1)$ ,  $(x+1, y+1)$ ,  $(x+1, y-1)$ ,  $(x-1, y+1)$ ,  $(x-1, y-1)$

**Criterio di adiacenza:** tre tipologie di adiacenza

1. A 4: i valori di grigio devono essere compresi nello stesso intervallo, il punto  $Q$  deve appartenere all'intorno-4 di  $P$
2. A 8: i valori di grigio devono essere compresi nello stesso intervallo, il punto  $Q$  deve appartenere all'intorno-8 di  $P$
3. A  $m$ : simile all'adiacenza a 8, è relativa alla diagonale, ma l'intorno a 4 dei pixel  $P$  e  $Q$  deve essere vuoto (non devono avere elementi in comune nel proprio intorno)

**Funzione di distanza:** esistono diversi sistemi di calcolo → ambiguità

1. Distanza a 4:  $D(p,q)=(x-s)+(y-q)$  → il cerchio euclideo viene approssimato ad un rombo
2. Distanza a 8:  $D(p,q)=\max(|(x-s)|,|(y-q)|)$  → il cerchio euclideo viene approssimato a un quadrato

### Lezione 6, slide 1 - 19

**Image Enhancement:** miglioramento dell'immagine; diverse modalità; la *finestra* considerata può variare a seconda dell'algoritmo usato o a discrezione dell'operatore stesso

L'intorno minimo di un pixel è il pixel stesso → **intorno puntuale**

Se l'intorno è maggiore di  $1 \times 1$  → **intorno locale**

Se viene considerata l'intera immagine → **intorno globale**

Se vengono considerate più immagini → **operatore algebrico/logico**

L'intorno puntuale consente, ad esempio, di **binarizzare** l'immagine (solo bianco e nero, senza scala di

grigio) → con particolari sistemi di individuazione della soglia, è possibile individuare semplicemente i bordi degli oggetti distinti dallo sfondo

**Trasformazioni di base:** si pongano X come valore iniziale di grigio, e Y come valore finale

1. **Lineari:** *identità* ( $y = x$ ) → nessuna trasformazione; *negativo* ( $y = 255-x$ )
2. **Esponenziali:** *nth root* (i valori di grigio vengono schiariti); *nth power* (i valori di grigio vengono scuriti)
3. **Logaritmiche:** *log* (dilatazione zona scura, riduzione zona chiara) e *inverse log* (dilatazione zona chiara, riduzione zona scura) → queste funzioni permettono di visualizzare al meglio immagini con preponderanza di elementi scuri (*log*) o chiari (*inverse log*)

**Gamma correction** → uso della funzione *log* o *inverse log*

**Contrast stretching** → è possibile applicarlo anche a singole sezioni dell'immagine, per esaltare il contrasto solo in range di determinati valori: *funzioni lineari a pezzi* → è necessario definire bene i punti “di rottura”, per migliorare il contrasto senza perdere informazioni

**Intensity slicing:** esaltazione solo di una determinata porzione dell'immagine, i cui valori di grigio appartengono tutti a un determinato range di valori → i valori esterni al range possono essere azzerati o lasciati così com'erano nell'immagine originale

## LEZIONE 5: 12 marzo 2009

### Lezione 6, slide 19 - 24

L'esaltazione di alcuni livelli di grigio non corrisponde all'esaltazione di particolari porzioni dell'immagine: se si vuole ottenere tale risultato, è necessario introdurre condizioni particolari

**Slicing:** i livelli di grigio vengono sostituiti da colori, seguendo un codice prestabilito → mappe di rischio sismico, carte altimetriche...

Per ottenere tale risultato, l'immagine deve prima essere scomposta nei tre livelli (RGB) e quindi si procede all'assegnazione di determinati intervalli di grigio a determinati colori

Ogni immagine è costituita da pixel. Ognuno di essi è codificato in scala di grigi a 8 bit, per cui ogni immagine è costituita da 8 “piani” di bit. L'informazione che caratterizza l'immagine può essere meglio definita solo in alcuni piani; per cui eliminando i livelli meno significativi si ottiene un risparmio di memoria (**compressione**).

