

ALMA MATER STUDIORUM – UNIVERSITÀ DI BOLOGNA  
SEDE DI CESENA  
SECONDA FACOLTÀ DI INGEGNERIA CON SEDE A CESENA  
CORSO DI LAUREA MAGISTRALE IN INGEGNERIA BIOMEDICA

# SEGMENTAZIONE AUTOMATICA DI REGIONI IN IMMAGINI ISTOLOGICHE

Tesi in

**Bioimmagini e Visione LM**

**Relatore**

Prof. Alessandro Bevilacqua

**Presentata da**

Marco Marchetti

**Co-Relatore**

Ing. Alessandro Gherardi

Ing. Filippo Piccinini

Dott. Wainer Zoli

Sessione Terza

Anno Accademico 2010-2011



# Sommario

INTRODUZIONE.....	7
-------------------	---

## CAPITOLO 1

ANALISI ISTOLOGICA .....	11
1.1 Il tumore.....	11
1.2 Istologia .....	13
1.3 Trattamento del campione .....	15
1.4 Ematossilina ed Eosina (H&E) .....	17
1.5 Microscopia.....	18
1.6 Microscopia in campo chiaro.....	22
1.7 Immagini digitali.....	23

## CAPITOLO 2

ELABORAZIONE DI IMMAGINI.....	27
2.1 Istogramma .....	28
2.2 Elaborazioni globali .....	30
2.2.1 Aumento del contrasto .....	30
2.2.2 Correzione gamma .....	32
2.2.3 Equalizzazione dell'istogramma .....	34
2.3 Elaborazioni Locali .....	35
2.3.1 <i>Sharpening</i> .....	36
2.3.2 Riduzione del rumore .....	37
2.4 Segmentazione .....	39
2.4.1 Sogliatura globale dell'istogramma .....	42
2.4.2 Sogliatura locale dell'istogramma.....	44

---

2.4.3 <i>Region growing</i> .....	45
2.4.4 <i>Watershed</i> .....	46
2.4.5 <i>K-means</i> .....	47
2.5 <i>Texture analysis</i> .....	50
2.6 Il colore .....	52
2.6.1 L'occhio .....	53
2.6.2 Spazio XYZ.....	55
2.6.3 Spazio Lab.....	57

## CAPITOLO 3

STATO DELL'ARTE.....	63
----------------------	----

## CAPITOLO 4

APPROCCIO AL PROBLEMA DI CLASSIFICAZIONE.....	73
---	----

4.1 Il colore come <i>feature</i> .....	75
4.2 Utilizzo dell'istogramma nel processo di classificazione.....	75
4.2.1 Confronto tra istogrammi .....	76
4.2.2 Misura di similarità tra istogrammi.....	80
4.3 Scansione dell'immagine .....	81

## CAPITOLO 5

ALGORITMO E IMPLEMENTAZIONE.....	83
----------------------------------	----

5.1 Definizione dei parametri .....	84
5.2 Modifica dello spazio colore.....	87
5.3 Estrazione delle feature.....	88
5.4 Classificazione .....	91
5.4.1 Funzionamento dell'algoritmo <i>k-means</i> .....	92
5.5 Salvataggio e visualizzazione .....	93

## CAPITOLO 6

ESPERIMENTI.....	97
------------------	----

6.1 Performance dell'algoritmo .....	97
6.2 Il parametro <i>nbin</i> .....	99
6.3 <i>Testbed</i> .....	103
6.4 Validazione della classificazione.....	103
6.5 Guida al test .....	105

---

6.6 Presentazione delle regioni identificate .....	114
6.7 Esempio di classificazione.....	115
6.8 Risultati del test .....	117

## **CAPITOLO 7**

<b>CONCLUSIONI.....</b>	<b>127</b>
-------------------------	------------

<b>BIBLIOGRAFIA.....</b>	<b>131</b>
--------------------------	------------



# Introduzione

Il progresso tecnologico e i nuovi strumenti di analisi medica hanno contribuito in maniera sostanziale alla diminuzione dei tempi di attesa nel processo di diagnosi di varie patologie. In particolare, in oncologia l'aumento del numero degli esami di screening associato alla drastica diminuzione dei tempi di diagnosi ha contribuito in maniera determinante alla diminuzione dei decessi. Nel campo della medicina il raggiungimento di diagnosi precoci, attraverso l'analisi microscopica, tempestiva ed immediata, di preparati istologici, risulta quindi essere un costante obiettivo primario.

In ambito clinico/ospedaliero, le unità di istologia patologica svolgono un ruolo fondamentale per la pianificazione di eventuali terapie mediche e/o chirurgiche, fornendo diagnosi su tessuti o cellule prelevate da pazienti in cui si sospetta una malattia. Tuttavia, questo tipo di analisi non viene effettuato solo a seguito di sospette patologie ma, nella pratica, ogni qualvolta viene effettuato un intervento chirurgico viene anche richiesta una analisi di campioni istologici. In particolare, la comune indagine istopatologica ha come obiettivo primario la distinzione e il rilevamento nei tessuti analizzati di cellule normali, infiammate, tumorali (distinguendo, inoltre, tra tumori benigni e maligni) ed altre condizioni patologiche. La figura professionale istruita per poter svolgere l'indagine è l'istopatologo, un medico specializzato nella discriminazione di tutte le differenti regioni presenti sul campione, a partire anche da una ridottissima porzione di

tessuto. In particolare, dopo aver prelevato dal paziente una porzione di tessuto da analizzare, si procede all'osservazione al microscopio di sottili sezioni di esso, trattate e colorate al fine di enfatizzare regioni e strutture.

Nonostante il referto dell'istopatologo sia, ad oggi, l'unico riferimento per l'analisi dei campioni, al fine di ottenere analisi oggettive e ripetibili risulta molto interessante sfruttare la possibilità di connettere ai microscopi dispositivi di acquisizione immagini, come telecamere o macchine fotografiche. Questo consente di ottenere immagini digitali che possono essere salvate, elaborate ed analizzate in maniera automatica, applicando tecniche note e consolidate, o sviluppando metodi specifici. A tal proposito, un notevole interesse è rivolto a quelle tecniche di elaborazione delle immagini che permettono una migliore visualizzazione in termini qualitativi, come ad esempio l'aumento del contrasto, ma che forniscono anche un valido strumento per l'analisi quantitativa di specifiche caratteristiche morfologiche degli oggetti presenti all'interno dell'immagine, come ad esempio perimetro e area, valutate in misura assoluta attraverso opportune conversioni pixel-micrometri.

In tale contesto, il presente lavoro di tesi ha avuto come obiettivo lo studio e l'implementazione di un opportuno metodo di segmentazione automatica volto a costituire uno strumento software semplice da utilizzare ed in grado di assistere l'istopatologo nell'identificazione di regioni percettivamente simili, presenti all'interno dell'immagine istologica. Il metodo sviluppato permette di elaborare un'ampia varietà di immagini istologiche e di classificarne le regioni esclusivamente in base al contenuto, senza sfruttare alcuna conoscenza a priori riguardante il tessuto biologico analizzato. Questa caratteristica permette di ottenere un comportamento dell'algoritmo estremamente generale e versatile che risponde ai requisiti di applicabilità in ambito ospedaliero. Le tecniche applicate alle immagini istologiche proposte in letteratura sono volte alla ricerca di caratteri patologici noti. Questa metodologia può risultare efficace qualora le immagini da analizzare provengano dallo stesso sistema di acquisizione e si indaghi un unico tipo di tessuto biologico. Tale condizione non è però quasi mai soddisfatta poiché

le analisi vengono condotte con la strumentazione presente nello specifico laboratorio. L'impossibilità di standardizzare i mezzi per l'acquisizione di immagini in tutti i laboratori implica la necessità di poter avere a disposizione software di carattere generale in grado, però, di lavorare in maniera specifica nei diversi contesti. Indagare ed individuare automaticamente regioni che possono essere considerate simili consente di escludere le aree non interessanti da analizzare rendendo più veloce ed affidabile il lavoro manuale, ma soprattutto consente di poter effettuare su queste regioni successive analisi automatiche. L'estrema generalità e l'impiego di misure statistiche sulle quali definire la classificazione ha il notevole vantaggio di ottenere risultati ripetibili e robusti anche al variare della risoluzione dell'immagine.

Il seguente elaborato è così suddiviso:

Il Capitolo 1 introduce e affronta in modo generale gli aspetti fondamentali dell'istologia patologica, descrivendo gli stadi fondamentali a cui è sottoposto il provino istologico in esame, quali prelievo, trattamento, colorazione analisi. Successivamente, vengono elencate le caratteristiche principali della microscopia "digitale", attraverso la quale vengono ottenute le immagini che saranno in seguito elaborate.

Il Capitolo 2 è volto alla trattazione delle tecniche più comunemente utilizzate nell'ambito dell'elaborazione delle immagini istologiche e fornisce i concetti di istogramma e spazio colore che saranno poi sfruttati nei capitoli successivi.

Il Capitolo 3 offre una panoramica su come le tecniche descritte nel capitolo precedente vengano impiegate al giorno d'oggi, ponendo l'attenzione sui vantaggi e sulle loro condizioni di applicazione.

Nel Capitolo 4 si descrive l'approccio seguito per pervenire all'algoritmo di classificazione e la teoria per supportarlo.

Il Capitolo 5 fornisce la descrizione implementativa dell'algoritmo offrendo una spiegazione dettagliata della piattaforma utilizzata e delle principali funzioni.

Nel Capitolo 6 sono esposti e discussi i risultati ottenuti con l'algoritmo realizzato.

Nel Capitolo 7 vengono riassunti i punti di forza e le caratteristiche innovative dell'algoritmo, discutendone anche i limiti e fornendo indicazioni per futuri miglioramenti.

# Capitolo 1

## Analisi istologica

L'istologia è la scienza che studia la struttura dei tessuti e le cellule di cui sono composti, sia da un punto di vista morfologico che funzionale, partendo da sottili fette di campioni tissutali fissati su vetro. L'analisi istologica, recentemente, è diventata un valido strumento per diagnosticare un vasto numero di malattie tra cui: carcinoma mammario, neoplasie cerebrali e tumori dei tessuti molli, e per tale motivo si ritiene necessario dare una breve descrizione di che cosa si intende per tumore.

### 1.1 Il tumore

In patologia, un tumore (dal latino tumor che significa rigonfiamento) o neoplasia (dal greco néos, nuovo, e plásis, formazione) è, nella definizione dell'oncologo R.A. Willis (accettata a livello internazionale), «una massa anormale di tessuto che cresce in eccesso ed in modo scoordinato rispetto ai tessuti normali, e persiste in questo stato dopo la cessazione degli stimoli che hanno indotto il processo» [1]. In particolare, un tumore maligno viene anche identificato con il termine di cancro.

---

La crescita incontrollata e caotica di un gruppo di cellule Fig. 1.1, a scapito dell'omeostasi tissutale, è determinata da alterazioni del patrimonio genetico, ed è alla base di una vasta classe di malattie, classificate principalmente seguendo tre diversi parametri:

istotipo originario delle cellule proliferanti (principalmente in tumori epiteliali, mesenchimali, delle cellule del sangue e del tessuto nervoso),

aggressività e decorso clinico previsto,

stadiazione tumorale, o classificazione TNM (per quanto riguarda i tumori maligni).

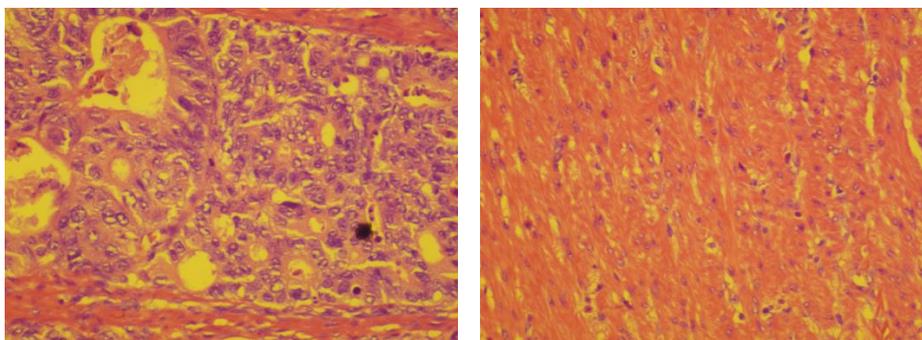


Fig. 1.1 Confronto tra immagini ottenute su di un tessuto tumorale ed uno sano

Quando il sistema di controllo della riproduzione di una cellula accumula una serie di danni irreversibili, si verifica tipicamente una degenerazione tumorale della stessa verso l'istotipo tumorale. Più precisamente, tutte le cellule cancerose e precancerose presentano alterazioni, spesso molto estese, del loro assetto cromosomico (cariotipo): il numero di cromosomi presenti nel loro nucleo può essere alterato e i cromosomi stessi possono essere danneggiati, moltiplicati o mancanti. Quest'ultima osservazione, fatta all'inizio del XX secolo da David von Hansemann e Theodor Boveri, fu la base della teoria dei tumori come "malattie cromosomiche" ed accettata in medicina, fino alla scoperta, a metà del secolo scorso, della struttura del DNA e del meccanismo delle mutazioni genetiche. L'alterazione cromosomica delle cellule tumorali solitamente è talmente estesa da

---

fornire la prova che tutte le cellule cancerose discendano da un'unica cellula madre mutata, in quanto condividono la stessa esatta tipologia di alterazione genetica. L'elevata complessità del danno genetico rende altamente improbabile l'eventualità della presenza di due cellule madri diverse che hanno subito, per caso, la stessa identica serie di mutazioni. Nonostante il meccanismo generale di origine dei tumori sia unico, questi possono manifestare una gamma molto vasta di evoluzioni e di sintomatologie. Denominatore comune a tutti i casi è un elevato aumento del numero di cellule prodotte, imputabile alla maggiore velocità di riproduzione cellulare. In genere la crescita di un tumore segue una legge cinetica con tempistiche all'inizio lente, ma caratterizzate da una progressiva accelerazione all'aumentare della massa del corpo. La dimensione critica di un tumore alla quale il tumore inizia a crescere molto velocemente e a dar luogo a primi sintomi problematici, è di circa 1 cm<sup>3</sup>.

Al giorno d'oggi grazie allo sviluppo tecnologico e ad analisi entrate a far parte dei comuni esami di diagnostica medica (PET, SPECT, analisi di markers tumorali presenti nel sangue, etc.), è possibile diagnosticare con alta accuratezza l'eventuale presenza di cellule tumorali. Spesso però i sintomi iniziali vengono trascurati o sottovalutati e le analisi diagnostiche vengono condotte tardivamente per poter intervenire efficacemente ed in maniera tempestiva. In questo ambito di notevole importanza è l'analisi dei tessuti, nota come istologia.

## 1.2 Istologia

L'istologia può essere suddivisa in due branche specifiche: l'istopatologia e la citopatologia, Fig. 1.2.

L'istopatologia pone la sua attenzione sull'alterazione d'interesse porzioni di tessuto, al fine di classificare eventuali patologie. La citopatologia, invece, esamina singole cellule analizzandone la microstruttura, funzione e chimica, per ottenere informazioni riguardanti il tessuto e la crescita cellulare.

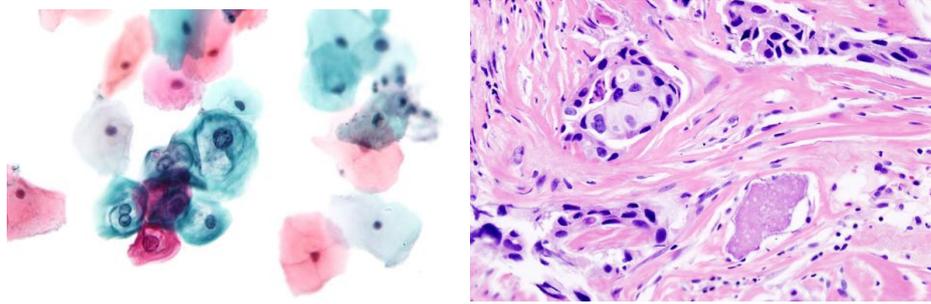


Fig. 1.2 A sinistra un provino citologico a destra uno istologico - Tratto da Wikipedia

Entrambe le procedure richiedono l'identificazione di oggetti troppo piccoli per essere osservati ad occhio nudo e perciò è necessario l'impiego di microscopi. Tuttavia, prima della visualizzazione occorre trattare il campione, principalmente per impedire la degradazione ed enfatizzare le strutture di interesse Fig. 1.3. A tal fine viene eseguito un protocollo che comprende diverse fasi quali: fissazione, disidratazione, inclusione, taglio, colorazione e montaggio.

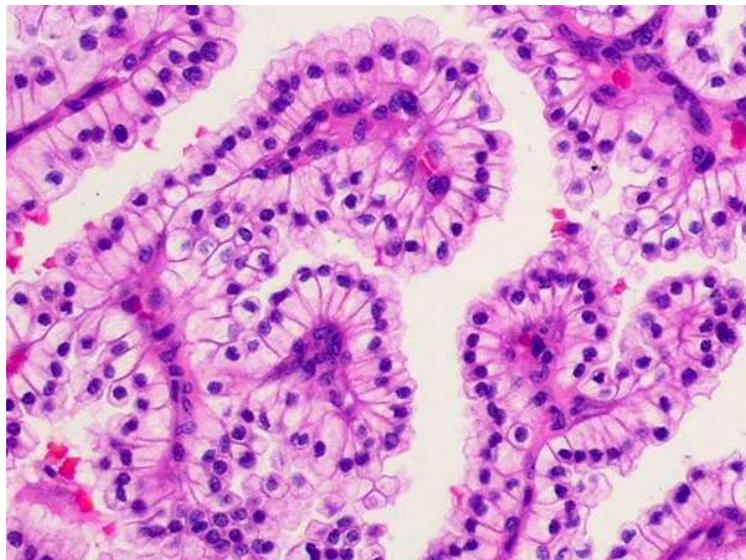


Fig. 1.3 Vetrino istologico di carcinoma renale

---

### 1.3 Trattamento del campione

La fissazione ha lo scopo di rendere stabile il tessuto, altrimenti soggetto a fenomeni di autolisi e decomposizione, e si realizza impiegando numerose sostanze come alcol, formalina, acido acetico glaciale, sublimato, acido picrico, acido osmico, etc. Salvo casi specifici, si preferisce l'impiego di una miscela di sostanze al fine di attenuare gli inconvenienti che ciascun componente chimico, usato singolarmente, potrebbe produrre sui tessuti trattati. La disidratazione è necessaria per eliminare l'acqua presente nel tessuto, in modo da renderlo compatibile con il mezzo d'inclusione, questo processo prevede una serie di passaggi del preparato in alcoli a gradazione progressivamente maggiore.



Fig. 1.4 Macchina automatica per la disidratazione, fissazione e impregnazione in paraffina dei campioni di tessuto - Tratto da Wikipedia

L'inclusione è un procedimento che permette al tessuto di acquistare una consistenza omogenea adatta al taglio e si realizza facendo impregnare il tessuto di sostanze indurenti e ad inclusione avvenuta si effettua il taglio del preparato. Nel caso si debba procedere all'osservazione al microscopio ottico si impiega un microtomo, dal quale si ottengono sezioni di tessuto di pochi micrometri di spessore che vengono successivamente collocate su particolari vetrini. La colorazione del campione avviene impiegando prodotti chimici naturali

---

(ematossilina, eosina, carminio, etc.) o sintetici (come i coloranti anilini). La colorazione permette di evidenziare le strutture dei tessuti facilitando l'osservazione del preparato. Il montaggio consiste nel coprire il campione con un vetrino che viene sigillato con un altro vetrino sottostante, usando apposite resine naturali o sintetiche.

Si ottengono così preparati stabili, che possono essere conservati ed osservati anche molti anni dopo Fig. 1.5.