### Lezione 7, slide 1 - 7

#### **Istogramma di un'immagine**

1. Individuazione di tutti i pixel con un determinato valore nella scala di grigi
2. Costruzione dell'istogramma a barre: sull'asse delle ascisse metto i valori possibili della scala di grigio (da 0 a 255), sulle ordinate invece il numero di pixel

Per confrontare due immagini di dimensioni diverse, è necessario normalizzare l'istogramma → va diviso il valore di pixel di una data sfumatura di grigio per il numero totale dei pixel dell'immagine stessa → valore percentuale; la **normalizzazione** dà, come risultato della somma di tutti i valori, 1. La “gradevolezza” nella percezione di un'immagine è associata all'uniformità delle percentuali di grigio → **equalizzazione** per uniformare i livelli di grigio e rendere l'immagine più gradevole. L'istogramma può aiutare a separare l'oggetto dallo sfondo o a determinare la soglia su cui lavorare con altri operatori (*intensity slicing, contrast stretching...*)

### Laboratorio HIPR2

**Pixel Connectivity:** <http://homepages.inf.ed.ac.uk/rbf/HIPR2/connect.htm>

**Connected Components Labeling:** <http://homepages.inf.ed.ac.uk/rbf/HIPR2/label.htm>

**Point Operations:** <http://homepages.inf.ed.ac.uk/rbf/HIPR2/pntops.htm>

1. Thresholding
2. Adaptive thresholding
3. Contrast stretching
4. Histogram equalization
5. Logarithm operator
6. Exponential/“Raise to power” operator

### Lezione 8, slide 1 - 12

Una somma delle immagini di una stessa scena riduce il rumore: perché? Il rumore altera i valori

dei pixel in modo casuale e ha media 0, quindi ciò che è segnale viene riconfermato con l'addizione, mentre il rumore (che in un punto aumenta il segnale e nello stesso di un'altra immagine lo riduce) viene assorbito e scompare.

**AND** di un'immagine con una maschera nera con finestra bianca → la porzione nella finestra resta invariata, il resto va a 0

**OR** di un'immagine con una maschera bianca con finestra nera → la porzione nella finestra resta invariata, il resto va a 255

**Scaling** → necessario per far rientrare nel range (0, 255) i valori che oltrepassano il massimo

### Lezione 9, slide 1 - 22

**Trasformazioni spaziali** → prendono in considerazione il pixel e il suo intorno

Bassa frequenza: variazione minima

Alta frequenza: variazione ampia

Questo è utile perché, tendenzialmente, i bordi di un oggetto sono ad alta frequenza (forte contrasto con lo sfondo). Togliendo le basse frequenze da un'immagine, si ottiene un rafforzamento dei bordi dell'oggetto.

#### **Come funziona una maschera?**

La maschera, costituita da un insieme di pixel (maschera minima: 3 x 3), i cui pixel hanno un determinato valore per modificare l'immagine originale. Essa viene spostata pixel dopo pixel per applicare l'effetto in modo omogeneo.

La maschera viene applicata dall'angolo superiore sinistro; alcune maschere modificano l'immagine in modo da poter modificare anche il pixel all'angolo estremo, altre invece iniziano ad applicare la maschera dal primo punto disponibile (in caso di una maschera 3 x 3, il primo punto è P(2, 2)).

Quindi, o la maschera non modifica i pixel dei bordi, o vengono inserite delle righe e delle colonne di pixel inesistenti nell'originale.

**I pixel dell'immagine vengono modificati secondo la maschera, tale per cui ogni valore del pixel viene moltiplicato per quello corrispondente nella maschera, quindi si effettua una somma di tutti i valori ottenuti e si sostituisce questo valore al pixel centrale della porzione di immagine a cui è sovrapposta la maschera.**

#### **Smoothing**

Sfocamento dell'immagine; la maschera attenua il contrasto dei pixel dei bordi.

Si realizza facendo la media dei valori del pixel centrale e del suo intorno, quindi sostituendo il valore del pixel centrale stesso con questo valore medio ottenuto.

Si dice che questo filtro è *passa basso*, nel senso che vengono esaltate le basse frequenze e ridotte o eliminate quelle alte.

Può servire per eliminare il rumore, in quanto esso è costituito da elementi ad alta frequenza, o eliminare i piccoli dettagli dell'immagine (per trovare più facilmente gli elementi di dimensioni maggiori che sono d'interesse)

#### **Filtro mediano**

La maschera *vuota* viene sovrapposta all'immagine originale. Quindi si fa la *mediana* dei valori all'interno della maschera e si sostituisce il valore ottenuto a quello del pixel centrale.

La mediana non è la media. La mediana non considera gli elementi di picco, come quelli causati dal rumore. Se il rumore è forte, questo filtro è il migliore.

## LEZIONE 6: 17 marzo 2009

Lezione 9, slide 10; 14; 16 – 18; 20 - 24

### **Filtri lineari – Convolutivi**

Caratterizzato dall'insieme dei valori

**Smoothing** → il valore per cui viene moltiplicato il valore dei pixel nell'intorno del pixel centrale è 1; il risultato viene diviso per il numero dei pixel della maschera

Valori sempre positivi → in una porzione di immagine perfettamente omogenea, essa viene lasciata intatta; la variazione cresce di pari passo con la frequenza

**Valore mediano** → va bene per il rumore random, a media zero e con valori molto differenti da quelli dell'immagine originale

### **Minimo**

Considera il valore minimo all'interno della maschera, e vado a sostituire al valore centrale della maschera il valore minimo individuato → erosione delle zone chiare, espansione delle zone scure

### **Massimo**

Considera il valore massimo all'interno della maschera, e vado a sostituire al valore centrale della maschera il valore massimo individuato → erosione delle zone scure, espansione delle zone chiare

Lezione 7, slide 1; 8 – 15; 20 - 21

### **Istogramma cumulato**

Considera i valori di grigio di un'immagine da zero a un dato valore, sommandoli prima di questo limite

### **Equalizzazione dell'istogramma**

Tutti i pixel di un determinato valore vengono trasformati alla stessa maniera; così è possibile, approssimativamente, uniformare i valori spostandoli in blocco per ogni valore; a una barra nell'istogramma originale, deve corrispondere una barra dell'istogramma finale (e quindi i valori nell'istogramma finale si trovano “spostati” rispetto all'istogramma originale).

Bisogna poi creare un operatore che uniformi i livelli di grigio dell'immagine alla stessa altezza

Il valore di ogni singola colonna dell'istogramma originale viene sostituito con il valore corrispondente nell'istogramma cumulato.

Moltiplico il numero di pixel di un dato valore nell'istogramma cumulato per la costante  $L/N$ , dove  $L$  è il numero di livelli di grigio, e  $N$  il numero di pixel dell'immagine, quindi ricostruisco l'istogramma con i nuovi valori

### **Matching**

Livellamento dei livelli di grigio di due immagini per renderle uniformi e poter applicare altri filtri

## LEZIONE 7: 18 marzo 2009

### Lezione 10, slide 1 - 25

Integrare comporta attenuare le alte frequenze → “derivare” il reticolo discreto porta a una maggiore definizione

**Sharpening (filtri passa – alto)** → preferenza per le alte frequenze → vengono esaltate le zone di transizione; la somma algebrica dei nuclei convolutivi è nulla