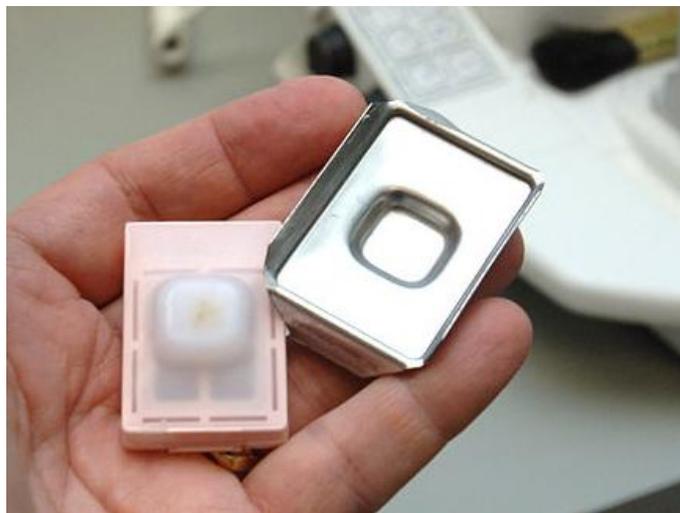


Fig. 1.5 Tessuto incluso nella paraffina pronto per il taglio al microtomo - Tratto da Wikipedia

---

## 1.4 Ematossilina ed Eosina (H&E)

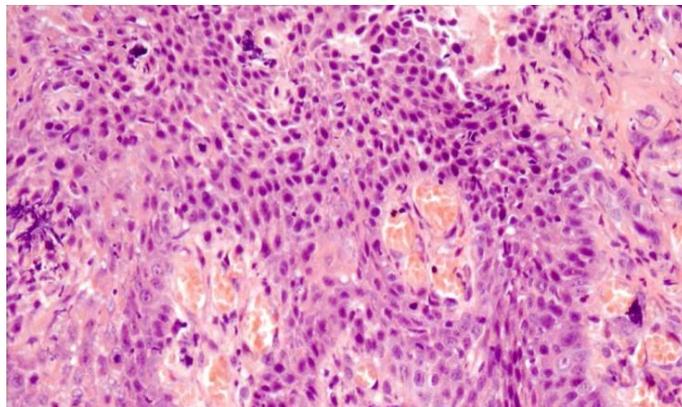


Fig. 1.6 Immagine di cellule colorate con H&E – Tratto da [2]

La colorazione è solo uno dei tanti passaggi che deve subire il campione prima di essere analizzato, ma sicuramente è uno dei più importanti al fine di ottenere una visualizzazione dei tessuti su cui stilare il referto medico. Ematossilina ed eosina (in inglese Hematoxylin and Eosin, H&E) è considerato il gold standard per la colorazione in analisi istopatologica (Fig. 1.6 rappresenta una comune immagine colorata con H&E). E' comunemente usato in istologia e negli esami istopatologici di routine, poiché rende evidenti particolari strutture come nuclei e citoplasma, con un conseguente aumento del contrasto degli oggetti di interesse nelle immagini del campione.

Generalmente, si dice che l'ematossilina sia un colorante basico che si lega a molecole acide (e quindi dotate di basofilia, cioè di affinità per molecole basiche) conferendo a esse un colore blu violaceo. In realtà, le proprietà tintoriali dell'ematossilina dipendono da reazioni chimiche complesse. Semplicemente l'ematossilina di per sé non è un colorante ma lo diventa a seguito di ossidazione in ematina, ottenuta durante la preparazione della soluzione di colorazione. Quest'ultima non è in grado di legarsi a molecole biologiche a meno che non cheli, cioè formi un legame covalente e uno coordinato con un metallo, ad esempio alluminio (allume) o ferro. Lo ione metallico è chiamato mordenzante e

---

ha la fondamentale funzione di legare l'emateina ai gruppi fosfato degli acidi nucleici. Nuclei, ribosomi e reticolo endoplasmatico rugoso hanno quindi una forte affinità per questo colorante a causa del loro elevato contenuto in DNA e RNA, rispettivamente. L'eosina (di solito Eosina-Y) è invece una molecola acida che colora le strutture basiche (acidofile) in rosso/rosa. La maggior parte delle proteine citoplasmatiche è di tipo basico, e quindi il citoplasma, a contatto con eosina, si colora di rosa o rosso chiaro e diviene facilmente distinguibile. In generale quindi, i tessuti colorati con H&E sono caratterizzati da un buon contrasto e le cellule hanno i nuclei di colore blu e il citoplasma di colore rosa o rosso. Senza l'utilizzo di coloranti non sarebbe possibile eseguire nessun tipo di analisi a causa del basso contrasto tipico dei tessuti biologici Fig. 1.7.

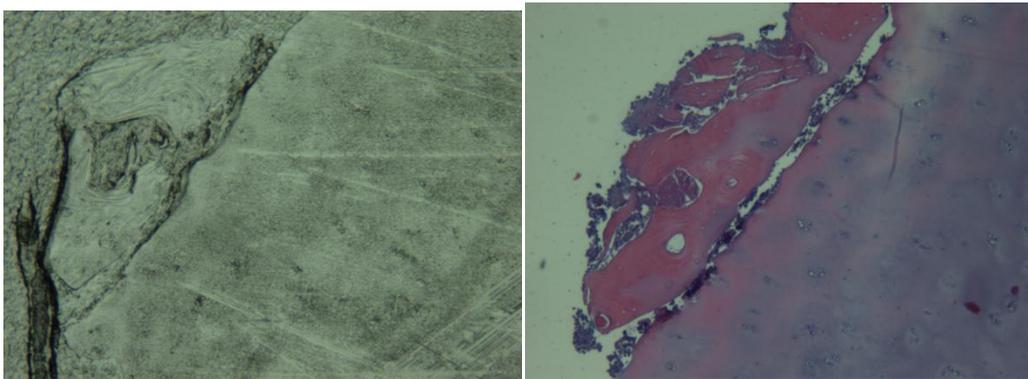


Fig. 1.7 Confronto tra tessuto originale e colorato con H&E

## 1.5 Microscopia

Per l'analisi accurata di biopsie, ovvero piccole porzioni di tessuto estratte dall'organo in esame, risulta fondamentale l'utilizzo di un microscopio: le cellule, anche se colorate, risultano troppo piccole e poco nitide per essere osservate esclusivamente ad occhio nudo. L'occhio umano, infatti, riesce a distinguere due oggetti come separati, se questi distano al massimo circa 100  $\mu\text{m}$ , mentre il diametro tipico delle cellule indagate è di circa 30  $\mu\text{m}$ .

---

Al giorno d'oggi è possibile sfruttare una vasta quantità di microscopi, caratterizzati da differenti ingrandimenti e tecnologie per l'acquisizione delle immagini. I microscopi possono essere suddivisi in base alla sorgente "luminosa" utilizzata: i microscopi ottici utilizzano fotoni, i microscopi elettronici elettroni. I microscopi elettronici a loro volta, possono essere distinti in base al tempo di acquisizione dell'immagine in microscopi a trasmissione e microscopi a scansione. Nei primi, il campione di spessore compreso tra i 5 nm e i 500 nm, viene attraversato verticalmente da un fascio di elettroni, e successivamente grazie ad un insieme di magneti, che funzionano come le lenti del microscopio ottico, si ottiene un'immagine ingrandita proiettata tipicamente su uno schermo fluorescente. Nei secondi invece l'immagine è ricavata scannerizzando con un fascio di elettroni estremamente collimato l'oggetto, che può essere anche relativamente spesso. In particolare è importante sottolineare che quest'ultima tecnologia può essere utilizzata anche per ricavare immagini tridimensionali.

Il potere risolutivo di un microscopio è limitato dalla lunghezza d'onda della radiazione elettromagnetica impiegata come sorgente luminosa per analizzare il campione: più piccola è la lunghezza d'onda maggiore è la risoluzione. La risoluzione di un microscopio è definita come la distanza minima alla quale due punti risultano distinti. Nel caso lo strumento utilizzi una radiazione elettromagnetica con una propria lunghezza d'onda associata, come i tradizionali microscopi ottici, risoluzione e lunghezza d'onda sono parametri tra loro strettamente correlati. In prima approssimazione, e non tenendo conto di aberrazioni ottiche, possiamo considerare che la relazione che lega il potere risolutivo "d", l'apertura numerica "AN" e la lunghezza d'onda della radiazione utilizzata "λ" sia:

$$d = \frac{\lambda}{2 * A_N} \quad 1.1$$

Questa relazione è generalmente nota come principio di Abbe. Il microscopio elettronico, utilizzando un fascio di elettroni, permette invece di ottenere una

---

risoluzione di parecchi ordini di grandezza superiore, questo perché agli elettroni è possibile associare una lunghezza d'onda data dall'equazione:

$$\lambda = \frac{h * v}{E} \quad 1.2$$

dove h è la costante di Plank, v la frequenza ed E l'energia associata al fascio di elettroni. La possibilità di diminuire la lunghezza d'onda aumentando E permette di ottenere risoluzioni fino a quattro ordini di grandezza in più rispetto ai microscopi ottici.

Nella pratica i campioni istologici vengono solitamente esaminati con un microscopio ottico sia per ragioni di semplicità che di economicità della strumentazione, quasi due ordini di grandezza più economici dei microscopi elettronici.

Il microscopio ottico (Fig. 1.8) rappresenta il mezzo diagnostico probabilmente più impiegato dagli istopatologi per la refertazione di preparati. Di seguito è riportata una breve descrizione della struttura di un microscopio. Infatti, indipendentemente dalle caratteristiche tecniche specifiche, tutti i microscopi ottici sono costituiti da tre elementi fondamentali: oculare, obiettivo e condensatore.

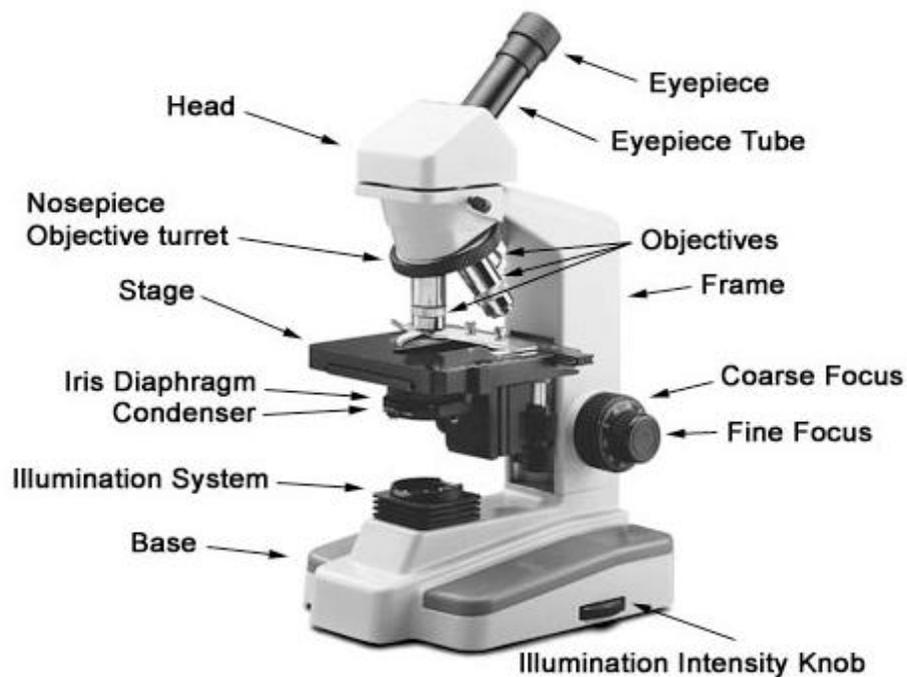


Fig. 1.8 Microscopio ottico

L'oculare ha la funzione di ingrandire l'immagine che si forma nel piano focale dell'obbiettivo e per questo è posizionato in modo che i loro assi coincidano e facendo sì che i loro fuochi si sovrappongano. L'obbiettivo rappresenta l'elemento più critico ed importante del microscopio poiché è soprattutto la qualità delle sue lenti che condiziona le prestazioni ed il valore dello strumento. L'obbiettivo è rappresentato da un doppio sistema di lenti convergenti a corta focale che proietta l'immagine ingrandita ed invertita dell'oggetto sotto osservazione nel piano focale inferiore dell'oculare in modo che questo possa "vederla" ed ingrandirla ulteriormente. La distanza di lavoro di un obbiettivo, ovvero la distanza della lente frontale dal campione, è piuttosto piccola variando dai 30 mm a 0,1 mm a seconda del suo potere di ingrandimento. La relazione che intercorre tra distanza di lavoro ed ingrandimento è inversamente proporzionale: maggiore è il potere di ingrandimento minore è la distanza di lavoro dell'obbiettivo. Ogni obbiettivo porta sulla sua montatura alcune sigle che descrivono le potenzialità, i limiti e le istruzioni d'uso per ogni particolare

---

obbiettivo. Generalmente tutti i microscopi impiegati in campo biomedico hanno, nel loro corredo di ottiche anche un obiettivo "ad immersione"; si tratta in generale di obiettivi dotati di forte potere di ingrandimento del campione e la sostanza in cui vengono immersi è quasi invariabilmente olio minerale per microscopia, per cui portano scritto, sulla loro montatura. "oil" (oppure "oel" o ancora "HI") e sono in genere contrassegnati da una riga di colore bianco o nero. Più specificamente si definiscono "ad immersione" quegli obiettivi che possono essere usati unicamente interponendo una goccia di liquido (in genere olio minerale) fra la loro lente frontale ed il vetrino coprioggetto del campione.

Il condensatore è un sistema di lenti posto al di sotto del tavolino traslatore del microscopio e la sua funzione è quella di convogliare la radiazione elettromagnetica sul campione, cosicché ogni punto di esso risulti essere illuminato e la sua immagine possa essere vista nitidamente attraverso l'oculare. Per ottenere questo, è necessario che vengano soddisfatte due condizioni fondamentali:

regolazione perfetta in altezza del condensatore, pena la comparsa di distorsioni cromatiche e la degradazione generale della nitidezza dell'immagine trasmessa;  
chiusura opportuna del diaframma di apertura in rapporto all'apertura numerica dell'obiettivo in uso (pena la diminuzione del potere di risoluzione e la formazione di immagini di qualità scadente).

## 1.6 Microscopia in campo chiaro

Nel microscopio a campo chiaro (in inglese *Bright Field Microscopy*) il tessuto, montato sul vetrino e fissato dal copri vetrino, viene attraversato da un fascio luminoso che può provenire da sopra o sotto il campione a seconda che si stia utilizzando un microscopio rovesciato o dritto, rispettivamente. Tramite un sistema di lenti il fascio viene proiettato all'oculare o all'ingresso di una telecamera in modo che l'osservatore possa vedere l'immagine ed eventualmente

---

salvarla per successive analisi. In queste condizioni gli oggetti appaiono scuri su sfondo chiaro, in maniera proporzionale alla propria densità Fig. 1.9.

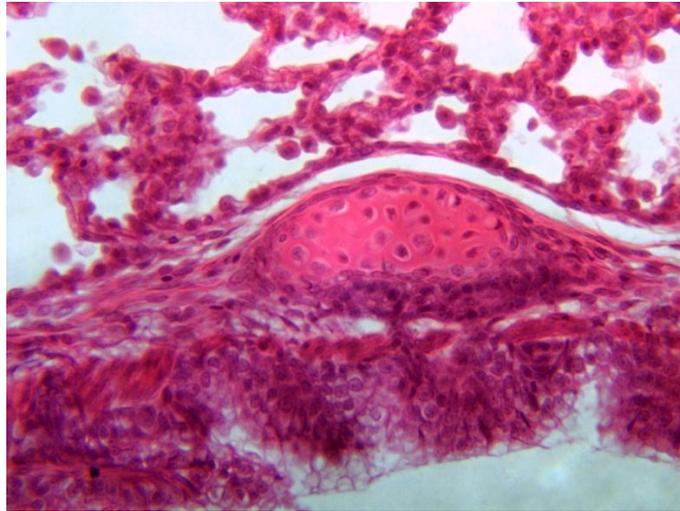


Fig. 1.9 Immagine acquisita con un microscopio a campo chiaro

Questo tipo di microscopio è il più semplice, sia in termini di complessità dei componenti impiegati, sia nel processo di acquisizione e formazione dell'immagine. Tuttavia il limite di tale tecnologia si riscontra nel modesto contrasto degli oggetti biologici visualizzati, che dipende dal rateo di assorbimento del tessuto. Questa restrizione è in parte superata colorando il campione sotto esame, in modo da aumentare il contrasto ed individuare meglio i dettagli. Purtroppo il processo di colorazione porta sempre alla morte del tessuto. Nonostante ciò la microscopia a campo chiaro è la tecnica più frequentemente usata in istopatologia per la diagnosi dei tessuti e la colorazione dei campioni è necessaria per una corretta ed esaustiva analisi del campione.

## 1.7 Immagini digitali

L'immagine digitale può essere definita come una matrice di numeri interi non negativi, ciascuno dei quali può essere rappresentato all'interno di una casella

---

definita "*picture element*" o *pixel*. L'immagine digitale può cioè essere vista come una griglia di pixel all'interno dei quali è allocato un numero che esprime il valore del parametro considerato in quel punto. Geometricamente il pixel assume la forma di un piccolo quadrato in cui la gradazione di grigio o la tonalità di colore riprodotta è uniforme. In termini generali, le immagini computerizzate (sia digitali che digitalizzate) sono quindi ottenute tramite l'attribuzione di valori numerici discreti ad ogni pixel di cui è composta la matrice. Perciò l'immagine digitale risultante è data da un insieme bidimensionale di numeri interi  $f(x, y)$ , dove  $x$  identifica le righe e  $y$  le colonne all'interno di una matrice. Il numero delle righe e delle colonne è rilevante per la definizione della qualità dell'immagine: infatti tanto più ampia è la matrice originale, tanto maggiormente l'immagine computerizzata si avvicina alla realtà (nel caso delle immagini digitali) o all'immagine analogica originale (nel caso delle immagini digitalizzate). Nelle immagini digitali, il livello di grigio associato al pixel varia in funzione del colore degli oggetti che compongono il campione studiato.

Ad ogni *pixel* è associato un valore che rappresenta il numero di fotoni che raggiungono il sensore della fotocamera codificato in 2, 8, 16, 32 o 64 bit, Fig. 1.10. In caso di codifica a 2 bit si parla di immagini in bianco e nero, in cui il valore dei pixel dell'immagine possono assumere solo i valori 0 oppure 1. Con codifiche maggiori si intendono immagini a toni di grigio o a colori, quest'ultime sono rappresentate da tre immagini a scala di grigio dove ognuna rappresenta un canale specifico.

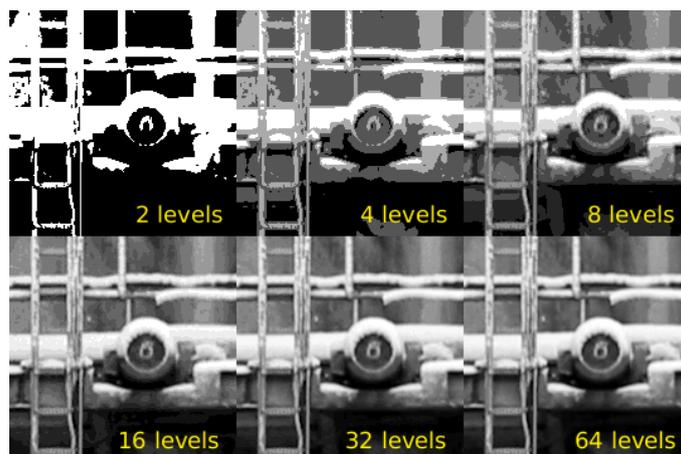


Fig. 1.10 Esempio di immagine a scala di grigio a differenti codifiche

Il colore è quindi codificato da una tripletta di valori in un opportuno spazio colore. Il numero di bit di codifica dei toni di grigio rappresenta il *range dinamico* dell'immagine o in termini radiologici la sua "risoluzione di contrasto".

L'impiego di immagini digitali presenta innumerevoli vantaggi quali l'archiviazione digitale e la loro elaborazione come dato. Le immagini analogiche infatti possono essere "elaborate", cioè modulate in funzione di una precisa finalità diagnostica, solo prima della loro acquisizione. Nel caso delle immagini digitali invece l'elaborazione può essere eseguita anche dopo l'acquisizione. In particolare le trasformazioni possibili delle immagini digitali possono riguardare sia la visualizzazione che l'analisi. Le funzionalità di visualizzazione comprendono: variazione dei livelli di grigio '*contrast enhancement*', inversione della scala dei grigi '*image reverse*', ingrandimento di particolari '*zooming*' e riduzione del rumore '*filtering*'. Tecniche di analisi comprendono misurazioni sull'immagine che possono interessare sia il valore dei pixel per determinare una particolare proprietà dell'oggetto o indagare delle distanze eseguendo una conversione metro/pixel. Grazie all'utilizzazione di questo insieme di tecniche, le immagini digitali permettono di estrarre dati quantitativi e aumentare la fruizione delle immagini.



# Capitolo 2

## Elaborazione di Immagini

Con in termine *Elaborazione di Immagini* si intende una vasta gamma di operazioni che è possibile eseguire su immagini digitali principalmente per migliorarne le qualità visive o per estrarre da essa informazioni salienti (che potrebbero essere rappresentate da un'altra immagine, in cui le sezioni di interesse sono tipicamente meglio definite e contrastate) o dati quantitativi da poter utilizzare per successive ulteriori analisi. Nel primo caso si parla di *Image Enhancement*, di cui si tratterà ampiamente in seguito, mentre nel secondo caso di *Image Analysis*.

Prima di procedere con l'elaborazione vera e propria occorre prestare attenzione alla fase preliminare, definita *pre-processing*, il cui scopo principale è quello di aumentare il rapporto segnale rumore dell'immagine. È infatti possibile, tramite opportune tecniche, correggere in parte alcuni errori dovuti al sistema di acquisizione quali rumore (trattato attraverso operazioni di filtraggio) e distorsioni geometriche o radiometriche (entrambe spesso trattate tramite calibrazioni). Tuttavia, nell'ambito del presente lavoro di Tesi questo aspetto non è stato trattato, in quanto la qualità delle immagini acquisite era sufficiente per gli scopi preposti

---

Terminata la fase di *pre-processing*, si procede con l'elaborazione vera e propria delle immagini, che può essere automatizzata oppure guidata da un operatore umano.

Prima di procedere con l'esplicazione di alcune tecniche comunemente usate per migliorare l'immagine, si ritiene necessario fornire la definizione di "istogramma", un concetto di cui molte procedure, descritte in seguito, fanno uso.