La derivata si mantiene nulla nelle zone uniformi; picco minimo nel passaggio dal chiaro allo scuro, picco massimo nel passaggio dallo scuro al chiaro; la “rampa” di salita/discesa del segnale può non essere ben definita, creando problemi di individuazione del bordo dell'immagine; la derivata seconda individua i punti di *zero crossing*, dove la rampa di variazione è ridotta al minimo. Le zone uniformi si perdono, le zone di alta frequenza si esaltano; per avere un dato utile, questo operatore va sovrapposto all'immagine originale per recuperare le informazioni perse nelle zone omogenee. L'effetto indesiderato di questi operatori è l'enfatizzazione del rumore (in quanto ad alta frequenza). La funzione  $f(x+1,y) - f(x,y)$  fa riferimento all'*image strip* (la “x + 1” è riferita alla posizione successiva sull'*image strip*).

La derivata prima è nulla solo nei punti uniformi; la derivata seconda lo è anche sulle rampe.

### **Operatore Laplaciano**

- Somma i contributi delle derivate seconde lungo le x e le y
- Sfrutta finestre viaggianti con somma algebrica interna pari a zero
- Per ottenere un risultato soddisfacente, il risultato di questo operatore va sommato al pixel “originale” dell'immagine di partenza, in modo da esaltare il contrasto dei bordi

### **Unsharp Masking**

Sottrazione dell'immagine trattata con un filtro di smoothing dall'immagine originale

### **High Boosting**

Riconferma dell'immagine originale prima di sottrarvi l'immagine filtrata con lo smoothing

### **Gradiente**

È un vettore che punta alla massima pendenza del segnale; è costituito da due componenti: derivata parziale prima lungo l'asse delle x e derivata parziale prima lungo l'asse delle y

Si può semplificare il calcolo eliminando i valori elevati al quadrato e la radice, approssimando infatti non si ottiene una differenza sostanziale

→ **Operatore di Roberts**

→ **Operatore di Sobel** → esaltazione dell'alta frequenza e smoothing del rumore; è necessario usare due maschere (una per le x e un per le y) in quanto ognuna è un *modulo*

## LEZIONE 8: 24 marzo 2009

Lezione 10, slide 26 – 28 (revisione)

Lezione 14, slide 4; 8 – 21; 25 - 42

Isolare un oggetto dallo sfondo.

Esempio: **isolare un singolo pixel**

1. Passare un filtro passa-alto isotropo sull'immagine → isolamento del singolo punto
2. Sogliare opportunamente l'immagine per esaltare solo il punto desiderato

Esistono maschere che esaltano strutture più complesse (linee)? Sì, sono dette **maschere viaggianti di tipo direzionale**

Esaltazione della linea secondo lo schema per cui la linea viene individuata osservando la preponderanza di contrasto lungo una determinata direzione piuttosto che un'altra

Operator **LoG** (Laplaciano su Gaussiano, in un'unica operazione)

Laboratorio HIPR2 – Esercitazione sull'operatore Laplaciano + Gaussiano

## LEZIONE 9: 25 marzo 2009

### Lezione 15, slide 1 - 31

La segmentazione consente di evidenziare o gli elementi di “bordo” o di “omogeneità” per distinguere oggetto e sfondo

**Edge Linking** → considera la direzione del gradiente, se è continua anche su due spezzoni di “bordo” è possibile collegarli; è un metodo comunque che varia da caso a caso, l'operatore deve scegliere al giusto livello, e guardando l'istogramma sceglie opportunamente questi valori

**Thresholding** a una soglia o due soglie

Soglie globali → considerano tutta l'immagine

Soglie locali → considerano anche l'intorno del pixel

Soglie adattive → localizzazione dello specifica area dell'immagine

**Criteri per l'individuazione delle “regioni”:**

- 1.
2. Rispetto del criterio di adiacenza
3. Non devono esserci intersezioni con lo sfondo
4. Ciascuna regione deve essere tale per cui tutti i pixel di una stessa regione abbiano lo stesso valore di grigio
5. Se caratterizzo due regioni con due valori di grigio diversi, unendole il valore di grigio di una delle due non deve essere valido per entrambe