## 2.1 Istogramma

L'istogramma è uno degli strumenti più semplici ed utili nell'elaborazione delle immagini. E' caratterizzato da un elevato contenuto informativo che può essere sfruttato per migliorare diverse caratteristiche per l'analisi dell'immagine: solitamente le operazioni sugli istogrammi precedono elaborazioni più complesse ed evolute.

Operativamente, l'istogramma è la rappresentazione grafica della quantità di *pixel* presenti nell'immagine per ciascun livello di grigio. Sull'asse delle ascisse sono riportati i *bin* dei valori di livello di grigio, mentre sull'asse delle ordinate è riportato il numero di occorrenze (numero di conteggi) per ogni *bin*. In altre parole, normalizzando il valore di ciascun *bin* per la somma totale dei pixel, l'istogramma diventa una funzione discreta di densità di probabilità che rappresenta la frequenza di ciascun livello di grigio nell'immagine.

L'istogramma viene memorizzato in un vettore di dimensione pari al numero dei livelli di grigio dato dalla codifica dell'immagine (2, 4, 8, etc.,  $n$  bit), o comunque nel numero di livelli in cui si sceglie di rappresentare l'immagine. Ciascun elemento del vettore dato rappresenta quindi il numero di pixel aventi livello di grigio pari all'indice dell'elemento. In Fig. 2.1 si vede un'immagine, sulla sinistra ed il relativo istogramma, a destra.

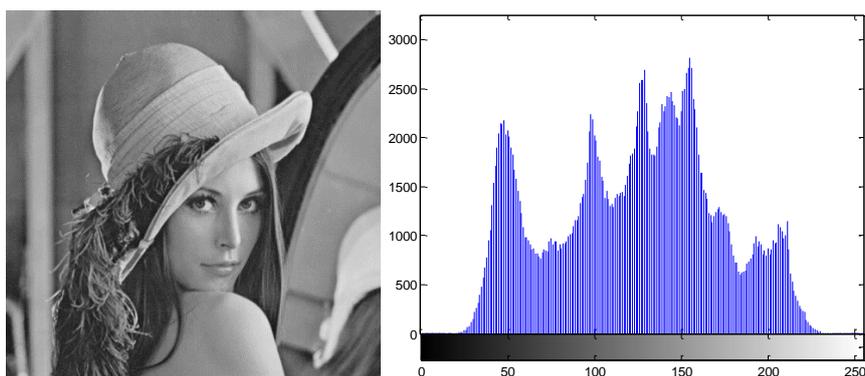


Fig. 2.1 Istogramma in 256 livelli di grigio dell'immagine "Lena", una delle immagini più note ed utilizzate a livello didattico nell'*image processing*

L'istogramma fornisce una rappresentazione compatta dell'immagine, trascurando però ogni informazione relativa alla posizione dei singoli pixel. Di conseguenza, immagini diverse possono essere caratterizzate dallo stesso istogramma. Nonostante ciò, l'istogramma si rivela un potente strumento in grado di fornire un'informazione globale e statistica sulla distribuzione dei valori dei pixel nell'immagine.

Per immagini a colori (argomento trattato al paragrafo 2.6) si possono generare tre istogrammi separati, ognuno dei quali riferito al canale in esame.

A partire dall'istogramma di un'immagine di riferimento è possibile effettuare una ricerca mirata al rilevamento di immagini simili, confrontando tra loro gli istogrammi. Essi rappresentano, infatti, un'importante informazione statistica identificativa dell'immagini e possono quindi essere usati per ottenere un'informazione sulla somiglianza: immagini naturali che rappresentano scenari simili avranno istogrammi simili.

L'istogramma è invariante rispetto a:

- rotazione (un esempio è riportato in Fig. 2.2)
- traslazione

ed è poco suscettibile a cambiamenti di scala.

---

Queste importanti proprietà risultano, nello studio di immagini istologiche, essenziali, in quanto i soggetti dell'immagine possono trovarsi in qualunque posizione spaziale ed avere forme e dimensioni molto differenti.

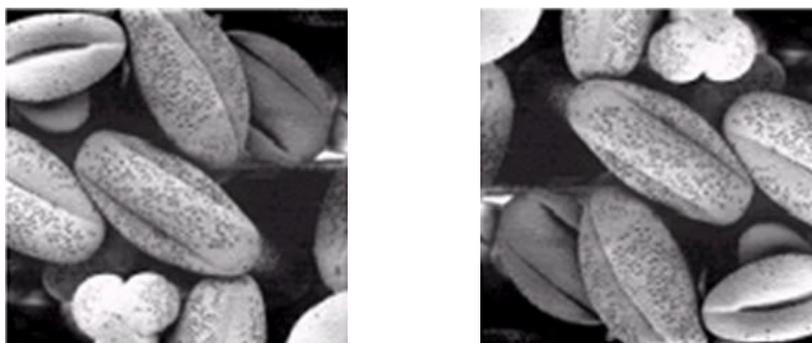


Fig. 2.2 Immagine ruotata di 180°

## 2.2 Elaborazioni globali

Con il termine *Elaborazioni Globali* si intendono tutte quelle operazioni volte a modificare il valore di un *pixel* specifico, utilizzando le informazioni ottenute da tutti i *pixel* dell'immagine originale utilizzando, ad esempio, una trasformazione basata sull'istogramma. Generalmente questo tipo di trasformazioni è alla base di una successiva elaborazione e viene tipicamente utilizzato per migliorare la visualizzazione di un'immagine.

### 2.2.1 Aumento del contrasto

Alcune immagini, ad esempio a causa di una scorretta impostazione dei parametri nel sistema di acquisizione o dalla natura stessa del campione in esame, possono essere poco contrastate. Il contrasto è tipicamente definito come un indice che quantifica la distanza tra i valori più chiari e più scuri di un'immagine in scala di grigio. Fig. 2.3 riporta la stessa immagine, con un diverso livello di contrasto (a destra, maggiore).



Fig. 2.3 Immagini a diverso contrasto (maggiore a destra)

Dall'istogramma è possibile comprendere se un'immagine risulta poco contrastata, poiché in tal caso sarà utilizzata solo una parte del *range* dinamico, ovvero non tutti i possibili livelli di grigio messi a disposizione dalla codifica scelta saranno realmente impiegati. Dopo aver preso visione dell'istogramma è possibile effettuare uno “stiramento” (*stretching*), applicando una trasformazione che espande l'istogramma per coprire l'intero range, cioè verso la parte inferiore (elementi più scuri) e verso la parte superiore (elementi più chiari), ottenendo così all'interno dell'immagine differenze più ampie, anche tra pixel vicini, quindi un maggiore contrasto. Tale operazione può essere realizzata facilmente per mezzo di funzioni di trasformazione lineari come mostrato nell'Eq. 2.1.

$$H_{stretching} = \frac{H - \min(H)}{\max(H) - \min(H)} \quad 2.1$$

L'uso esclusivo di trasformazioni lineari, anche se preserva l'andamento dell'istogramma originale come si può vedere in Fig. 2.4, può non essere sufficiente per determinate applicazioni.

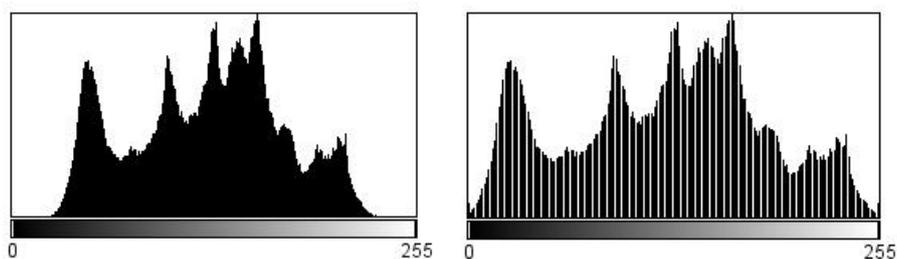


Fig. 2.4 Istogramma originale a destra e dopo l'applicazione dello *stretching* a sinistra

Esistono altre trasformazioni di “stiramento dell’istogramma”, di tipo globale, che si basano su trasformazioni non lineari. Esempi di queste trasformazioni sono la correzione gamma e l’equalizzazione dell’istogramma.

### 2.2.2 Correzione gamma

Un particolare tipo di trasformazione non lineare dell’istogramma si ottiene agendo su un parametro chiamato *gamma* che permette di modificare il contrasto diversamente per livelli di grigio bassi e alti, inoltre, permette di approssimare la caratteristica logaritmica del sistema visivo umano e di compensare le non linearità introdotte dai dispositivi di acquisizione e/o visualizzazione. La luminosità percepita dal sistema visivo umano, infatti, è una funzione logaritmica dell’intensità della luce incidente sull’occhio e ciò comporta che, a variazioni di intensità costante si abbia una sensazione non altrettanto lineare nel cambiamento della luminosità. Il parametro *gamma*, posto all’esponente nell’Eq. 2.2, consente di migliorare il contrasto dell’immagine seguendo le caratteristiche del sistema visivo umano. Infatti, l’occhio umano percepisce meglio le differenze tra le tonalità scure. La trasformazione attuata tra i livelli di grigio è la seguente:

$$y = L \left( \frac{x}{L} \right)^{\frac{1}{\gamma}} \quad 2.2$$

dove  $L$  è il massimo valore di grigio assunto dall’immagine ed  $x$  è il livello di grigio che deve essere trasformato. La funzione riportata nell’Eq.2.2 definisce una famiglia di trasformazioni che agisce in modo nettamente differente a seconda del

---

valore di *gamma*. Al variare del valore di  $\gamma$  si ottiene una famiglia di curve passanti per i punti  $(0,0)$  ed  $(L,L)$ , in Fig. 2.5 è riportata l'andamento dell'Eq. 2.2 al variare di *gamma*.

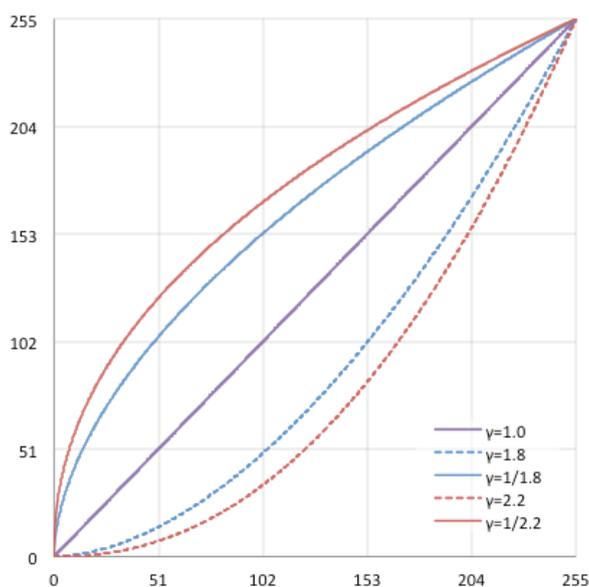


Fig. 2.5 Correzione gamma

Questa trasformazione permette di lasciare inalterato il massimo dell'istogramma, mentre comprime i valori centrali di grigio verso valori più chiari o più scuri a seconda che *gamma* sia maggior minore di uno, un esempio di trasformazione dell'istogramma a seguito di una correzione con *gamma* pari a 2.2 è visibile in Fig. 2.6.

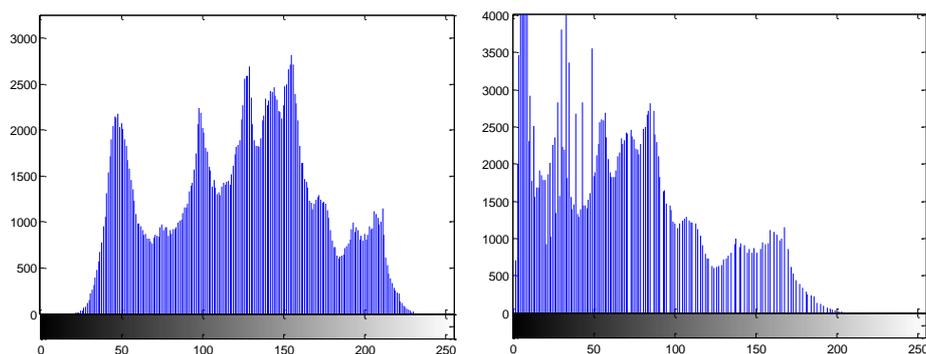


Fig. 2.6 Istogramma originale a sinistra e a destra corretto con gamma pari a 2.2

### 2.2.3 Equalizzazione dell'istogramma

L'operazione di equalizzazione ha lo scopo di ottenere un istogramma con un andamento idealmente costante ottenendo così un istogramma piatto, come si può vedere in Fig. 2.7.

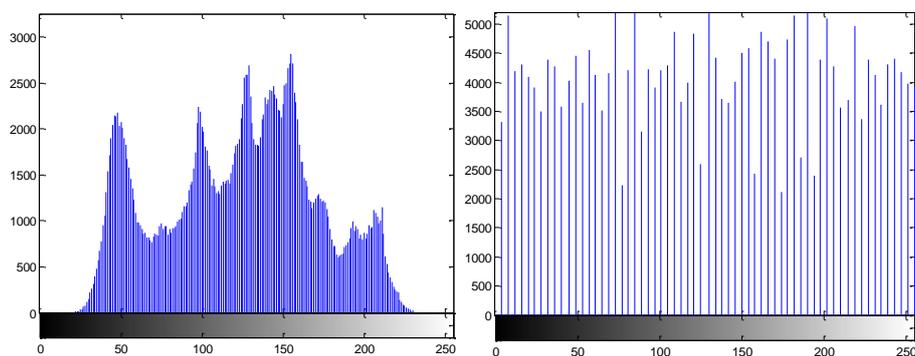


Fig. 2.7 Istogramma originale a sinistra e dopo l'equalizzazione a destra

Questo obiettivo viene raggiunto sostituendo al valore  $x$  di livello di grigio un valore  $y$  dato dalla seguente trasformazione:

---

$$y = \frac{L}{T} \sum_{i=0}^x N_i \quad 2.3$$

dove  $T$  è il numero complessivo dei pixel dell'immagine,  $L$  è il valore massimo dei livelli di grigio,  $N_i$  è il numero di pixel che hanno un livello di grigio uguale a  $i$ .

L'equalizzazione dell'istogramma mira quindi ad aumentare il più possibile sia la dinamica che la gamma tonale dell'immagine, ottenendo un miglioramento complessivo, come mostrato in Fig. 2.8.



Fig. 2.8 L'immagine di destra è ottenuta equalizzando l'istogramma dell'immagine di sinistra

## 2.3 Elaborazioni Locali

Le modifiche di un'immagine a livello globale permettono di migliorare la qualità dell'immagine senza richiedere un elevato costo computazionale, purtroppo, però, tendono a perdere efficacia quando la scena presenta alti gradienti di illuminazione. Infatti, applicare uno *stretching* dell'istogramma su di un'immagine che già utilizza tutto il *range* dinamico non produce alcun miglioramento del contrasto. Altre metodologie per la manipolazione dell'immagine, invece, modificano i livelli di grigio dei *pixel* in relazione alla loro posizione. Le tecniche di elaborazione locale forniscono il valore di ogni *pixel*

---

nell'immagine migliorata  $g(x,y)$  in funzione del corrispondente valore del *pixel* nell'immagine originale  $f(x,y)$  o dei valori dei *pixel* presenti in un opportuno intorno.

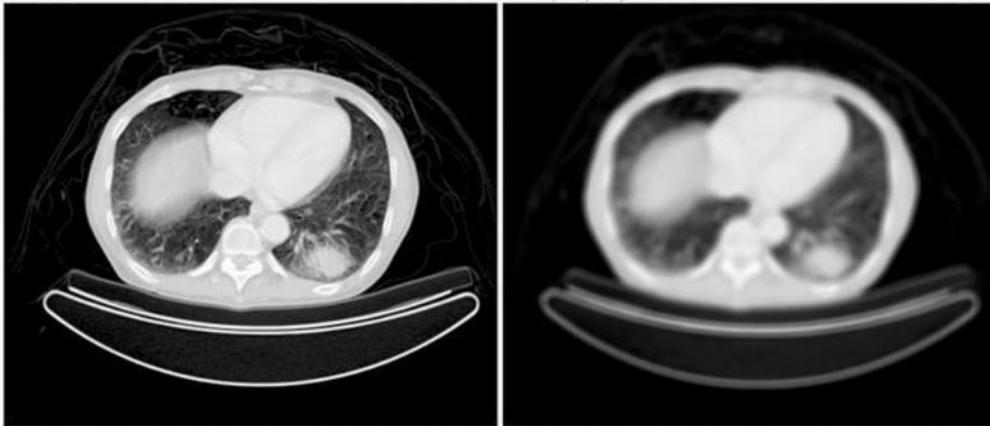


Fig. 2.9 Esempio di elaborazione locale: filtro “media”

Tipicamente, questo tipo di trasformazioni si basa sulla *convoluzione*, una semplice, ma allo stesso tempo potente, operazione matematica che consiste nel moltiplicare tra di loro, elemento per elemento, due vettori o matrici, solitamente di diverse dimensioni, per sommarli alla fine producendo così un valore scalare da sostituire al valore locale del *pixel* nell'immagine finale. Questo tipo di operazione può essere usato nell'*Image Enhancement* per implementare operatori in cui i valori dei *pixel* da produrre non sono altro che una combinazione lineare della matrice di *pixel* su cui agire. Quindi gli ‘operatori’ sono matrici che contengono i fattori moltiplicativi che sono applicati al *pixel* preso in analisi e ai suoi vicini.

Di seguito verranno esposte due operazioni di filtraggio locale: la prima è volta all'aumento del contrasto mentre la seconda mira alla riduzione del rumore.

### 2.3.1 *Sharpening*

Agire sull'istogramma per aumentare il contrasto è una soluzione adottata a livello globale su tutta l'immagine senza in alcun modo riferirsi al suo contenuto.

---

Un'altra tecnica di miglioramento dell'immagine che sfrutta il valore del pixel in esame in relazione ai suoi vicini è lo *sharpening*. Questa tecnica ha lo scopo di esaltare le transizioni di intensità rendendo più nitida l'immagine. Fig. 2.6 mostra, a destra, i risultati di questa operazione effettuata sull'immagine originale, a sinistra.



Fig. 2.10 Effetto dello *sharpening*

Un'immagine appare nitida quando la transizione di intensità tra un *pixel* e l'altro è netta: le transizioni di intensità tra *pixel* adiacenti sono, infatti, legate alle proprietà differenziali dell'immagine, derivata spaziale prima e seconda. Sfruttando tali proprietà lo *sharpening* agisce sulla derivata seconda dell'immagine consentendo di enfatizzare i contorni degli oggetti. Tipici filtri di *sharpening* sono definiti in Eq. 2.4.

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix} \quad \begin{bmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{bmatrix} \quad 2.4$$

### 2.3.2 Riduzione del rumore

Operatori come lo *sharpening*, che lavorano sulla derivata esaltando le transizioni di luminosità, possono generare artefatti se le immagini presentano elevate componenti di rumore. Il rumore può essere concepito come una componente additiva sull'immagine (segnale) che va a corrompere l'informazione. I filtri più

---

semplici ed utilizzati per diminuire il rumore sono la media ed il filtro mediano. Entrambi sono caratterizzati da un unico parametro: la dimensione del filtro. Il filtro “media” esegue una media sui valori dei *pixel* contenuti nella finestra centrata nella posizione del *pixel* che si sta valutando. Questo filtraggio ha lo scopo di attenuare valori isolati di *pixel* con la conseguenza di tendere a rendere localmente uniformi i valori dei *pixel*. In Fig. 2.11, a destra, viene riportata l’immagine ottenuta con un filtro media, sull’immagine originale (a sinistra).

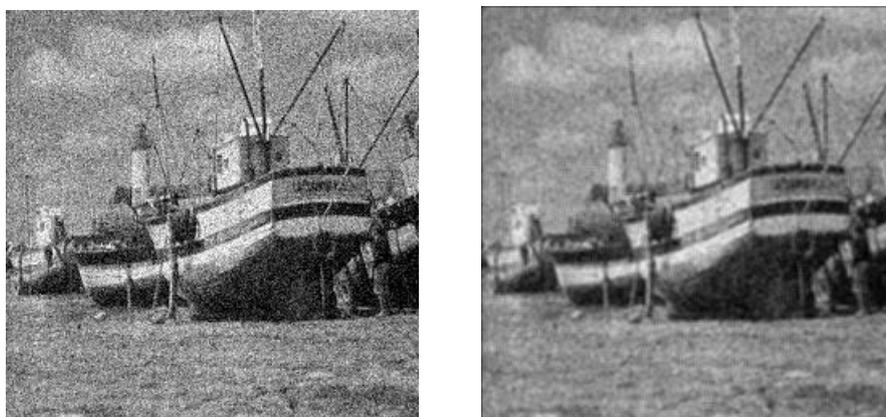


Fig. 2.11 Filtro media

Il filtro mediano, invece, è utilizzato principalmente per rimuovere un particolare tipo di rumore chiamato ‘sale e pepe’, che si presenta come una casuale occorrenza di *pixel* bianchi e neri Fig. 2.12.



Fig. 2.12 Filtro mediano

Il filtro mediano è ottenuto calcolando la mediana della distribuzione dei valori dei *pixel* presenti all'interno della finestra analizzata ed ha l'interessante proprietà di essere *edge preserving*: i contorni degli oggetti non vengono degenerati preservando la nitidezza dell'immagine.

## 2.4 Segmentazione

Le tecniche di *Image Enhancement* discusse precedentemente possono servire a migliorare l'immagine complessiva per finalità di analisi qualitative e quantitative. Tuttavia potrebbe essere necessario individuare, ed estrarre dall'immagine, alcuni oggetti di particolare interesse. Tale processo è noto come *segmentazione*. Lo scopo della segmentazione è semplificare e/o cambiare la rappresentazione delle immagini in qualcosa che risulti più significativo e facile da analizzare. La segmentazione delle immagini è un problema chiave nella *computer vision* e, in generale, nel trattamento di qualunque immagine digitale finalizzato all'analisi dei contenuti. Tale processo permette di suddividere l'immagine in regioni significative e costituisce il primo passo verso l'estrazione di informazioni di alto livello. Un buon metodo di segmentazione deve essere robusto, cioè in grado di fornire risultati soddisfacenti anche quando applicato a diverse immagini della stessa tipologia. I problemi che ricorrono più comunemente, e che sono da evitare, sono la sotto-segmentazione, cioè la fusione

---

nella stessa zona di oggetti semanticamente diversi, e la sovra-segmentazione, cioè la suddivisione in zone multiple di uno stesso oggetto). Al fine di rendere l'algoritmo di segmentazione il più possibile indipendente dal contesto, esso non dovrebbe presupporre alcuna conoscenza a priori sull'immagine, altrimenti risulterebbe difficile garantire che il risultato sia soddisfacente per ogni tipo di problema. Tuttavia, si può cercare di ottenere la suddivisione dell'immagine in un numero di regioni, tali che, ognuna di esse sia omogenea e coerente, rispetto a un dato criterio, e che l'unione di due regioni non lo sia. In [3] è data la seguente definizione formale del problema della segmentazione: se  $P(\cdot)$  è un predicato di omogeneità, ovvero una caratteristica estratta dall'immagine e associata ad ogni *pixel*, definito su un'insieme di *pixel* connessi e se  $F$  è l'immagine, la segmentazione consiste nel determinare un numero di  $n$  sottoimmagini o regioni  $(S_1, S_2, \dots, S_n)$  tali che:

$$\bigcup_{i=1}^n S_i = F \text{ con } S_i \cap S_j = \emptyset \text{ (} i \neq j \text{)} \quad 2.5$$

Il predicato di uniformità  $P_s = true \forall i = 1 \dots n$  in tutte le regioni  $S_i$  e  $P(S_i \cup S_j) = false \forall (i, j) i \neq j$  quando  $S_i$  e  $S_j$  sono adiacenti. Così facendo, si sposta il concetto di segmentazione dall'identificazione degli oggetti semantici ad un requisito di omogeneità sulle regioni (insiemi di *pixel* connessi).

Per ottenere una segmentazione il più vicino possibile a quella attesa è fondamentale la scelta, spesso dettata dall'ambito applicativo, del predicato di omogeneità. Infatti, la partizione di un'immagine in regioni omogenee non garantisce la suddivisione in oggetti semantici: citando [4], il problema della segmentazione di immagini è anche un problema di percezione psico-fisica, soprattutto nel caso di scene particolarmente complesse, e quindi non è risolvibile in maniera completamente analitica. Inoltre, la segmentazione automatica (non supervisionata) di immagini viene tipicamente considerato un problema mal posto, in particolare non è detto che esista un'unica soluzione e che questa sia

---

robusta, proprio a causa del fatto che gli oggetti, dal punto di vista semantico, spesso non corrispondono a zone omogenee rispetto a caratteristiche di basso livello (colore, tessitura, etc.). Tuttavia, se si considera la similarità tra zone in maniera coerente a quella dell'occhio umano, cioè si usano misure di similarità e di omogeneità di regioni corrette da un punto di vista percettivo, è possibile avvicinarsi al risultato atteso con un buon grado di approssimazione.

E' stato dimostrato che la percezione del colore e della tessitura è fondamentale nella visione umana [5]. Quindi, l'approccio migliore da seguire è quello di cercare di imitare nel modo più fedele possibile il sistema percettivo umano, utilizzando le informazioni congiunte di colore e tessitura per ottenere misure di similarità ed omogeneità corrette da un punto di vista percettivo.

In generale, un buon metodo di segmentazione dovrebbe soddisfare i seguenti criteri:

- le regioni devono essere il più possibile omogenee;
- i confini delle regioni devono essere compatibili con le variazioni della misura di similarità adottata;
- aree percettivamente uniformi non devono essere divise in più parti, a maggior ragione in regioni con ombreggiatura graduale e con tessitura;
- piccoli dettagli, se ben definiti in forma e contrasto, non devono essere fusi con le regioni confinanti.

Esistono molte tecniche di segmentazione, sia per quanto riguarda le immagini monocromatiche che per le immagini a colori. Le tecniche si differenziano sia per la definizione del criterio di omogeneità tra regioni sia per l'algoritmo usato per costruire quest'ultime. In generale, tali tecniche si dividono in:

- Tecniche di tipo *bottom-up*, in cui la segmentazione è effettuata partendo da caratteristiche di basso livello dell'immagine a prescindere dal contenuto semantico:
  - tecniche basate sulla discontinuità: *edge detection*
  - tecniche basate sull'omogeneità: *histogram thresholding, clustering, region based.*

- 
- Tecniche *top-down* presuppongono la conoscenza ad alto livello dell'immagine per facilitare l'estrazione di oggetti:
    - *Active contour snakes*
    - *Template matching*

Di seguito sono spiegati i principali metodi di segmentazione presenti in letteratura, metodi che sono stati inizialmente studiati e realizzati per il trattamento di immagini in scala di grigio, ma che sono stati, in seguito, estesi anche all'utilizzo di immagini a colori.

### 2.4.1 Sogliatura globale dell'istogramma

Questo approccio di segmentazione risulta particolarmente efficiente quando i valori medi delle regioni da individuare differiscono in maniera significativa. Se questa condizione è verificata, allora l'istogramma dell'immagine presenterà dei picchi in corrispondenza di tali valori che prendono il nome di "mode". Il caso migliore per utilizzare questa tecnica è quello in cui l'istogramma dell'immagine risulta bimodale (due sole mode), come mostrato in Fig. 2.13: in questo caso la segmentazione può essere ottenuta usando una sogliatura con singolo parametro.

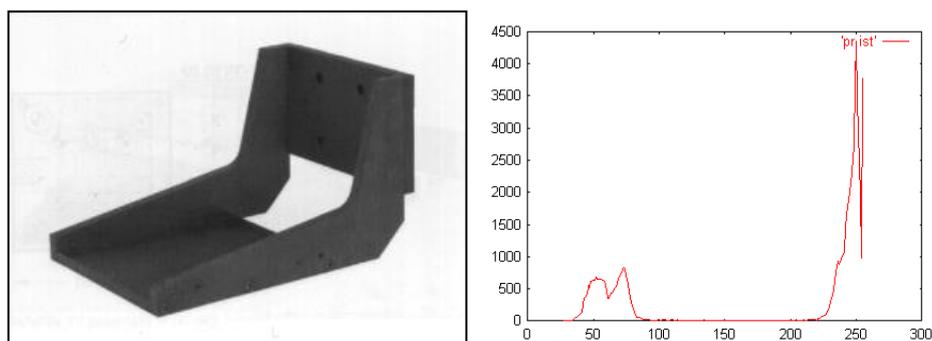


Fig. 2.13 A destra l'istogramma relativo all'immagine di sinistra

Tipicamente, il risultato della segmentazione è un'immagine binaria (in bianco e nero) dove ottenuta ponendo al valore 1 (bianco) i *pixel* della matrice di uscita se l'intensità del *pixel* originale è maggiore della soglia altrimenti lo stesso viene ad

---

assumere valore 0 (nero). Un esempio di binarizzazione per sogliatura è mostrato in Fig. 2.14.

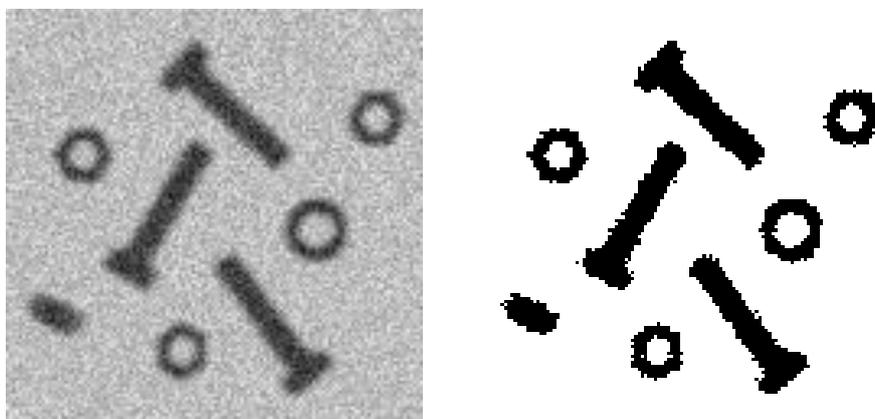


Fig. 2.14 Segmentazione con sogliatura globale

Generalmente, però, l'istogramma delle immagini di interesse non è mai bimodale e presenta picchi multipli. Un esempio è rappresentato dalla Fig. 2.15, a sinistra si vedono i soggetti di 4 diverse intensità, tra loro omogenee e, a destra, il relativo istogramma. In questo caso occorre usare valori di soglia multipli, uno per ogni zona da trovare, dopo avere però effettuato un *denoising* dell'istogramma con una finestra a media mobile, in cui ogni valore del *bin* consiste nel valore medio dei *bin* simmetricamente adiacenti.

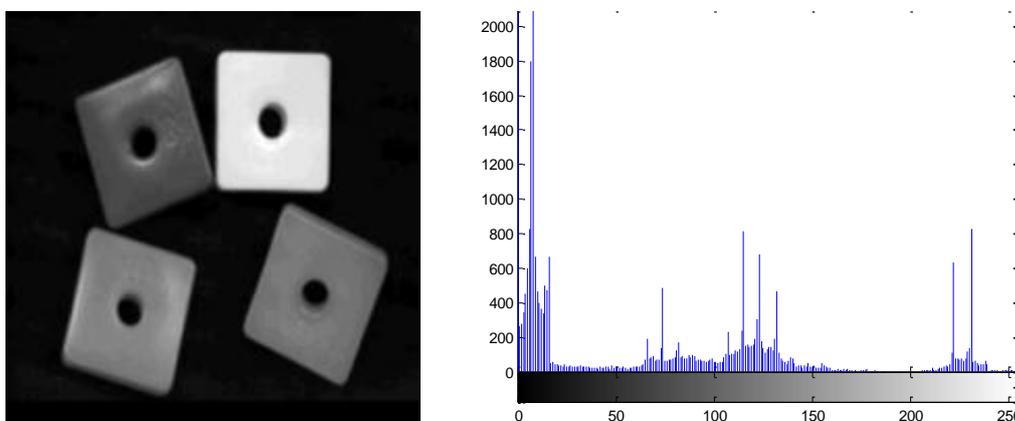


Fig. 2.15 Istogramma multimodale a destra dell'immagine a sinistra

---

La sogliatura presenta nella sua estrema semplicità anche il suo massimo limite. Infatti, per ottenere risultati soddisfacenti, si suppone che gli oggetti da ricercare nell'immagine abbiano tutti un valore di grigio che li rappresenti.

#### 2.4.2 Sogliatura locale dell'istogramma

In molti casi la sogliatura globale non può essere applicata come passo risolutivo del processo di segmentazione, poiché può accadere che non si ottenga il risultato sperato su tutta l'immagine ad esempio a causa di non uniformità di illuminazione che sfuma, in maniera anche non lineare, i livelli di grigio degli oggetti. In questo caso, quando lo sfondo non risulta uniforme e la differenza degli oggetti, in termini di toni di grigio, risulta marcata, tipicamente la sogliatura globale può lavorare correttamente solamente in una parte dell'immagine, producendo risultati insoddisfacenti nelle altre. Se è possibile acquisire un'immagine dello sfondo senza gli oggetti di interesse, allora può essere possibile pre-elaborare le immagini sfruttando questa informazione e continuando ad utilizzare la sogliatura globale. Se tuttavia neppure questo è possibile, è conveniente utilizzare una tecnica leggermente più complessa che si basa sempre sull'istogramma.



Fig. 2.16 Immagine segmentata mediante sogliatura locale

In questo caso si suddivide l'immagine in sotto-immagini, spesso di forma quadrata, su cui viene determinato l'istogramma e il conseguente valore di sogliatura col quale eseguire la segmentazione. Le sotto-immagini possono sovrapporsi, ma se ciò accade sarà necessario implementare un passo successivo

---

di fusione (*merging*) dei risultati ottenuti. Solitamente la sogliatura locale è computazionalmente più onerosa rispetto alla sogliatura globale, sebbene risulti però molto utile per segmentare oggetti in immagini con sfondo non uniforme. Questo metodo di segmentazione è funzione sia della soglia scelta per ogni sotto-immagine che della loro dimensione: infatti, dimensioni troppo grandi non permettono l'identificazione di particolari, mentre dimensioni troppo piccole fanno decadere il concetto di istogramma (normalizzato) come distribuzione di probabilità dei valori dei *pixel*.

### 2.4.3 *Region growing*

Se la sogliatura è un metodo *one-step*, in cui la segmentazione di un'immagine avviene in un solo passo, il *region growing* è un approccio alla segmentazione iterativo che ricerca gruppi di *pixel* con intensità simile. Questo tipo di segmentazione parte dall'analisi di uno o più *pixel* chiamati *seed* (semi), che appartengono alla struttura di interesse, i quali possono essere scelti da un operatore umano o da una procedura automatica. L'algoritmo valuta i *pixel* circostanti i *seed*, i quali sono esaminati uno ad uno ed aggiunti alla regione di *growing* solo se rispettano un criterio di omogeneità. La procedura continua fintanto che nessun altro *pixel* adiacente può essere aggiunto. Al termine, l'oggetto sarà rappresentato da tutti quei *pixel* che hanno "passato" il test di uniformità durante la procedura di *growing*, in Fig. 2.17 si è riportato un esempio di segmentazione del fulmine ottenuta con il metodo del *region growing*. Un esempio di test di uniformità può essere il confronto della differenza di intensità tra il valore del *pixel* da aggiungere e quello della media di una data regione. Questa tecnica può anche essere resa adattiva per esempio ricalcolando la soglia di inclusione dei potenziali *pixel* in base alla deviazione standard dei *pixel* già inclusi.

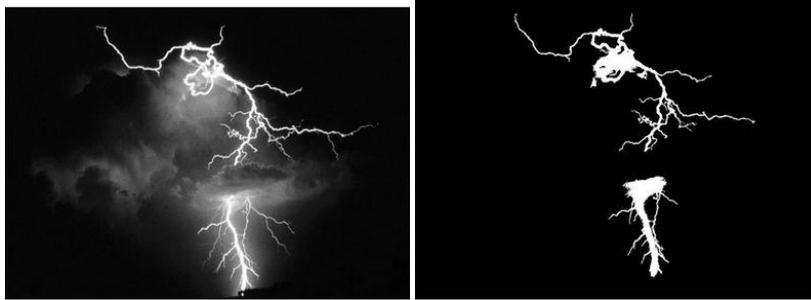


Fig. 2.17 Esempio di segmentazione tramite region growing

I risultati ottenuti dipendono fortemente dalla scelta del criterio di omogeneità, se questo non è scelto nel modo appropriato le regioni si perdono nelle aree adiacenti o si fondono con regioni che non appartengono all'oggetto di interesse.

Inoltre, variare la posizione dei *seed* di partenza, può portare alla creazione di regioni differenti. I vantaggi di questo metodo sono la capacità di segmentare correttamente regioni che hanno le stesse proprietà, ma che sono spazialmente separate e di generare regioni interconnesse.

#### 2.4.4 *Watershed*

La segmentazione *watershed* è una tecnica di segmentazione *region-based* che utilizza la 'morfologia' dell'immagine per rilevare i contorni degli oggetti presenti. Un'immagine può essere rappresentata in una forma bidimensionale o tridimensionale, in quest'ultimo caso la terza dimensione è determinata dal valore di luminosità dei *pixel*, quindi l'immagine diviene una superficie tridimensionale che ricorda un paesaggio montuoso.

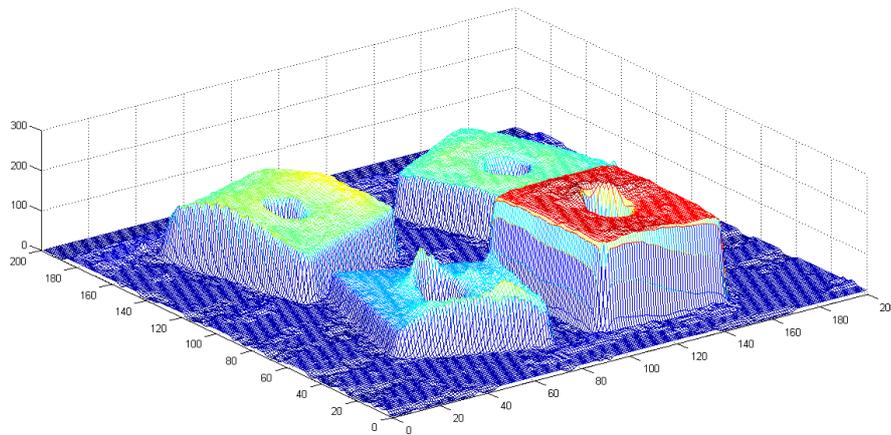


Fig. 2.18 Immagine in Fig. 2.15 rappresentata come superficie tridimensionale

Essa richiede preliminarmente la selezione di almeno un *seed* all'interno di ogni oggetto dell'immagine, incluso lo sfondo trattato come elemento separato. Anche in questo caso i *seed* sono scelti da un operatore o determinati da una procedura automatica specifica nel riconoscimento degli oggetti di interesse. I *seed* fungono da marcatori per gli oggetti che vengono ingranditi in funzione della morfologia (superficie) dell'immagine. Per comprendere il procedimento, si può pensare all'immagine come ad una superficie in cui i pixel scuri rappresentano i bacini (*watershed*, appunto) e quelli chiari rappresentano i bordi dei bacini stessi. Concettualmente, l'algoritmo inizia a riempire i bacini fino a quando almeno due di essi si fondono. I loro bordi prendono il nome di "dighe" e formano il bordo dell'oggetto.

#### 2.4.5 *K-means*

Gli algoritmi di *clustering* appartengono al settore della classificazione non supervisionata, in cui lo spazio dei dati viene suddiviso in aree coerenti, sfruttando il concetto di similarità tra i dati. Questi algoritmi trovano impiego anche nell'elaborazione di immagini. Uno dei più comuni algoritmi di *clustering* è il *K-means* che, a partire dallo spazio delle *feature* effettua un loro raggruppamento in *K* partizioni. La suddivisione viene effettuata sulla base di una metrica in grado

---

di definire una misura di similarità la del valore assunto dagli oggetti che fanno parte dello spazio dei dati, assumendo che essi si distribuiscano in modo gaussiano.

Il *K-means* assume che gli attributi degli oggetti siano rappresentati come vettori, formando uno spazio vettoriale, proponendosi di minimizzare la varianza totale intra-*cluster*. Ogni *cluster* viene identificato mediante un centroide o punto medio.

L'algoritmo segue una procedura iterativa descritta schematicamente come segue:

1. creazione di  $K$  partizioni assegnando, ad ognuna di esse, un elemento del vettore multidimensionale, scelto casualmente oppure usando alcune informazioni euristiche;
2. calcolo del centroide di ogni gruppo;
3. aggiornamento delle partizioni, associando ogni nuovo elemento al *cluster* il cui centroide è più vicino ad esso;
4. si ripetono i punti 2 e 3 fino a convergenza.

Formalmente, tale procedura è definibile come segue: dati  $m$  vettori in  $\mathbb{R}^N$ , definiamo  $X = X_1, X_2, \dots, X_m$  come insieme degli oggetti. Si definisce partizione degli oggetti il gruppo di insiemi  $P = P_1, P_2, \dots, P_k$  che soddisfano tutte le seguenti proprietà:

- $\bigcup_1^K P_i = X$ : tutti gli oggetti devono appartenere ad almeno un *cluster*;
- $\bigcap_1^K P_i = \emptyset$ : ogni singolo oggetto può appartenere ad un solo *cluster*;
- $\emptyset \subset P_i \subset X$ : almeno un oggetto deve appartenere ad un *cluster* e nessun *cluster* può contenere tutti gli oggetti.