**Region Growing** → fusione di regioni con i pixel adiacenti che soddisfino un dato criterio di omogeneità

**Split & Merge**

**Split** → suddivisione progressiva dell'immagine in zone sempre più piccole per ottenere aree uniformi

**Merge** → riunione di aree adiacenti che soddisfano lo stesso criterio

### Lezione 14, slide 1 – 38 (revisione)

## LEZIONE 10: 26 marzo 2009

Scegliendo opportunamente gli *edge* si riesce a calibrare l'elaborazione dell'immagine non secondo metodi puramente empirici

### Esercizi numerici

#### Esercizio 1: SMOOTHING

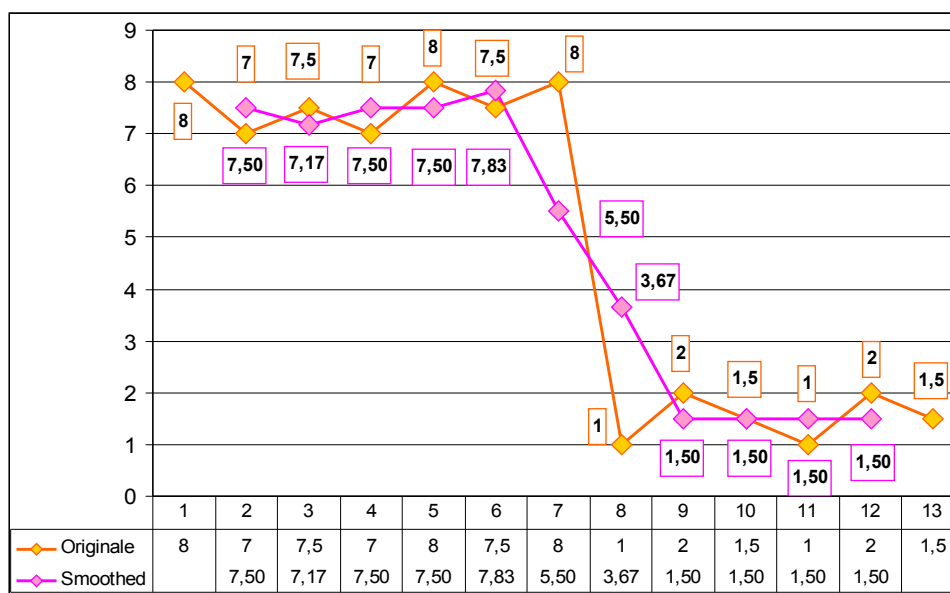
Applicando una maschera di dimensioni 3 x 1 (3 orizzontali, 1 verticale), verranno eliminate la prima e l'ultima colonna. La **media** viene quindi applicata dal secondo al dodicesimo pixel (escludendo il primo ed il tredicesimo).

$M_1 = (8 + 7 + 7,5)/3 = 7,5 \rightarrow$  nuovo insieme di pixel: nullo; 7,5; 7,5

$M_2 = (7,5 + 7,5 + 7)/3 = 7,17 \rightarrow$  nuovo insieme di pixel: 7,5; 7,17; 7

Ecc.

Si noti come nella posizione del potenziale bordo (tra il settimo e l'ottavo pixel) si abbia una rampa molto meno ripida



#### Esercizio 2: TRESHOLDING

Immagine 8 x 8

2	2	2	2	8	8	8	8
2	2	2	2	8	8	8	8
2	2	2	2	8	8	8	8
2	2	2	2	8	8	8	8
2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2

Applicazione di una maschera viaggiante 3 x 3

```
-1 0 1
-1 0 1
-1 0 1
```

I valori della maschera vengono moltiplicati per i valori reali dell'immagine in corrispondenza della maschera; i valori ottenuti vengono sommati tra loro; sostituzione del valore ottenuto dalla somma al valore del pixel centrale

Immagine elaborata

0	0	18	18	0	0	0
0	0	18	18	0	0	0
0	0	18	18	0	0	0
0	0	18	18	18	18	18
0	0	18	18	18	18	18
0	0	0	0	0	0	0

### Esercizio 3: SOBEL

2	2	2	2	8	8	8	8
2	2	2	2	8	8	8	8
2	2	2	2	8	8	8	8
2	2	2	2	8	8	8	8
2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2

Sobel applica due maschere viaggianti in contemporanea, una verticale e una orizzontale

Maschera Orizzontale

-1	-2	-1
0	0	0
1	2	1

Maschera Verticale

-1	0	1
-2	0	2
-1	0	1

I valori della maschera vengono moltiplicati per i valori reali dell'immagine in corrispondenza della maschera; i valori ottenuti vengono sommati tra loro; viene estratto il valore assoluto della somma; sostituzione del valore ottenuto dalla somma dei due valori (dalle x e dalle y) al valore del pixel centrale. Esaltazione dei bordi; riduzione del rumore

0	0	24	24	0	0	...
0	0	24	24	0	0	...
0	0	36	36	0	0	...
...	...	...	...	...	...	...
...	...	...	...	...	...	...
...	...	...	...	...	...	...

### Esercitazione con HIPR2 e ImageJ

## LEZIONE 11: 31 marzo 2009

### Lezione 11, slide 1 - 41

#### **Il colore, percezione e visione del colore**

Il colore non è un fenomeno fisico, ma luminoso

Quando le lunghezze d'onda che vengono percepite dal nostro occhio giungono con la medesima intensità, vengono percepite informazioni di luce monocromatica → **percezione dell'intensità**

Struttura dell'occhio → Coni: percezione del colore, tre tipologie sensibili a determinati range di sensibilità, dette *famiglie*

1. Da 400 a 550 nm, picco a 440 nm (blu – violetto) → 2% dei coni
2. Da 450 a 650 nm, picco a 540 (verde) → 33% dei coni
3. Da 490 a 690 nm, picco a 580 nm (rosso) → 65% dei coni

Percezione della luce acromatica → intensità massima percepita: 555 nm (se non fosse acromatica, sarebbe giallo-verde) → gli abbaglianti e i fendinebbia delle auto hanno frequenze intorno a questo valore per essere maggiormente visibili

Differenza tra **sintesi additiva** (luminosa: Red, Green, Blue) e **sottrattiva** (pigmenti: Cyan, Magenta, Yellow)

CMY sono secondari nella sintesi additiva, mentre sono primari nella sottrattiva

Sistema colorimetrico in XYZ → ogni colore è composto da tre elementi cromatici (rispettivamente di R, G e B) → tecnica ingegneristicamente ottima, ma per l'elaborazione dell'immagine così come viene percepita dall'occhio umano non ha molto senso

**Modelli colore:**

**1. RGB**

Rappresentabile come un cubo, alle cui tre direzioni spaziali corrispondono i tre colori principali Red, Green e Blue. Ciascun segmento sugli assi si normalizza il valore da 0 a 1 (per le 255 variabili di colore di ciascun canale); indicazioni dei tre colori con percentuali (*teoria del tristimolo*) → approssimazione, in quanto questa legge non riesce a indicare ogni singola sfumatura (il *verde visibile* comprende diverse sfumature di verde, mentre quello della formula del tristimolo ha un unico valore definito); ogni vertice sugli assi cartesiani definisce un colore puro (Red, Green, Blue).

Il magenta, ad esempio, ha valore massimo di Red e di Blue, mentre 0 di Green (“tripletta”: 101); il ciano ha “tripletta” 011.

La diagonale del cubo è posizionata in modo da avere valori sempre identici per tutti i canali, quindi è una scala di grigio. Quindi la diagonale del cubo è la scala dell'intensità.