Deve valere anche la seguente condizione  $1 < K < N$ ; ovvero il numero di *cluster*  $K$  deve essere almeno due, non avrebbe infatti senso né cercare un solo *cluster* né avere un numero di *cluster* pari al numero di oggetti. Una partizione viene rappresentata mediante una matrice  $U \in \mathbb{N}^{K \times N}$ , il cui generico elemento  $u_{ij}$  normalmente assume il valore 0 ma quando vale 1 indica l'appartenenza dell'oggetto  $x_j$  al *cluster*  $x_i$ . Indichiamo, quindi, con  $C = C_1, C_2, \dots, C_m$  l'insieme dei  $K$  centroidi e definiamo la funzione obiettivo da minimizzare come:

---

$$V(U, C) = \sum_{i=1}^K \sum_{X_j \in P_i} \|X_j - C_i\|^2 \quad 2.6$$

Dall'Eq. 2.6 si calcola il minimo seguendo la procedura iterativa:

1. genera  $U_v$ , e  $C_v$ , casuali;
2. calcola  $U_n$ , che minimizza  $V(U, C_v)$ ;
3. calcola  $C_n$ , che minimizza  $V(U_v, C)$ ;
4. se l'algoritmo converge ci si ferma, altrimenti si assegna  $U_v = U_n$ ,  $C_v = C_n$ , e si ritorna al passo 2

Tipici criteri di convergenza sono: il cambiamento nullo nella matrice  $U$ , o la differenza, non superiore ad una soglia, fra i valori ottenuti dalla funzione obiettivo in due iterazioni successive.

La più popolare implementazione di questo algoritmo è il cosiddetto metodo di Lloyd (1956) nel quale sono implementate le seguenti euristiche per i passi 2 e 3:

- al passo 2 viene associato ciascun punto al centroide a lui più vicino;
- al passo 3 viene ricalcolato ogni centroide come la media dei punti assegnati a quel *cluster*.

L'ampio utilizzo di questo algoritmo deriva dalla sua velocità di convergenza e semplicità di implementazione.

L'algoritmo *K-means* non garantisce la convergenza ad un minimo globale della funzione obiettivo a causa dell'inizializzazione casuale dei punti di partenza, generando così, ad ogni ripetizione, una soluzione differente. Un altro svantaggio dell'algoritmo è che esso richiede, nella sua forma originale, di scegliere il numero  $K$  di *cluster* da trovare.

Al variare del numero di *cluster* si cambia completamente il partizionamento dello spazio dei dati, perché l'algoritmo *k-means* intrinsecamente lo suddivide in modo omogeneo. L'algoritmo funziona bene quando i dati sono naturalmente partizionati, ovvero quando si distribuiscono in maniera uniforme su tutto lo

---

spazio e i *cluster* presentano una distribuzione di tipo sferico intorno ad un punto medio come mostrato in Fig. 2.19.

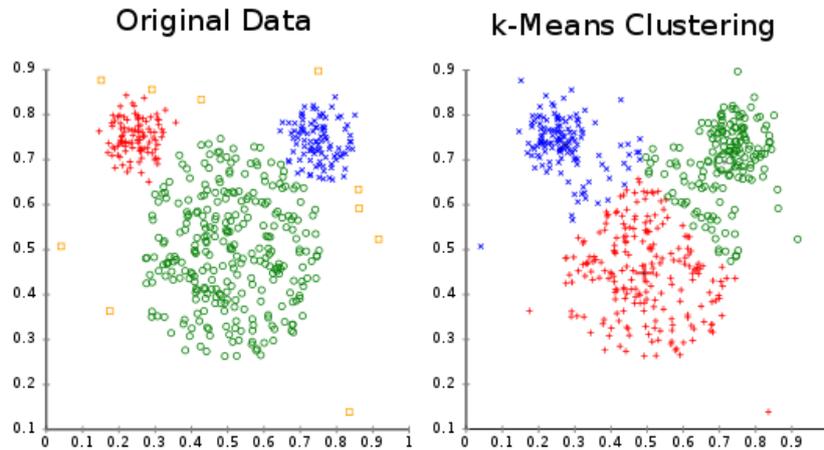


Fig. 2.19 A sinistra dei dati rappresentati in uno spazio bidimensionale e a destra il partizionamento ottenuto con  $K=3$

Nella presente Tesi però i dati da elaborare non sono punti in un piano, ma vettori che rappresentano una distribuzione di probabilità e come verrà discusso nel Capitolo 4 la distanza scelta per la suddivisione dello spazio non è la distanza euclidea, comunemente usata nel k-means e per il partizionamento dei dati in Fig. 2.19, ma la distanza chiamata *city-block* o *Manhattan*.

## 2.5 Texture analysis

Nel campo dell'elaborazione di immagini si è consolidata la definizione, con il termine tessitura (*texture*), di una qualunque disposizione geometrica e ripetitiva dei *pixel* di un'immagine.

Il sistema visivo umano identifica, e riconosce, facilmente diversi tipi di tessiture caratterizzandole in modo soggettivo. Non esiste, infatti, una definizione generale di tessitura ed una metodologia di misura accettata universalmente. Infatti, il concetto di omogeneità o di regolarità non è oggettivamente definito, in quanto riguarda le peculiarità percettive di ogni singolo individuo. Senza riuscire a dare

una definizione generale ed esaustiva si può comunque intuire come la presenza di tessitura in un oggetto di un'immagine sia un fattore di fondamentale importanza per il suo riconoscimento, soprattutto laddove essa rappresenti scene naturali. In Fig. 2.20 sono rappresentati alcuni esempi di *texture* tratti da Wikipedia, queste presentano una ripetizione perfetta di un unico elemento nella colonna di sinistra, fino ad avere una ripetizione del tutto casuale di pixel nella colonna di destra.

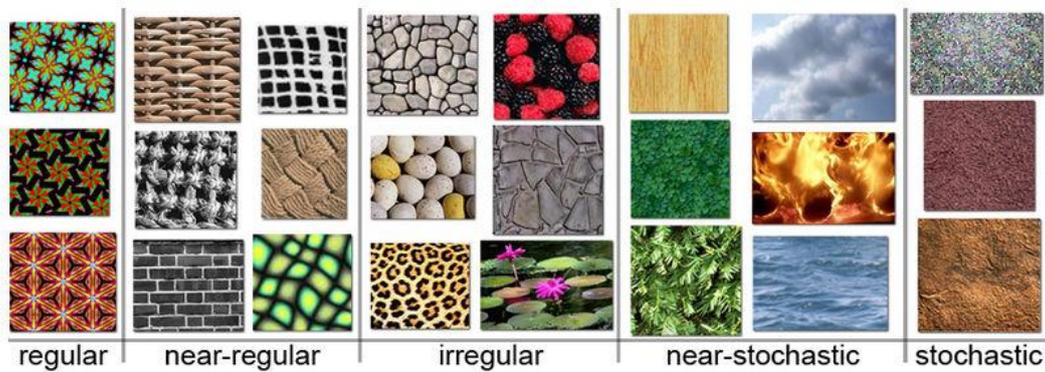


Fig. 2.20 Esempi di *texture*

L'analisi quantitativa della tessitura può essere ricondotta ad una funzione di relazioni statistiche e strutturali tra l'oggetto e l'elemento di base chiamato *texton*. Il nostro sistema visivo determina facilmente le relazioni tra le strutture geometriche fondamentali, caratterizzate da una determinata tessitura, che compone macrostrutture come ad esempio il rivestimento regolare di un fabbricato o di un pavimento rispettivamente composto da tegole o mattonelle.

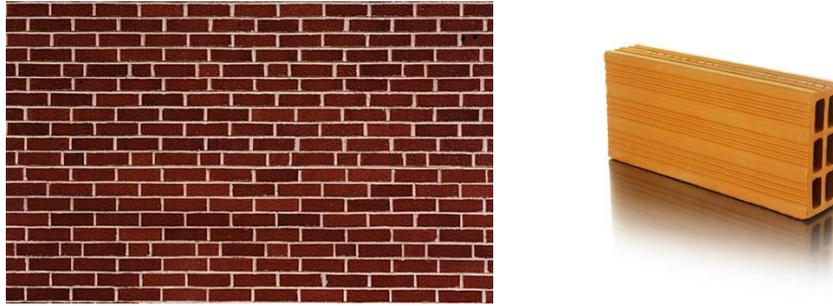


Fig. 2.21 *Texture e texon*

Analogamente, il nostro sistema percettivo è abile ad interpretare con facilità una tessitura composta da microstrutture come si può verificare, ad esempio, osservando un'immagine da satellite con vari tipi di coltivazioni di un territorio, o le microstrutture associate ad un vestito, o ancora quelle osservate da un'immagine al microscopio.

La *texture* di un oggetto può essere percepita da un osservatore in modo diverso in base alla dimensione ed alla risoluzione dell'immagine: ad esempio, una scena di un muro di mattoni ad una risoluzione grossolana è percepita come formata da un unico oggetto ed i dettagli sono persi, mentre ad una risoluzione più elevata vengono riconosciuti. Si può affermare che una regione è percepita come *texture* quando il numero di oggetti primitivi presenti è molto elevato; altrimenti, se solo pochi oggetti primitivi sono presenti allora questi sono individuati come singole entità.

## 2.6 Il colore

Abbiamo già parlato dei colori, ma in questo paragrafo ne daremo una spiegazione più estesa. Un'immagine a colori è rappresentata da tre matrici, ognuna delle quali rappresenta l'immagine in scala di grigio riferita al corrispondente canale Fig. 2.22.

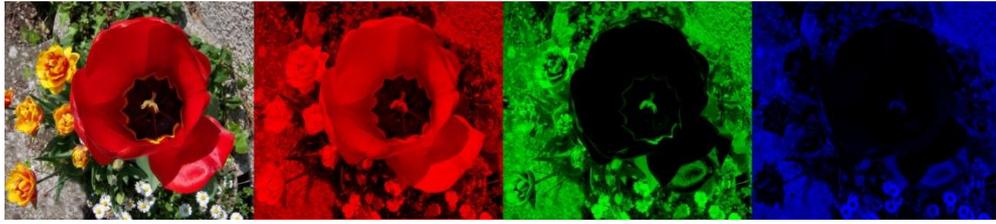


Fig. 2.22 Scomposizione di un'immagine (a sinistra) nelle tre componenti RGB (le ultime tre)

Le immagini acquisite da dispositivi elettronici sono rappresentate nello spazio colore RGB poiché gli schermi utilizzano un sistema di riproduzione del colore di tipo additivo che è ben rappresentato da questo spazio.

Uno spazio dei colori è la combinazione di un modello di colore e di una appropriata funzione di mappatura di quest'ultimo. Un modello di colore è un modello matematico astratto che descrive un modo per rappresentare i colori come combinazioni di numeri, tipicamente tre, quattro o più valori detti componenti colore, rispettivamente per: immagini comuni, tipografiche e telerilevate. Nella presente tesi si farà sempre riferimento ad immagini composte da tre componenti colore. Come è noto, i modelli sono delle astrazioni e, in quanto tali, possono trovare un impiego universale, a patto che ad essi vengano affiancate regole adatte allo specifico problema in esame. Mediante la combinazione di tali regole è possibile creare, o perfezionare, se è già esistente, lo spazio colore scelto.

Poiché il colore è una proprietà percettiva è necessario descrivere brevemente qual è lo "strumento" con il quale gli animali, ed in particolare noi esseri umani acquisiamo le immagini dal mondo esterno: l'occhio.

### 2.6.1 L'occhio

L'occhio contiene due tipi di fotorecettori: i bastoncelli e i coni. I primi sono circa  $120 \times 10^6$ , e sono posti prevalentemente alla periferia della retina, sono altamente sensibili alle basse luminosità e sono i principali responsabili della visione notturna (o scotopica). I coni, invece, sono presenti in numero molto inferiore, circa  $6 \times 10^6$ , e si collocano prevalentemente nella parte centrale della retina che

---

prende il nome di fovea. Sono molto meno sensibili alla luce, ma sono i maggiori deputati alla visione diurna ed a ricevere l'informazione sul colore. I coni e i bastoncelli, se illuminati da una luce con spettro  $I(\lambda)$ , dove  $\lambda$  è la lunghezza d'onda, producono un impulso elettrico che viene inviato ai neuroni deputati alla decodifica dell'informazione visiva.

La tensione generata è:

$$V \propto \int S(\lambda) * I(\lambda) d\lambda \quad 2.7$$

dove  $S(\lambda)$  è lo spettro di assorbimento.

Se i bastoncelli sono di un solo tipo in tutta la retina, lo stesso non vale per i coni, poiché ne esistono ben tre tipi. Ogni tipologia di cono è sensibile a specifiche lunghezze d'onda: lunghe (L), medie (M) e corte (S), in Fig. 2.23 è presente la risposta delle tre tipologie di coni alle varie lunghezze d'onda. Questo meccanismo permette di discriminare, in una certa misura, le lunghezze d'onda della luce incidente nell'occhio, donandoci la visione a colori.

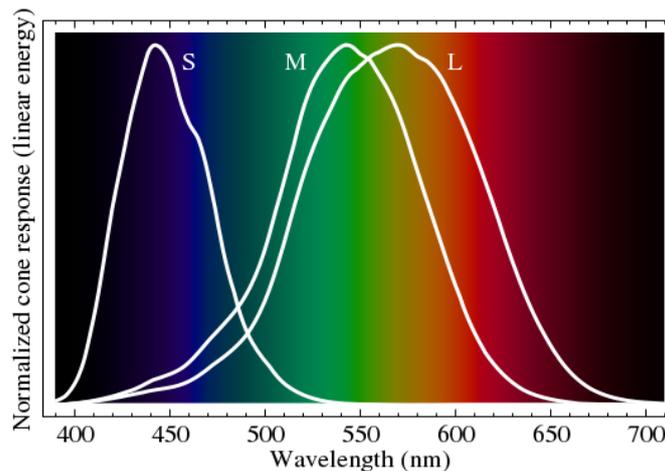


Fig. 2.23 Spettro di assorbimento normalizzato di coni S, M e L

---

È quindi possibile pensare di rappresentare il colore tramite 3 grandezze scalari, funzioni dello spettro  $I(\lambda)$  della luce incidente. La corrispondenza tra spettro e colore non è però biunivoca e ciò genera i “metameri” ovvero condizioni in cui spettri diversi sono percepiti come lo stesso colore.

Dalla triplice natura dei coni nasce l’idea di rappresentare il colore come una terna di numeri scalari. Un modo semplice per riprodurre una vasta gamma di colori consiste nel mescolare opportunamente tre sorgenti primarie. Lo spazio colore RGB rispecchia tale filosofia. Questo spazio può essere rappresentato da un cubo (Fig. 2.24), in cui gli assi principali rappresentano le componenti colore Rosso, Verde e Blu dove nella sua rappresentazione normalizzata le coordinate  $(0,0,0)$  rappresentano il colore nero e  $(1,1,1)$  il bianco.

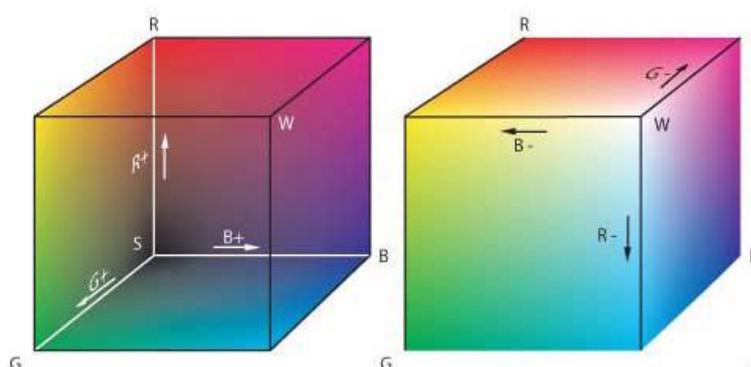


Fig. 2.24 Il cubo rappresentante lo spazio RGB

Se lo spazio RGB è il più comune non è necessariamente il migliore da utilizzare in ogni situazione. Oltre ad RGB, in questa tesi sono stati presi in considerazione altri due spazi colore: XYZ e LAB.

### 2.6.2 Spazio XYZ

Nel 1931 la *Commission Internationale de l'Eclairage* (Commissione Internazionale per l'Illuminazione) definì uno spazio colore, chiamato XYZ, che comprendeva tutte le tinte percepibili dall’occhio umano, a prescindere dalla luminanza ovvero l’effettiva quantità di luce che arriva all’occhio. Lo spazio XYZ

---

può essere rappresentato su di un piano se vengono applicate le seguenti regole di normalizzazione:

$$x = \frac{X}{X+Y+Z} \quad y = \frac{Y}{X+Y+Z} \quad z = \frac{Z}{X+Y+Z} \quad 2.8$$

e sapendo che  $x + y + z = 1$ .

Infatti, qualunque colore all'interno di questo spazio bidimensionale, mostrato in Fig. 2.25 a destra, può avere una luminanza che varia dal bianco al nero e, se si tiene conto anche di questo fattore, lo spazio così definito diviene tridimensionale e rappresentato mediante coordinate XYZ, mostrato in Fig. 2.25 a sinistra.

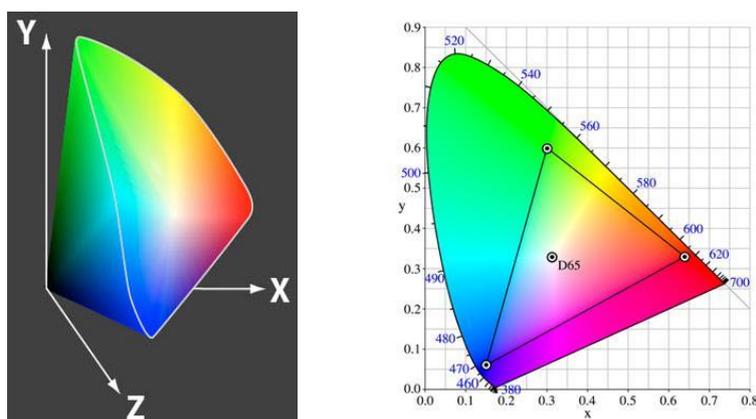


Fig. 2.25 A destra rappresentazione tridimensionale dello spazio XYZ e a sinistra il diagramma di cromaticità

Il modello CIE 1931 si basa, come altre codifiche note, sull'utilizzo di tre colori primari che, opportunamente miscelati tra loro in sintesi additiva, permettono di ottenere tutti i colori che l'occhio umano può percepire. A differenza dei metodi RGB o CMYK, usati rispettivamente in sintesi additiva e sottrattiva, il diagramma di cromaticità proposto dalla CIE non dipende dal comportamento del dispositivo di visualizzazione o di stampa in quanto basato sul concetto di *Osservatore Standard*. Quest'ultimo è definito a partire dalle proprietà del sistema visivo dell'uomo e si basa su analisi sistematiche effettuate su un vasto campione di

---

osservatori umani. In base a numerosi studi effettuati nel primo dopoguerra fu notata l'impossibilità di riuscire a riprodurre per sintesi additiva tutti i colori comunque si scegliesse la terna di primari reali da miscelare.

Poiché lo spazio di colore del CIE può rappresentare tutte le tinte percepibili, è preso come riferimento per tutti gli altri. I primari scelti dalla CIE per generare tutti i colori visibili sono tinte ipersature: colori più saturi di quanto i nostri fotorecettori retinici siano in grado di decifrare e per questo sono chiamati "primari immaginari". I tre primari immaginari sono stati denominati X, Y, e Z. X corrisponde ad un rosso violaceo ipersaturo contraddistinto da due picchi nello spettro cromatico rispettivamente intorno ai 450 nm e ai 600 nm (quest'ultimo molto superiore al primo), Y e Z corrispondono a tinte spettrali, sempre ipersature, con lunghezza d'onda dominante rispettivamente di 520 nm e 477 nm. Inoltre, la tinta Y corrispondente al "verde ipersaturo", e presenta un andamento proporzionale alla nostra sensibilità alla luminosità delle tinte. Scelti i tre primari è possibile ottenere, per sintesi additiva, qualsiasi tinta reale ed utilizzare uno spazio tridimensionale, avente per assi i tre primari, per codificare tutti i colori.

### 2.6.3 Spazio Lab

Un problema significativo dello spazio colore XYZ è che ad uguali distanze geometriche non corrispondono uguali distanze percettive. Tale concetto è stato dimostrato con l'esperimento di MacAdam (1942): un osservatore tenta di riprodurre un colore dato mescolando 3 sorgenti primarie e, a causa della diversa sensibilità dell'occhio di ogni soggetto, il colore ottenuto non è identico al campione. Ripetendo più volte l'esperimento e misurando la dispersione, il risultato è che l'errore commesso varia notevolmente a seconda del colore di partenza, in Fig. 2.26 è presente una rappresentazione grafica degli errori (ellissi) in funzione del colore di partenza (punto nero al centro di ogni ellisse).

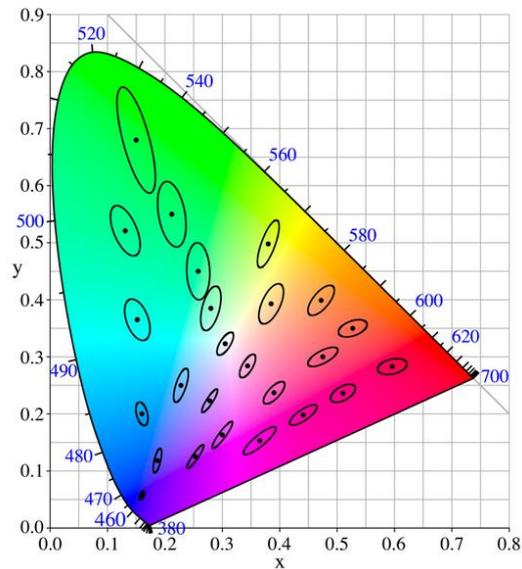


Fig. 2.26 Ellissi di MacAdam

Quindi si è pensato di generare uno spazio colore distorto dallo spazio XYZ con una funzione non lineare, in modo da trasformare le ellissi di MacAdam in cerchi della stessa dimensione.