Nella Grafica Web è consigliabile utilizzare un sottoinsieme dei 256 colori, raggruppati come *web safe colors*: <http://www.liconet.com/htmltools/web-safe-colors.html>

**2. CYM**

La conversione da RGB a CYM comporta la definizione di un cubo con le basi opposte a quello RGB →  $C = 1 - R$ ,  $Y = 1 - B$ ,  $M = 1 - G$ .

L'aspetto psicofisico del colore però complica le cose: i colori percepiti nelle combinazioni di RGB e della combinazione CYM non sono perfettamente gli stessi della conversione matematica da RGB a CYM (esempio: modifica del colore dal video alla stampa).

**3. HSI**

Conversione da RGB a HSI → rotazione del cubo RGB, fino ad ottenere un cono in cui l'asse del cono è la diagonale delle intensità del cubo (*intensity*); lungo la circonferenza della “base” del cono si hanno tutti i colori al massimo della propria intensità (*hue*), che può essere un angolo basato su un punto di partenza convenzionale (il rosso); la distanza dal centro della base (dove si trova il bianco) è un'altra variabile, detta *saturation*.

Le variabili che determinano il colore sono quindi la distanza dal vertice del cono (*intensity*), la distanza dal centro della base (*saturation*) e la posizione lungo la circonferenza della base (*hue*).

Il cubo di rappresentazione della *hue* presenta uno stacco netto sulla faccia anteriore, in quanto il colore definito come “base” è il rosso (che si trova proprio sulla faccia anteriore); nel completare il cerchio, lo 0 del rosso “tocca” il 360 del blu-magenta, per questo si ha uno stacco così netto.

Il cubo di rappresentazione della *saturation* presenta il vertice in cui si trova il bianco come nero (in quanto il raggio di distanza dal centro del cerchio è minimo).

Il cubo di rappresentazione dell'*intensity* nasconde la variazione all'interno, sulla diagonale del cubo.

Esercitazione con HIPR2 e ImageJ

### **Colorizzazione dell'immagine in bianco e nero**

L'immagine in bianco e nero viene replicata in tre piani, uno per ogni canale colore (RGB), dopodiché vengono settati i valori in modo che ad un determinato range di grigi corrisponda un determinato colore nella scala RGB.

Ogni piano viene rappresentato in bianco e nero, per cui se, ad esempio, si desidera visualizzare solo il piano di R, se nell'immagine sono evidenziati come colori possibili il rosso (255, 0, 0), il giallo (255, 255, 0) e il bianco (255, 255, 255), l'immagine apparirà completamente bianca (in quanto i valori su questo canale sono sempre al massimo possibile).

Se un sensore è sensibile a più di tre bande, l'operatore deve scegliere quali bande caricare (3 su n); esistono bande più sensibili a determinati oggetti sul terreno, per cui per evidenziare un dato oggetto viene selezionata la banda corrispondente e ignorata un'altra. Dopo questa fase, viene selezionato un *falso colore* per evidenziare gli elementi utili.

### **Come elaborare immagini acquisite a colori**

Immagini recepite con tre sensori, sensibili ognuno a uno dei tre piani colore.

Per elaborarla, la soluzione ideale (anche se non matematicamente perfetta) è elaborare ognuno dei tre piani in modo indipendente con una maschera identica del tipo *operatore puntuale*.

Il tipo di modifica realizzabile dipende dalla codifica dell'immagine: se ho un'immagine codificata in RGB e voglio scurirla, ad esempio, influirò sui tre piani colore; se invece essa è codificata in HSI agirò solo sul canale *intensity*, lasciando *hue* e *saturation* inalterati.

Se voglio modificare la tinta (*hue*), ad esempio per ricolorare l'immagine con i complementari, sfrutterò la *color wheel* imponendo una "rotazione" dei colori di 180°: se sono nella codifica HSI, influirò sugli angoli, in modo che tutti i colori con *hue* da 0° a 180° andranno da 180° a 360°, mentre quelli da 180° a 360° andranno da 0° a 180°; se sono in RGB, invertirò la retta del colore.

**Aumento del contrasto:** funzione quadratica

**Scurirla:** funzione esponenziale (sotto la bisettrice)

**Schiarirla:** funzione logaritmica (sopra la bisettrice)

**Equalizzazione:** aumento del punto di partenza della *saturation* e logaritmica sull'*intensity*

Lavorare su una codifica RGB o HSI potrebbe sembrare lo stesso, in realtà sussistono delle minime differenze (evidenziabili, ad esempio, con una sottrazione tra l'immagine modificata in RGB e quella modificata in HSI).

### Compressione delle immagini

Salvaguardare la qualità riducendo l'occupazione di memoria.

Un criterio di compressione è l'eliminazione delle ridondanze di informazioni → si parte da 3 presupposti:

**1. Ridondanze di codifica** (uso di bit ridondante)

Ha una base probabilistica e non percettiva. Il valore di grigio del pixel è valutato come random. Lo strumento di eliminazione di questo tipo di ridondanza è l'istogramma. Il valore a cui viene normalizzato l'insieme dei valori dipende dalla profondità dell'immagine (se ci sono grigi da 0 a 255, questo valore varrà 8).

Per elaborare l'immagine con questo filtro, viene visualizzato l'istogramma dell'immagine, si contano i pixel di ogni valore di grigio; in un'immagine come nell'esempio (slide 9), i bit sono 3. Moltiplico quindi la percentuale di ogni singola tonalità per il numero di bit. Dopodiché la “lunghezza” dei valori dei bit viene modificata rendendola *variabile*, in modo che ai pixel meno diffusi venga assegnata l'informazione più grande, e a quelli più frequenti le informazioni più ridotte.

**2. Ridondanza psico-visuale** (per la profondità di colore... che però non viene percepita dal nostro occhio)

Se il numero di livelli di grigio viene eccessivamente ridotto, si rischia di visualizzare dei *falsi contorni*. Per evitare questo, è possibile impostare l'algoritmo in modo da considerare anche i pixel adiacenti a quello modificato, in modo da ridurre l'effetto di “stacco” tra livelli.

**3. Ridondanza inter-pixel**

Analisi spaziale dell'immagine, in modo da identificare eventuali regolarità di adiacenze di determinati valori. Viene analizzata la correlazione tra un pixel e se stesso, quindi con quello adiacente, ecc...

Il grafico di un'immagine con ripetizioni significative ha una costituzione regolare.

Se l'immagine rispetta determinati criteri di ripetizione, è possibile trasformare la codifica dell'immagine in modo che quando viene trovato un pixel che ha adiacente un pixel disposto in modo identico a un altro punto precedentemente identificato, questa relazione venga rappresentata da un numero minore di bit rispetto all'intera codifica dell'insieme di pixel.

Esistono due diversi tipi di *codec* di compressione delle immagini:

- **Lossy** → si ha perdita di informazioni
- **Non-Lossy**

**Huffman coding:** i livelli di grigio simili vengono accorpati progressivamente; questo sistema si blocca quando raggiunge i due valori. Quindi si procede a ritroso assegnando un valore binario ad ogni valore (quindi anche a quelli “rimossi” dalla precedente operazione, in quanto memorizzati comunque).

**Sistema aritmetico:** gli intervalli in cui viene splittata l'immagine vengono divisi progressivamente in valori sempre minori, fino ad ottenere un'immagine compressa al meglio.

**LZW (non-lossy):** ogni stringa di pixel che può ripetersi nell'immagine viene identificata da un “indirizzo” di riferimento in un “dizionario”, in modo che all'occorrenza di una data stringa (anche molto lunga), essa possa essere sostituita dall'indirizzo corrispondente, risparmiando molto spazio.

**Formati immagine – A mappa di bit:** BMP, TIFF, PNG, JPEG, GIF...