La teoria dell'opponenza cromatica suggerisce che l'occhio umano, oltre ai tre colori primari (RGB), genera tre sensazioni: rosso-verde; giallo-blu e bianco-nero.

Da questo è stato generato uno spazio colore in cui gli assi corrispondono a :

- L, componente chiamata chiarezza e rappresenta l'asse black-White;
- a, componente Red-Green;
- b, componente Yellow-Blue.

Il sistema LAB separa l'informazione relativa al colore (croma) da quella acromatica della luminosità. Tinta e croma sono qui definite dalle coordinate  $a$ , dal verde al rosso, e  $b$ , dal blu al giallo,  $L$  identifica la luminosità e cresce dal basso verso l'alto, come mostrato in Fig. 2.27.

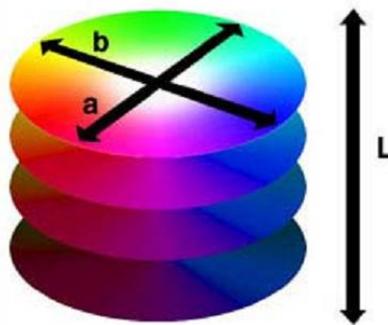


Fig. 2.27 Rappresentazione dello spazio Lab

Pertanto, una sezione orizzontale del modello LAB è un piano che rappresenta tutti i colori con la stessa luminosità.

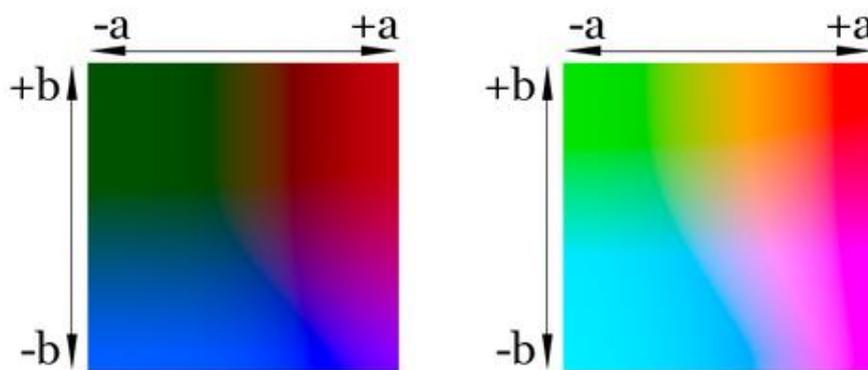


Fig. 2.28 Spazio Lab con luminosità 75% a destra e 25% a sinistra

Anche se è stato concepito per essere uno spazio uniforme, ovvero coppie di punti ugualmente distanti (geometricamente) tra loro rappresentano colori che sono analogamente distanti percettivamente, Lab non lo è completamente, ad esempio nella zona dei gialli la distanza di minima variazione è più alta, e nei blu è più bassa, oltremodo rimane uno spazio molto migliore, dal punto di vista percettivo dello spazio XYZ.

La luminanza, canale Y, dello spazio XYZ non è percettivamente uniforme, infatti, un aumento costante della luminanza a partire da una bassa non provoca la

---

stessa percezione di una identico aumento a partire da valori più alti, legge di Weber-Fechner. Quindi per costruire lo spazio Lab, la relazione che c'è tra luminanza e chiarezza è funzione del rapporto tra il valore Y del colore desiderato e del valore del bianco di riferimento  $Y_n$ .

$$\text{se } \frac{Y}{Y_n} \leq \left(\frac{24}{116}\right)^3$$

$$\text{allora } L = \left(\frac{24389}{27}\right) * \frac{Y}{Y_n} + \frac{468}{19} \quad 2.9$$

$$\text{altrimenti } L = \left(\frac{Y}{Y_n}\right)^{\frac{1}{3}}$$

Il bianco di riferimento, per cui  $Y=Y_n$ , ha dunque un valore di  $L=100$ , ed un nero perfetto per cui  $Y=Y/Y_n=0$  ha un valore di  $L=0$  e un grigio che, per l'occhio, appare a metà il bianco di riferimento e il nero ha un valore di  $L$  di circa 50. La relazione tra fattore di luminanza e chiarezza è tale che, quando il fattore di luminanza è basso, l'occhio percepisce anche una piccola differenza di luminanza mentre quando è elevata la stessa differenza viene percepita con minore intensità ed è quindi necessaria una differenza maggiore per causare una percezione uguale alla precedente. Le trasformazioni che sono applicate per passare dallo spazio XYZ a Lab sono di seguito indicate:

---

$$a = 500 * (f_x - f_y)$$

$$b = 500 * (f_y - f_z)$$

$$\text{se } \frac{X}{X_n} \leq \varepsilon \text{ allora } f_x = k * \frac{X}{X_n} + \frac{16}{116}$$

$$\text{altrimenti } f_x = \left(\frac{X}{X_n}\right)^{\frac{1}{3}}$$

$$\text{se } \frac{Y}{Y_n} \leq \varepsilon \text{ allora } f_y = k * \frac{Y}{Y_n} + \frac{16}{116} \quad 2.10$$

$$\text{altrimenti } f_y = \left(\frac{Y}{Y_n}\right)^{\frac{1}{3}}$$

$$\text{se } \frac{Z}{Z_n} \leq \varepsilon \text{ allora } f_z = k * \frac{Z}{Z_n} + \frac{16}{116}$$

$$\text{altrimenti } f_x = \left(\frac{Z}{Z_n}\right)^{\frac{1}{3}}$$

In cui  $\varepsilon=(24/116)^3$  e  $k=841/108$ . Con queste definizioni la formula per il calcolo di L diventa:

$$L = 116 * f_y - 16 \quad 2.11$$



# Capitolo 3

## Stato dell'arte

Le tecniche di elaborazione descritte precedentemente trovano terreno fertile per l'analisi di numerose tipologie di immagini. L'ambito medico e biomedico può contare su numerosi studi ed applicazioni di *image processing* per immagini provenienti da TAC, risonanza magnetica oppure acquisite tramite microscopia per analisi istologica e citologica. Le applicazioni create per l'istologia sono volte a facilitare la classificazione di malattie e ad alleggerire il carico di lavoro degli anatomopatologi, identificando, in modo automatico o semi-automatico, le regioni sospette nei campioni in esame. L'importanza di questi studi è indiscutibile per risparmiare tempo e risorse, si pensi a tal proposito che negli Stati Uniti circa l'80% delle biopsie prostatiche eseguite ogni anno risultano benigne e ciò si traduce in un dispendio di tempo per l'anatomopatologo ed una spesa per il servizio sanitario [6].

L'esame istologico avviene solitamente guardando un vetrino al microscopio sul quale l'anatomopatologo effettua la diagnosi. Questa procedura è alquanto 'operatore dipendente', poiché molto soggettiva e basata sull'esperienza e sul livello di training dell'anatomopatologo. Risulterebbe quindi molto utile affiancare al lavoro di questa figura strumenti in grado di elaborare le

---

informazioni (contenute nell'immagine) in maniera riproducibile e con dati quantitativi che li aiutino ad effettuare una diagnosi. Infatti in letteratura è possibile ritrovare diversi studi [7] e [8], che pongono l'attenzione sulla riproducibilità della classificazione mediante la scala *Gleason* per cancro alla prostata. In [7] si riporta un esperimento condotto su un campione di 10 anatomopatologi a cui è stato chiesto di classificare 64 dei vetrini istologici che presentavano le stadiazioni di tumore da 1 a 10. I risultati del test riportano che mediamente solo il 56% degli anatomopatologi era in accordo sulla stadiazione fatta eseguita dai colleghi. In [8] si sottolinea come la classificazione sia suscettibile rispetto alla dimensione del campione ottenuto dalla biopsia e dalla risoluzione dell'immagine. In particolare laddove le lesioni sono molto piccole si riscontrando errori clinicamente rilevanti (17%) in merito al grado di stadiazione tumorale effettuata dall'anatomopatologo.

La possibilità di estrarre informazioni quantitative, tramite *image processing* da un vetrino, permette poi il confronto dei dati tra diversi campioni.

Alcune tecniche di elaborazione di immagini molto semplici, ma non per questo meno importanti, trovano impiego in citologia. In [9] viene presentato un metodo di segmentazione di immagini citologiche con lo scopo di individuare le cellule di linfoma presenti nel sangue. L'approccio utilizzato impiega la tecnica chiamata *active contour snake*. Secondo questo modo di procedere l'operatore deve individuare, in modo qualitativo, il contorno dei linfomi da segmentare, in questo modo il software trova i contorni esatti delle cellule Fig. 3.1. I risultati riportati sono ottimi, ma come sostengono gli autori, il software è basato in modo specifico sulla ricerca di linfomi nel sangue e risulta difficilmente applicabile altrove.

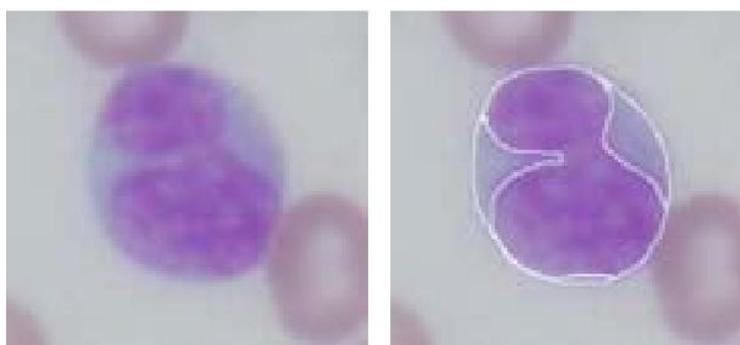


Fig. 3.1 Esempio estrazione dei contorni di una cellula – Tratto da [9]

Dai campioni istologici, a differenza di quelli citologici, si può ottenere una visione più completa della malattia ed i suoi effetti sui tessuti. Per questa ragione l'immagine istologica rimane il metodo di diagnosi *gold standard* per un considerevole numero di patologie, tra cui quasi tutti i tipi di cancro [10].

Uno dei prerequisiti per identificare il *grading*, ovvero l'aggressività del tumore, è la ricerca di particolari strutture come linfociti, nuclei cancerosi e ghiandole. La loro presenza, dimensione, forma e morfologia sono infatti le caratteristiche che vengono utilizzate per diagnosticare la severità della malattia come riportato in [11] e [12] rispettivamente per tumori prostatici ed al seno.

L'estrazione di particolari strutture all'interno dell'immagine è un tipico problema di segmentazione che può essere affrontato in diversi modi. In [13] viene proposta una tecnica automatica per la segmentazione di nuclei in tessuti prostatici e cutanei. L'estrazione degli oggetti si basa sulla sogliatura globale dell'immagine e, per migliorare il risultato, sono applicati operatori morfologici quali erosione e dilatazione. Gli autori sostengono di aver ottenuto risultati incoraggianti, ma migliorabili, aggiungendo al software un'identificazione automatica dei contorni per guidare le operazioni morfologiche.

Un'altra tecnica per l'identificazione di nuclei è proposta in [14] dove viene utilizzato un metodo ibrido. La procedura inizia identificando la posizione del nucleo tramite una sogliatura globale e, successivamente, i contorni sono estratti utilizzando un *active contour snake* che dipende esclusivamente da un parametro che indica il grado levigatezza del contorno Fig. 3.2. Da prove sperimentali

---

condotte su un database molto grande di immagini in scala di grigio, contenenti un'unica cellula grande circa un quarto dell'intera immagine, gli autori sono riusciti ad ottenere ottimi risultati di segmentazione.

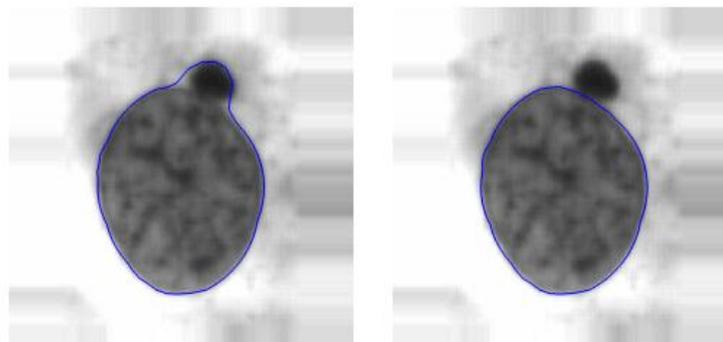


Fig. 3.2 Identificazione dei contorni da parte di [14]

Gli autori sostengono che, questo metodo, del tutto automatizzato, possa essere impiegato anche con immagini più complesse ma ancora non sono state fatte prove in merito.

Un altro approccio alla segmentazione di nuclei è proposto in [15], dove viene impiegato un algoritmo di *clustering* in *logica fuzzy*. Tale algoritmo prende in considerazione il colore, la circolarità e la dimensione degli oggetti presenti nell'immagine con la finalità individuare due classi: sfondo e cellule. Gli autori sostengono che questo tipo di approccio è vincente rispetto ad una segmentazione basata esclusivamente sul colore.

Gli articoli citati in precedenza si focalizzano sul problema della segmentazione di nuclei, mentre in [16] è proposta una tecnica per la segmentazione di ghiandole (Fig. 3.3).



Fig. 3.3 Tessuto benigno e pattern di ghiandola cerchiata in giallo – Tratto da [16]

Il processo di identificazione delle ghiandole inizia con una classificazione dei pixel in base al colore tramite un *nearest neighbor classifier* addestrato con circa 200 campioni per il riconoscimento di 5 tipi di regioni: stroma, lume, sfondo bianco, nuclei e citoplasma. L'estrazione dei contorni della ghiandola avviene considerando l'unione della classe dei nuclei con quella del citoplasma, per poi passare ad indagare la sua conformazione e le sue caratteristiche. Il metodo è molto interessante perché è impiegato su un'immagine istologica piuttosto grande e non solo sui nuclei come le tecniche precedenti. Ovviamente tutte le immagini da elaborare devono contenere le classi suddette e, secondo gli autori, le informazioni estratte riescono a discriminare tra tumori di stadio 3 e 4. Sfortunatamente il metodo non è adattivo, infatti l'immagine viene classificata sulla base dell'addestramento effettuato in precedenza: un addestramento non accurato o incompleto porta sicuramente ad errori nella classificazione.

Un altro metodo per l'identificazione di ghiandole è quello descritto in [17] che fa uso di un *active contour snake* applicato su zone bianche dell'immagine, considerate il centro della ghiandola. Le zone su cui eseguire la segmentazione sono scelte in base alle dimensioni ed alla loro struttura come si può vedere in Fig. 3.4

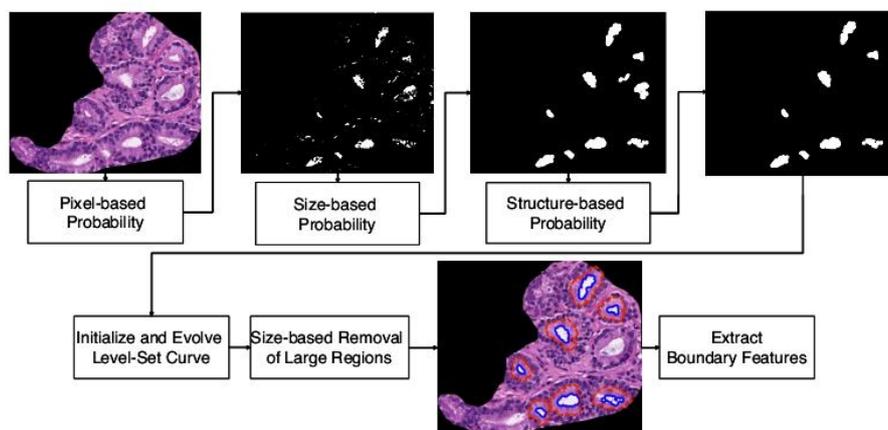


Fig. 3.4 Funzionamento dell'algoritmo [17] – Tratto da [17]

Il risultato presentato dagli autori è buono ma non sempre eccellente in quanto si possono incontrare ghiandole artefatte o non individuarle laddove il centro della ghiandola non sia identificato.

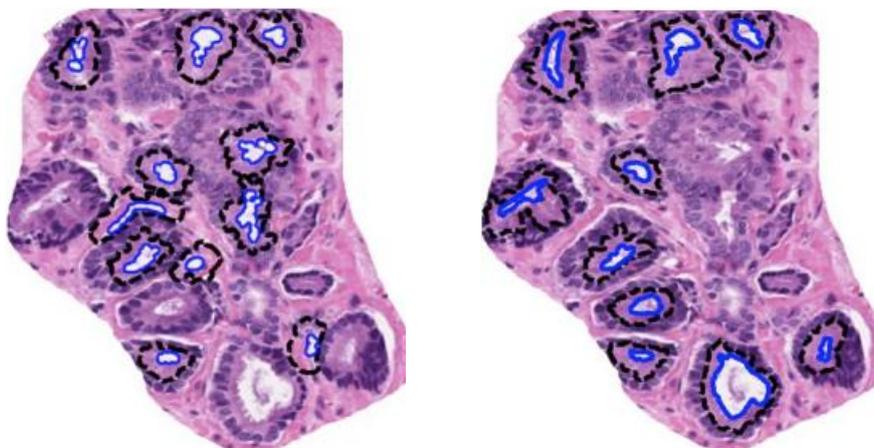


Fig. 3.5 A destra risultato della segmentazione automatica e a sinistra segmentazione manuale fatta dall'anatomopatologo – Tratta da [17]

Un diverso tipo di approccio, che a detta degli autori dovrebbe simulare il modo di procedere nell'indagine di un vetrino istologico da parte di un anatomopatologo, è descritto in [18]. In questo studio si fa uso di una tecnica

---

chiamata multi risoluzione: l'immagine viene manipolata in modo da riuscire a simulare la visione della stessa a diversi ingrandimenti Fig. 3.6.

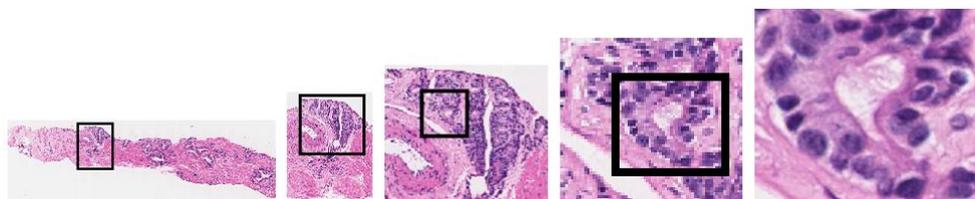


Fig. 3.6 Esempio di indagine multi risoluzione – Tratta da [18]

Da queste immagini vengono estratte circa 600 *feature* da ogni pixel che poi vengono analizzate per eseguire la classificazione. Gli autori sostengono che il metodo risulta robusto ed accurato tuttavia è stato testato solo su tre immagini ed inoltre i criteri decisionali sono definiti a priori.

Algoritmi di *clustering* non supervisionati, come il *k-means*, trovano applicazione in diversi campi che spaziano dalla visione industriale, al riconoscimento automatico di immagini [19], al *perceptual hashing* [20] sino alla segmentazione di immagini mediche [21], [22]. Generalmente, il *k-means* è applicato utilizzando come valori d'intensità dei *pixel*, calcolati usando la rappresentazione RGB o in qualche altro spazio colore. Ad esempio, in ambito istologico, in [23], per l'individuazione di regioni tumorali, effettuano una segmentazione in due passi utilizzando sempre l'algoritmo *k-means*. La prima fase è condotta per isolare il background e lo stroma dalle porzioni ricche di cromatina, in funzione dei valori di luminanza. Successivamente l'algoritmo viene applicato nello spazio colore Lab per separare, in due *cluster*, le strutture di interesse dallo stroma. Tuttavia gli autori non determinano, in maniera quantitativa, la bontà della segmentazione e nemmeno forniscono comparazioni tra i risultati ottenuti applicando l'algoritmo su immagini in scala di grigio e quelli ottenuti su immagini a colori. Inoltre, il *k-means* è utilizzato come metodo di *thresholding* globale per l'identificazione di *feature* istologiche rilevanti. Nell'articolo [19] l'algoritmo *k-means* viene esteso in modo da identificare automaticamente il numero di *cluster* necessari alla

---

segmentazione ed anche i centroidi iniziali, altrimenti generati casualmente. Inoltre, per migliorare l'accuratezza delle regioni segmentate, rimuovere il rumore ed i dettagli non necessari alla classificazione delle immagini, è stato impiegato, a seguito del processamento, un filtro basato sull'istogramma locale di ciascun *pixel* dell'immagine applicato in uscita dall'algoritmo di *labeling*.

Per superare alcuni errori di classificazione, che si possono riscontrare utilizzando come dati di partenza per il *k-means* solo i valori dei singoli *pixel*, si procede ad effettuare una valutazione delle caratteristiche comuni ai *pixel* prossimi a quello elaborato. Recenti sono stati i tentativi di utilizzare le informazioni relative ai vicini dei *pixel* durante la fase di segmentazione mediante istogramma locale, piuttosto che utilizzare solo il valore di un unico *pixel*, come riportato dai lavori [19], [24], [25]. In [21], gli autori, usano l'algoritmo *k-means* per segmentare immagini istologiche in 4 classi, che biologicamente rappresentano: lume ghiandolare, stroma, citoplasma delle cellule epiteliali e nuclei cellulari, basandosi sulla media dell'intensità di colore. Tuttavia gli autori ammettono che, senza utilizzare informazioni locali, un approccio simile potrebbe portare a risultati non corretti. Per aumentare la precisione, essi calcolano le componenti principali dell'immagine RGB usando il metodo PCA, e normalizzano ogni componente per essere usata dal classificatore *k-means*. Tuttavia, al fine di ottenere la segmentazione finale, sono necessarie altre operazioni come: filtraggio morfologico, riempimento dei buchi e *region growing*, rendendo tutto il processo molto sensibile ai parametri impiegati. Dal momento che utilizzare le informazioni del vicinato dei *pixel* produce un'enorme quantità di dati rispetto ad approcci di tipo globale o basati sull'intensità dei singoli *pixel*, in [25] altri autori fanno uso del metodo PCA per ridurre la dimensionalità dei vettori contenenti le caratteristiche del vicinato al fine di migliorare le performance computazionali. Gli autori sfruttano il *k-means* per la segmentazione accurata, a loro parere, di immagini provenienti da MRI, istologiche o acquisite da microscopia a fluorescenza.

---

Pur essendo computazionalmente costosa, la possibilità di utilizzare le informazioni sul vicinato, durante la segmentazione, è un tentativo che sembra essere valido e promettente per aumentare l'accuratezza del processo. Un recente lavoro [24] sfrutta la distribuzione locale di intensità dei *pixel* di una *patch* locale per raccogliere informazioni necessarie alla corretta classificazione di ogni *pixel*. Utilizzando un modello chiamato *Gaussian Mixture Model* (GMM), gli autori hanno sviluppato un metodo di classificazione dei *pixel* simile al *k-means* utilizzando la distanza di *Bhattacharyya* come metrica di somiglianza tra le distribuzioni GMM. I risultati sono incoraggianti rispetto ad altri algoritmi di segmentazione come: *mean shift clustering*, *multiphase active contour* e quelle basate *Markov Random Fields*.

L'uso di istogrammi applicati come dati di ingresso all'algoritmo *k-means*, per ottenere una segmentazione è stato impiegato in un recente lavoro [26]. Tale studio è stato condotto su immagini naturali ed il *k-means* è sfruttato per raffinare le segmentazioni ottenute dall'applicazione di un altro *k-means* usato sulle immagini rappresentate in diversi spazi colore. L'idea si basa sulla considerazione che ogni segmentazione ottenuta in uno spazio colore diverso produca parte del risultato di interesse, infatti per ogni spazio si possono osservare delle mancanze. Dalle diverse segmentazioni si può produrre un risultato migliore ottenendo una rappresentazione più verosimile della scena percepita. Questo è stato realizzato attraverso un algoritmo basato su misure di similarità tra istogrammi locali centrati intorno a ciascun *pixel* e all'istogramma di riferimento di ciascuna classe. I risultati mostrati sono comparabili con quelli ottenuti dagli algoritmi di segmentazione considerati lo stato dell'arte. È stato infatti possibile paragonare le performance degli algoritmi dal momento che i risultati sono stati ottenuti sul medesimo *dataset* di immagini. Tuttavia la sensibilità dei quattro parametri dell'algoritmo dello studio [26] sembra richiedere un *tuning* approfondito.

Un compromesso tra accuratezza e complessità di qualunque metodo sembra essere un obiettivo difficile da raggiungere dal momento che tutti i lavori sperimentali precedentemente citati fanno uso di elaborazioni, prima o dopo,

---

l'applicazione dell'algoritmo di segmentazione, richiedendo spesso parametri aggiuntivi. Le tecniche presentate fino ad ora sono tutte valide e sono un ottimo punto di partenza per una segmentazione di immagini, tuttavia non è possibile confrontarle [6]. La maggior parte degli autori, infatti, non fornisce il *database* di immagini su cui sono condotte le prove. I metodi descritti si basano inoltre su una conoscenza a priori dell'immagine da processare e quindi non possono essere utilizzati su altre tipologie. Infine non sono riportati i tempi di esercizio dei suddetti algoritmi rendendo impossibile verificare se i risultati possono essere ottenuti in tempi ragionevoli.

# Capitolo 4

## Approccio al problema di classificazione

La presente Tesi si propone di individuare aree percettivamente simili di immagini acquisite al microscopio, con l'intento di fornire un aiuto agli istopatologi nella classificazione dei preparati istologici. L'inquadramento del problema risulta alquanto complesso per i motivi di seguito spiegati.

Dal punto di vista umano, risulta del tutto normale rapportarsi con il mondo circostante assegnando a quello che vediamo un significato, ma tale processo mentale è del tutto sconosciuto ad un calcolatore. Le immagini, infatti, contengono informazioni che sono associate ad oggetti presenti nei nostri ricordi, ma nel calcolatore sono rappresentate come una matrice multi dimensionale di semplici numeri. Inoltre, le informazioni che otteniamo grazie alla vista hanno una particolarità che ci permette, in molti casi, di identificare oggetti tra loro diversi, addirittura anche se questi sono sconosciuti: il colore. Il colore non è una proprietà fisica degli oggetti ma è una proprietà percettiva generata dai segnali nervosi che i nostri fotorecettori nella retina mandano al cervello quando

---

assorbono radiazioni elettromagnetiche di determinate lunghezze d'onda ed intensità. Di conseguenza la rappresentazione del colore diventa un problema cruciale se si vuole utilizzare tale informazione. Come se ciò non bastasse, occorre considerare anche il mezzo di acquisizione: infatti, lo strumento, le condizioni di illuminazione e la sua calibrazione, cambiano da un laboratorio all'altro, con lo svantaggio di generare immagini, anche uguali, ma con diverse caratteristiche apparenti.

Poiché l'istopatologo riesce a pervenire ad una diagnosi a partire dall'immagine, non si può escludere di riuscire a realizzare un algoritmo *ad hoc*, in grado di imitarne il comportamento, sfruttando ciò che l'essere umano non ha: la capacità di ottenere misure esatte, la ripetibilità dell'analisi, l'oggettività. Tale algoritmo risulterebbe sicuramente un'ottima soluzione se, in un caso così complesso, chiunque, senza esperienza ed istruzione, guardando la stessa immagine riuscisse a pervenire al medesimo risultato, ma purtroppo ciò non accade. Infatti, servono numerosi anni di studio e molta esperienza per acquisire le competenze e le capacità necessarie ad effettuare una diagnosi corretta. Tuttavia, per contro in alcuni casi è possibile che anche persone non professionalmente formate riescano ad identificare regioni e strutture diverse all'interno dell'immagine, senza tuttavia riuscire ad assegnare a queste differenze un significato specifico, né sapere che cosa queste regioni rappresentino. La nostra mente è molto abile a riconoscere e a cercare somiglianze. L'immagine spinge la mente a ricercare dei *pattern* che si conoscono, ma se questi non sono presenti nei nostri ricordi, ci limitiamo a trovare somiglianze tra gli elementi presenti. Il grado di dettaglio dell'immagine e la dimensione della scena rappresentata, sono dei fattori di estrema importanza nel raggruppamento di oggetti percettivamente simili. Si pensi solamente ad un'immagine raffigurante un muro ed un'altra con solo il mattone di cui il primo è composto. Nel primo caso ci si concentra sull'oggetto muro come singola entità, anche se composto da mattoni, nella seconda tutta l'attenzione è nuovamente posta sul soggetto dell'immagine, che però è costituito dal mattone. Questo

---

semplice esempio spiega bene l'idea della *texture analysis* di cui si è già parlato precedentemente e di cui questa tesi fa uso.

#### 4.1 Il colore come *feature*

Il processo di scelta delle *feature* ha avuto inizio con la visualizzazione di un elevato numero di immagini provenienti da provini istologici. Considerando che la maggior parte di essi sono trattati con E&H per colorare nuclei e citoplasma, ho deciso di usare il colore come *feature* sulla quale basare l'analisi per il riconoscimento delle classi. La scelta del colore come *feature* non è l'unica possibile [9] [16] [17], tuttavia se ne avessi utilizzate altre, ad esempio, la dimensione, l'eccentricità e l'orientamento dei nuclei, sarebbe stato necessario effettuare una segmentazione molto accurata di questi ultimi che, come spiegato in precedenza, risulta piuttosto complessa. La scelta del colore è dunque vantaggiosa in quanto non richiede alcuna segmentazione preliminare.

#### 4.2 Utilizzo dell'istogramma nel processo di classificazione

Per confrontare sottoimmagini simili, utilizzando come *feature* il colore, si è pensato di impiegare l'istogramma. Come già detto in precedenza, l'istogramma ha la proprietà di poter confrontare zone diverse, praticamente in maniera indipendente dalle loro trasformazioni geometriche. Per questa ragione, dopo avere convertito le immagini da analizzare nello spazio Lab si è proceduto con il confronto dei loro istogrammi, sui tre canali, posti come un unico vettore. L'istogramma fornisce, infatti, la possibilità di avere un'informazione svincolata dalla posizione spaziale assunta da ogni singolo pixel dell'immagine di partenza rispetto all'immagine in analisi. Questa proprietà è di notevole importanza laddove le immagini da confrontare presentino il medesimo soggetto a cui sono state applicate trasformazioni geometriche quali rotazione o traslazione.

---

### 4.2.1 Confronto tra istogrammi

Il confronto tra istogrammi è una problematica piuttosto complessa, ma che permette di riconoscere immagini simili con uno sforzo computazionale ridotto.

Fig. 4.1 mostra due immagini simili.

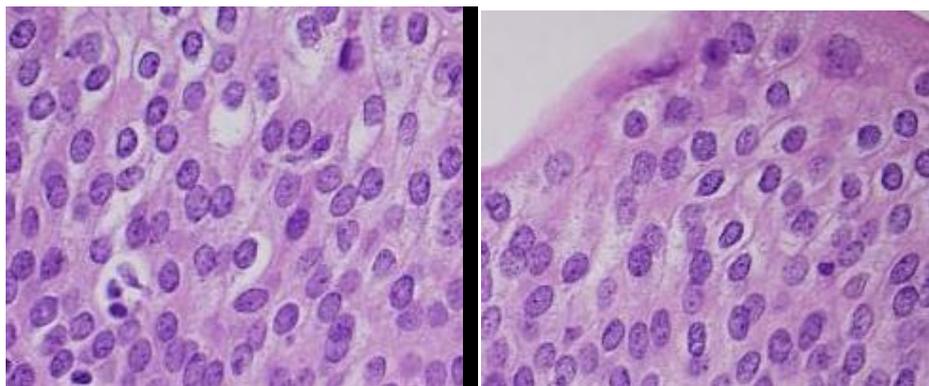


Fig. 4.1 Immagini che possono essere considerate simili

Un calcolatore non può “vedere” queste due immagini, ma le può solo elaborare come sequenze di byte che contengono informazioni riguardo ai pixel che le costituiscono. Non può essere comunque ipotizzabile, come potrebbe essere lecito pensare istintivamente, valutare la similarità tra le due immagini andando a confrontare le codifiche binarie di ogni singolo pixel estratto dall’una e dall’altra immagine. Questo genere di confronto non avrebbe nessuna valenza generale: basti pensare che, adottando tale strategia, confrontare due immagini di diverse dimensioni non sarebbe possibile, o almeno, qualsiasi operazione di confronto fornirebbe un risultato di significato difficilmente interpretabile.



Fig. 4.2 Trasformazioni geometriche della stessa immagine originale (in alto a sinistra)

Ad esempio, il confronto tra un'immagine ed una sua trasformazione geometrica (Fig. 4.2) non produrrebbe un riconoscimento positivo. Risulta quindi indispensabile trovare un metodo per confrontare due o più immagini o, nel mio caso, sotto-immagini, che non prenda in esame le loro bitmap, bensì effettui i confronti avvalendosi di una qualche forma di rappresentazione del loro contenuto che, possibilmente, risulti anche indipendente dalle dimensioni. In particolare, uno degli obiettivi di questo lavoro di tesi, è quello di riuscire a riconoscere immagini che presentino un contenuto informativo simile. In ragione di ciò, è naturale individuare un metodo per caratterizzare il contenuto di un'immagine in base alla distribuzione delle sue caratteristiche cromatiche. La rappresentazione del contenuto di un'immagine, facendo riferimento alle sole caratteristiche cromatiche, non tiene ovviamente in considerazione la forma, ovvero le informazioni legate alla distribuzione spaziale dei singoli pixel introducendo la possibilità che due immagini, all'apparenza molto differenti, possano comunque avere due rappresentazioni molto simili. D'altra parte, una rappresentazione del contenuto che tiene conto delle sole informazioni cromatiche di un'immagine, presenta un aspetto molto positivo: la facilità con il quale può essere implementato. A tal fine, si può definire una struttura idonea a fornire una rappresentazione del contenuto di un'immagine basata solo sulle caratteristiche cromatiche: *l'istogramma di colore di un'immagine*.

L'utilizzo di un istogramma come "descrittore", è una delle varie possibilità esistenti per descrivere il contenuto di un'immagine. Come già evidenziato,

---

questo tipo di rappresentazione non è necessariamente la più efficace, poiché non prende in considerazione alcuna informazione legata alla forma dell'immagine, ma ha il vantaggio di poter essere realizzata in modo abbastanza semplice.

Con questo tipo di rappresentazione del contenuto, può quindi accadere che due istogrammi associati ad immagini diverse siano uguali. Come premesso, tale evenienza è conseguenza diretta del fatto che, con questa rappresentazione del contenuto, non è presente nessuna informazione sulla forma dell'immagine. Ad esempio, le immagini riportate in Fig. 4.3, seppur diverse, hanno due istogrammi di colore uguali.



Fig. 4.3 Immagini diverse presentano lo stesso istogramma

Quello appena esposto rappresenta sicuramente un caso limite, poiché le immagini sintetiche prese in considerazione non rappresentano alcuna scena, oggetto o paesaggio, ma bensì due scacchiere, costruite appositamente con caselle di uguali colori disposte in ordine diverso, proprio per mettere in risalto il concetto espresso in precedenza, riguardo al fatto che non disponendo, all'interno dell'istogramma, di nessuna informazione sulla distribuzione spaziale dei pixel, può accadere di considerare uguali due immagini che in realtà sono completamente diverse.

Nella presente Tesi non è stato utilizzato l'istogramma di colore, poiché a fronte di un'immagine a colori codificata con 8 bit per canale, tale descrittore può rappresentare fino a 16 milioni di componenti. L'informazione necessaria alla

---

classificazione impiegata è stata ricavata dai tre istogrammi ottenuti dai tre canali dell'immagine a colori. A titolo di esempio in Fig. 4.4 si riportano in 9 immagini, tre per ogni regione simile.

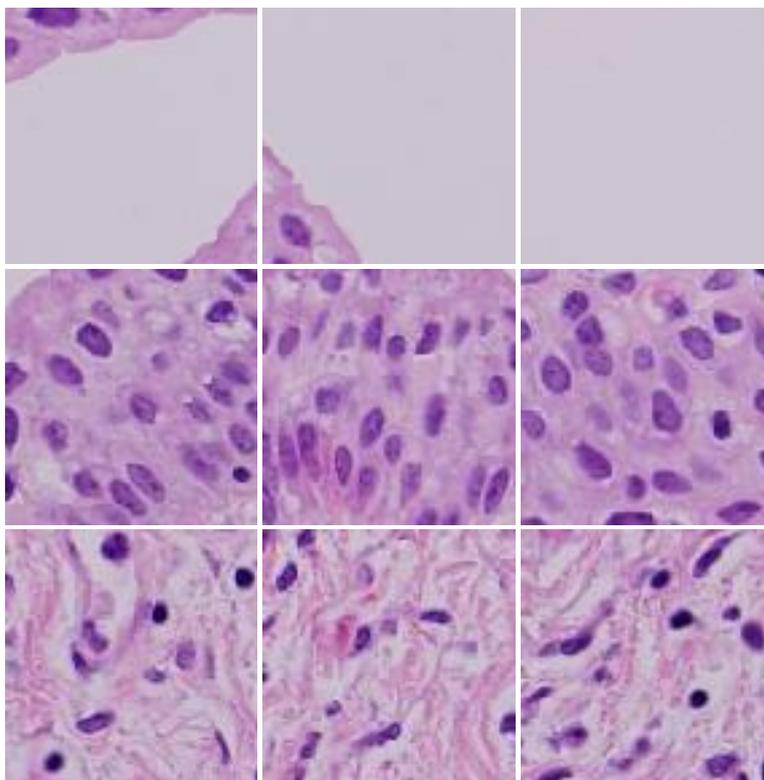


Fig. 4.4 Immagini di riferimento

Dalle immagini in Fig. 4.4 sono stati calcolati gli istogrammi e in Fig. 4.5 si riporta una rappresentazione bidimensionale della somma delle differenze di ogni canale nello spazio RGB a sinistra e Lab a destra.

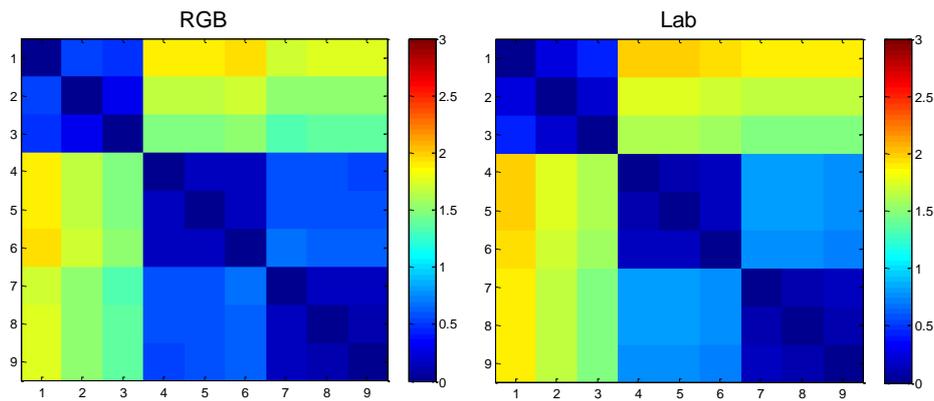


Fig. 4.5 Differenza tra istogrammi a destra nello spazio Lab e a sinistra in RGB

Una spiegazione più dettagliata e rigorosa è fornita nel paragrafo successivo.

#### 4.2.2 Misura di similarità tra istogrammi

L'approccio utilizzato nella risoluzione del problema in analisi è chiamato *Histogram Intersection*, e si rivela robusto se:

- lo sfondo dell'immagine non è uniforme;
- il soggetto dell'indagine è fotografato da diverse angolazioni;
- è presente una differenza di risoluzione tra le immagini.

Se da due immagini in scala di grigio si calcolano i rispettivi istogrammi [27],  $I$  e  $M$ , contenente ciascuno  $n$  bin, l'intersezione tra i due è definita come:

$$\sum_{j=1}^n \min(I_j, M_j) \quad 4.1$$

Il risultato dell'Eq. 4.1 indica il numero di pixel che presentano lo stesso valore in entrambe le immagini. Supponendo per semplicità immagini contenenti lo stesso numero di pixel, per ottenere un valore normalizzato tra 0 e 1, l'intersezione viene calcolata come segue:

---


$$H(I, M) = \frac{\sum_{j=1}^n \min(I_j, M_j)}{\sum_{j=1}^n M_j} \quad 4.2$$

La normalizzazione degli istogrammi, e quindi della loro intersezione, non va in alcun modo ad aumentare la capacità di discriminazione tra le aree dell'immagine, ma permette di confrontare immagini di diverse dimensioni. Il calcolo dell'intersezione tra istogrammi risulta inoltre molto semplice computazionalmente e facilmente applicabile all'algoritmo di classificazione *k-means*, in quanto applicare l'Eq.4.2 è equivalente ad impiegare come metrica la distanza *d city-block* definita nell' Eq. 4.3 come:

$$d = \sum_{j=1}^n |A_j - B_j| \quad 4.3$$

dove A e B sono gli istogrammi da confrontare. Se si considera *T* pari al numero dei pixel contenuti nell'immagine allora è possibile ritrovare l'equivalenza riportata nell'Eq. 4.4:

$$1 - H(I, M) = \frac{1}{2T} \sum_{j=1}^n |I_j - M_j| \quad 4.4$$

dove  $H(I, M)$  è calcolato secondo l'Eq. 4.2.

### 4.3 Scansione dell'immagine

L'immagine viene suddivisa in sotto-immagini in base alla disponibilità di calcolo degli strumenti a disposizione, in questo lavoro di Tesi le prove sono state compiute con un computer Acer Aspire 5738G con processore Intel Core Duo P7450 2.13GHz e memoria ram di 4GB. Di conseguenza, il numero di sotto-immagini non è fisso ed è possibile variarlo grazie al parametro *p* che permette di

---

ridurre il numero di sotto-immagini. Infatti, i centri delle sotto immagini sono definiti da due vettori che corrispondono alle coordinate sull'asse delle ascisse e delle ordinate. I centri delle sotto immagini sono equidistanti di  $p$  punti su entrambi gli assi, un esempio di suddivisione di un immagine è mostrata in Fig. 4.6, dove a destra i centri delle sotto immagini sono distanziati di  $p=1$  punti mentre a sinistra di  $p=2$  punti.

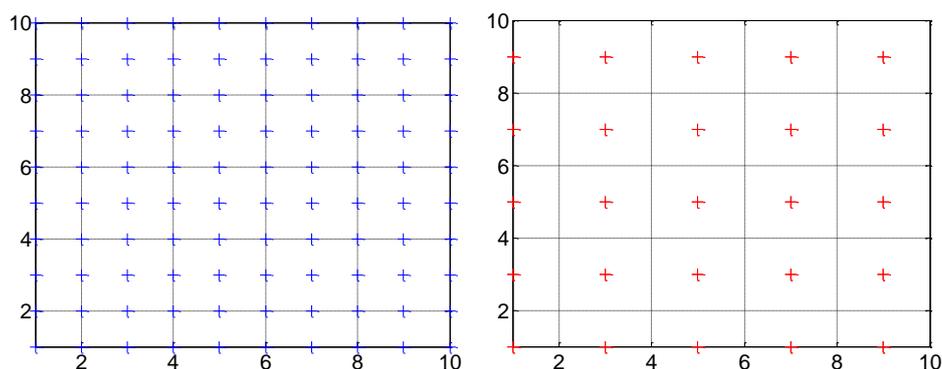


Fig. 4.6 Suddivisione di una matrice con  $p=1$  a sinistra e  $p=2$  a destra

Il valore di  $p$  deve essere un intero positivo e lo sforzo computazionale è pari all'inverso di  $p^2$  quindi valori di  $p$  maggiori di 1 peggiorano la risoluzione della classificazione. La risoluzione in questo caso è da intendersi come la distanza che intercorre tra i punti realmente calcolati segnati con il '+', mentre quelli mancanti verranno generati per interpolazione.

Le sotto-immagini sono state scelte, per semplicità, di forma quadrata con lato pari a  $2*f+1$ , dove  $f$  è un valore impostato dall'operatore. Tale valore è fondamentale ai fini di una buona classificazione e va scelto in base all'immagine che si sta osservando. Successivamente verrà descritto il protocollo definito e messo a punto per aiutare l'utente nella corretta identificazione del valore di  $f$ .

# Capitolo 5

## Algoritmo e implementazione

L'algoritmo realizzato nella presente Tesi è stato sviluppato in Matlab<sup>®</sup> (abbreviazione di *Matrix Laboratory*) è un ambiente per il calcolo numerico e l'analisi statistica che comprende anche l'omonimo linguaggio di programmazione, interpretato, creato dalla *MathWorks*. In questo ambiente di lavoro è possibile manipolare matrici, visualizzare funzioni e dati, implementare algoritmi, creare interfacce utente ed interfacciarsi con altri programmi. Inoltre, è presente un *image toolbox* dedicato alla manipolazione, elaborazione e visualizzazione di immagini. Proprio sfruttando alcuni degli strumenti forniti dal *toolbox* è stato implementato l'algoritmo di classificazione.

Più precisamente l'algoritmo sviluppato si compone di quattro parti fondamentali:

1. definizione dei parametri;
2. acquisizione delle *feature*;
3. classificazione;
4. salvataggio dei risultati.

---

In Fig. 5.1 è rappresentato un diagramma di flusso che spiega sinteticamente tutto l’algoritmo di classificazione, a sinistra, e il dettaglio delle chiamate a destra.

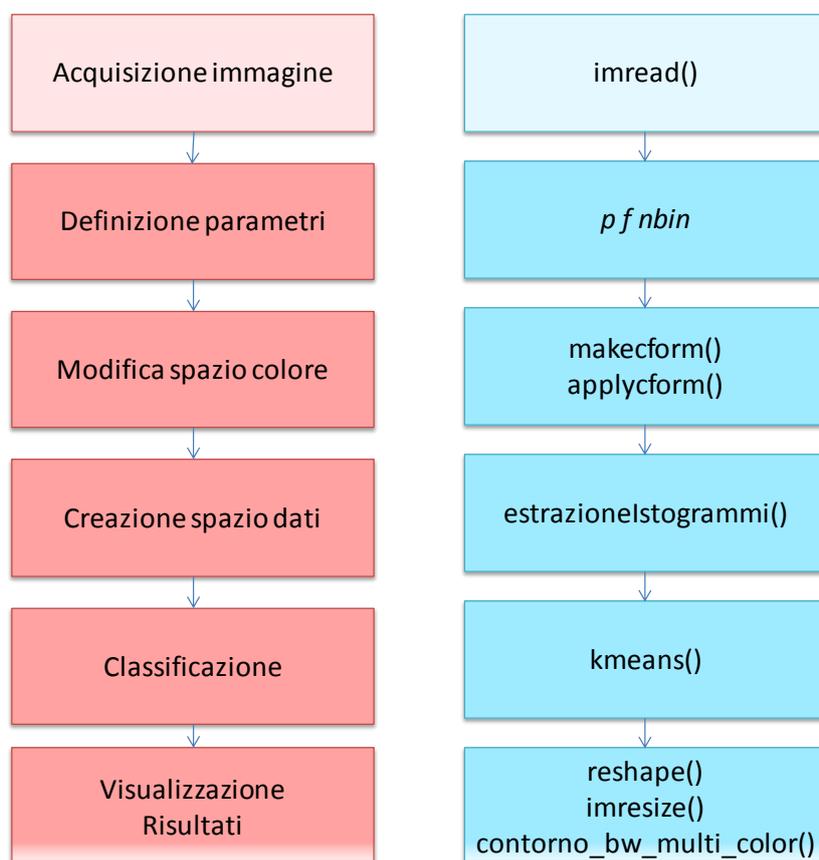


Fig. 5.1 Diagramma di flusso

## 5.1 Definizione dei parametri

I parametri necessari al funzionamento dell’algoritmo sono tre:

- $p$  definisce la griglia di punti, già mostrata in Fig. 4.6, da cui estrarre le *feature*;
- $f$  rappresenta la dimensione della finestra dalla quale estrarre la *feature*;
- $k$  indica le classi “percepite” dall’utente all’interno dell’immagine.

Il parametro più importante per una corretta classificazione è la dimensione della finestra  $f$  da cui vengono estratte la *feature*. Poiché la dimensione delle immagini

---

da classificare non è uguale e le scene cambiano in maniera significativa, sarebbe molto difficile inserire un valore a priori, senza alcun riferimento. Per facilitare la scelta di  $f$ , e non costringere l'operatore a dover acquisire competenze sulla sua relazione con la dimensione delle immagini, rischiando una scorretta classificazione, si è proceduto a definire un opportuno protocollo e creare un metodo di scelta che sfruttasse un approccio interattivo con l'immagine. Fig. 5.2 mostra un'immagine di preparato istologico con sovrapposta la finestra di ricerca.



Fig. 5.2 Esempio di immagine con sovrapposta la finestra di estrazione delle feature

Questo approccio è stato realizzato tramite alcune *function* già presenti in Matlab: *imread* permette di caricare i tipi più comuni di immagini (.png, .tif, .jpg,...), *imshow* permette di visualizzare a schermo un'immagine, *imrect* realizza una *drag box* modificabile in posizione e di dimensione. In modo del tutto arbitrario sono stati assunti come valori (10,10) e (50,50), rispettivamente, ma possono essere parametrizzati in base all'immagine originale. L'utente può trascinare e modificare la *drag box* avendo come riferimento l'immagine sottostante. La dimensione della *drag box* può essere modificata a piacimento mentre le sue proporzioni no dal momento che le sotto-immagini sono state scelte di forma

---

quadrata. Per ottenere una *drag box* con proporzioni fisse si usa il comando *setFixedAspectRatioMode* che mantiene il rapporto tra la dimensione del lato maggiore e quello minore. Una volta scelta la finestra di dimensione opportuna il comando di invio viene confermato da un *doppio click* da parte dell'utente e il comando *wait* si occupa di aggiornare la dimensione della finestra sulla base della selezione. Poiché la dimensione ottenuta è pari al lato della *drag box* per ottenere il parametro *f* si esegue la seguente formula:

$$f = \text{round}\left(\frac{\text{dimensione drag box} - 1}{2}\right) \quad 5.1$$

Di seguito è riportato il codice:

```
Img=imread('imgProva.tif');  
imshow(Img)  
h=imrect(gca,[10 10 50 50])  
setFixedAspectRatioMode(h,true)  
position=wait(h);  
dimensione=round((position(end)-1)/2)
```

L'ultimo parametro lasciato all'utente è la scelta del numero di classi percepite, tale parametro non è stato automatizzato poiché soggettivo e funzione del contenuto dell'immagine. Nel caso specifico, immagini di campioni istologici, provenienti anche dallo stesso tessuto possono contenere un diverso numero di classi Fig. 5.3.

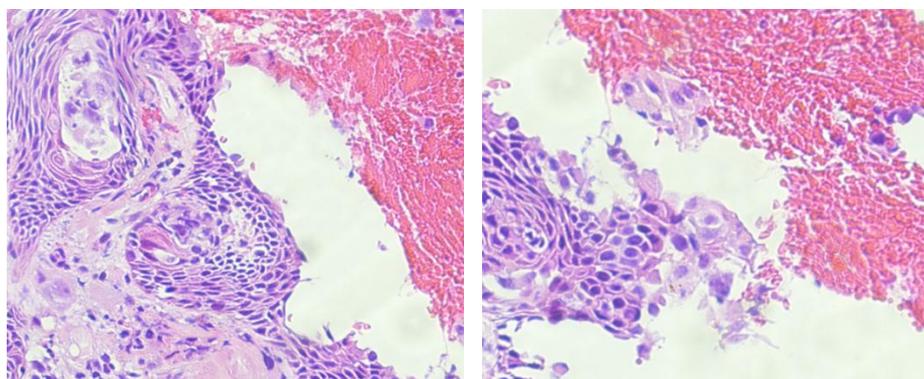


Fig. 5.3 Porzioni diverse della stessa immagine che contengono un numero di classi diverso

---

Questo deriva dall'ingrandimento del microscopio utilizzato per l'acquisizione delle immagini che solitamente permette di indagare porzioni molto piccole di tessuto. Indagare settori così ridotti porta a dover classificare regioni che si presentano in maniera chiara in alcune immagini e potrebbero non essere presenti in altre. Il parametro da assegnare per dichiarare il numero di classi è  $K$  ed è inserito tramite tastiera.

## 5.2 Modifica dello spazio colore

Prima di estrarre l'istogramma da ogni sotto immagine bisogna definire lo spazio colore. Le immagini caricate in Matlab sono disponibili in RGB ma, nel mio caso, è necessario cambiare tale spazio in LAB e XYZ. Tramite la funzione *makecform* è possibile creare una struttura contenente la trasformazione voluta. Per la trasformazione da RGB a LAB è necessario dare in ingresso a *makecform* la stringa '*srgb2lab*'. Senza altre informazioni il comando assegna al *White Point* le seguenti coordinate  $wpe=[0.9642,1,0.8249]$ , questi valori sono ricavati a partire dallo spazio XYZ e sono normalizzati rispetto all'asse Y di questo spazio, e di default sono inizializzati come:

```
xyz(:,1) = hex2dec('7b6b') / hex2dec('8000');  
xyz(:,2) = 1.0;  
xyz(:,3) = hex2dec('6996') / hex2dec('8000');
```

successivamente il comando esegue una trasformazione dallo spazio sRGB allo spazio XYZ con riferimento al *White Point* appena aggiunto, viene caricato il profilo ICC dello spazio sRGB e definito il verso della trasformazione, in questo caso da sRGB a XYZ, ora si hanno a disposizione le coordinate dei tre colori in sRGB mappate sullo spazio XYZ in particolare *Red*  $[0.436,0.222,0.014]$ , *Green*  $[0.385,0.716,0.097]$ , *Blue*  $[0.143,0.060,0.714]$ , ora il comando esegue una nuova trasformazione che relazione il *White Point* di default con quello definito dallo standard D50 di coordinate  $wps=[0.9642,1,0.8251]$ . Di default Matlab usa la matrice di conversione *Bradford* contenente i seguenti valori:

---


$$chad = \begin{bmatrix} 0.8951 & 0.2664 & -0.1614 \\ -0.7502 & 1.7135 & 0.0367 \\ 0.0389 & -0.0685 & 1.0296 \end{bmatrix} \quad 5.2$$

Di seguito viene proposto la procedura con cui Matlab implementa la conversione da uno spazio XYZ con *White Point* pari a *wps* ad uno pari a *wpe*, e la relativa trasformazione nello spazio originale sRGB

```
% Convert white points from XYZ to primaries:
rgbs = wps * chad';
rgbe = wpe * chad';
% Construct chromatic-adaptation matrix:
rgbscale = diag(rgbe ./ rgbs);
cdata.adapter = chad \ rgbscale * chad;
```

quindi si procede ad un adattamento dello spazio XYZ:

```
% Construct cform:
c = assigncform(@applychad, 'XYZ', 'XYZ', 'double',
cdata);
```

A questo punto si ha: il profilo sRGB dello spazio originale, il *White Point* nel sistema di riferimento D50, la matrice di trasformazione del *White Point* contenuta nella variabile *adapt.cdata.adapter*, e le coordinate delle componenti RGB mappate nello spazio XYZ.

Ora che mi sono ricondotto nello spazio XYZ che, come descritto precedentemente può essere preso come riferimento per tutti gli altri, Matlab sfrutta una successiva trasformazione, da XYZ a LAB.

### 5.3 Estrazione delle feature

A tal proposito è stata realizzata una *function* chiamata *EstrazioneIstogrammi.m*, che prende come input l'immagine da classificare *img\_in*, il parametro *p*, la dimensione della finestra *f*, ed un altro parametro definito *nbin*. Il parametro *nbin* serve a diminuire la quantità di dati da elaborare e di default viene assegnato pari a 16, questo valore è stato ottenuto empiricamente dopo numerose prove

---

ricercando un compromesso tra risultato riproducibile e sforzo computazionale. Tutte le prove ed i risultati proposti sono ottenuti con *nbin* pari a 16. Il programma inizia con la definizione delle coordinate dei centri delle sotto-immagini, che sono calcolati in Matlab dal seguente codice:

```
%% Vettori dei centri dove valutare gli istogrammi
vettore_passi_I=dim+1:p:dimensione_mosaico(1)-dim;
vettore_passi_J=dim+1:p:dimensione_mosaico(2)-dim;
num_passi=[length(vettore_passi_I
                length(vettore_passi_J)];
```

il *vettore\_passi\_I* ed il *vettore\_passi\_J* contengono, rispettivamente, i valori di riga e colonna da prendere come centri delle sotto-immagini mentre *num\_passi* contiene il numero totale di sotto-immagini per riga e colonna.

Gli istogrammi vengono salvati in un vettore preventivamente creato chiamato *Ch*:

```
%% Inizializzazione Ch
Ch=zeros(nbin,prod(num_passi),3);
```

*Ch* è un vettore multi dimensione che ha numero di righe pari a *nbin*, colonne pari al prodotto tra gli scalari contenuti nel vettore *num\_passi*, mentre la terza dimensione, pari a 3, è dovuta al numero di canali dell'immagine.

Il riempimento del vettore *Ch* è affidato al ciclo *for* di seguito riportato:

---

```

dim=f;
for m=1:3
k=1;
    for j=vettore_passi_J
        for i=vettore_passi_I
            img=img1(i-dim:i+dim,j-dim:j+dim,m);
            ch1=histc(img(:), linspace(0,255,nbin));
            Ch(:,k,m)=ch1;
            k=k+1;
        end
    end
end
fprintf(['Canale ' num2str(m) ' fatto \n' ])
end

```

L'immagine in ingresso viene scandita per colonne ed in ogni centro definito dalle variabili  $i$  e  $j$  si estrae una sotto immagine per ogni canale definito dalla variabile  $m$  che varia da 1 a 3.

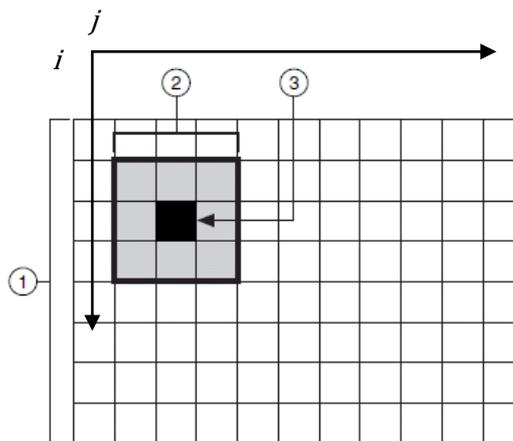


Fig. 5.4 Esempio di estrazione

Per ogni sotto-immagine si calcola l'istogramma tramite la funzione *histc* che richiede in ingresso la sotto-immagine sotto forma di vettore ed un altro vettore che indica come discretizzare l'istogramma.

Infine per rendere ottenere gli istogrammi normalizzati si esegue la seguente operazione:

---

```
%% Normalizzazione istogrammi
Ch=Ch/(2*dim+1)^2;
```

L'output di questa *function* è lo spazio dei dati, ovvero un vettore che contiene tutte le informazioni sugli istogrammi delle sotto-immagini.

## 5.4 Classificazione

Una volta generato lo spazio dei dati si procede ad una classificazione tramite l'algoritmo *kmeans*. Per utilizzare tale algoritmo è necessario fornire i dati sotto forma di matrice dove ogni riga contiene gli attributi di ogni sotto-immagine e non la matrice multidimensionale che ho creato. Scomponendo nei tre canali gli istogrammi associati alle sotto-immagini e ricomponendoli come segue si possono ottenere i dati nella maniera corretta:

```
%% Sistemazione dati
Ch1=Ch(:, :, 1);
Ch2=Ch(:, :, 2);
Ch3=Ch(:, :, 3);
dati=[Ch1', Ch2', Ch3'];
```

Matlab fornisce un'implementazione dell'algoritmo *kmeans* che presenta numerose opzioni, di default utilizza la distanza euclidea ed esegue un unico tentativo nella ricerca della soluzione. Queste opzioni sono state modificate assegnando alcune regole. In particolare, la distanza utilizzata è quella denominata *city-block*, già definita nell' Eq. 4.3 e, poiché l'algoritmo è dipendente dai valori iniziali, che sono scelti casualmente dallo spazio dei dati, si è scelto di eseguire 3 prove. La sintassi è di seguito riportata:

```
opts = statset('Display', 'final');
[cluster_membs cluster_center]=kmeans(dati, K, ...,
    'distance', 'cityblock', ...,
    'replicates', 3, 'Options', opts);
```

$K$  è il numero di classi percepite dall'utente ed è pari al numero di classi in cui suddividere lo spazio dei dati.

---

### 5.4.1 Funzionamento dell'algoritmo *k-means*

Il *kmeans* inizia con il riconoscere le opzioni inserite, in questo caso il tipo di distanza, il numero di esecuzioni ed il numero di cluster da individuare. I cluster possono essere inizializzati a partire da:

- campioni presi a caso dallo spazio dei dati;
- campioni uniformemente distribuiti nello spazio dei dati;
- una scansione preliminare del 10% dello spazio dei dati;
- una matrice data in ingresso dall'utente.

Nel presente caso si è utilizzata una inizializzazione casuale. L'algoritmo tenta di suddividere lo spazio dei dati in  $K$  classi che presentano la minima varianza intra-classe rispetto ad un centroide secondo la metrica scelta.

L'inizializzazione dei centroidi avviene nel modo seguente:

```
%% Inizializzazione dei centroidi
case 'sample'
    C = X(randsample(n, k), :);
    if ~isfloat(C)           % X may be logical
        C = double(C);
    end
```

$C$  contiene ora  $K$  vettori. Ottenuti i centroidi, è necessario calcolare la distanza di ogni punto dello spazio dei dati  $X$  che ha dimensione pari a quella dei dati in ingresso alla funzione, questo lo si fa mediante la function *distfun*:

```
%% Distanza di ogni punto rispetto ai centroidi
D = distfun(X, C, distance, 0);
```

la distanza *city-block* viene calcolata nel seguente modo:

---

```

case 'cityblock'
    for i = 1:nclusts
        D(:,i) = abs(X(:,1) - C(i,1));
        for j = 2:p
            D(:,i) = D(:,i) + abs(X(:,j) - C(i,j));
        end
        % D(:,i) = sum(abs(X - C(repmat(i,n,1),:)), 2);
    end
end

```

dove *nclust* è pari a *K*. Il vettore *D* ha dimensione pari a  $X \times K$  e contiene in ogni colonna la distanza per ogni punto dello spazio rispetto ad uno dei cluster. Ora vengono assegnati i punti dello spazio dei dati in modo che la distanza dai *cluster* sia minima, *idx* contiene l'assegnazione dei punti dello spazio ai *cluster* e *m* è un vettore contenente il numero di punti assegnato ad ogni cluster.

```

% Assegnazione
[d, idx] = min(D, [], 2);
m = accumarray(idx, 1, [k, 1]);

```

finita l'assegnazione vengono ricalcolati i centroidi tramite la *function gcentroids*:

```

%% Calcolo dei nuovi centroidi
[C(changed, :), m(changed)] = gcentroids(X, idx,
changed, distance)

```

I nuovi centroidi sono assegnati ai valori mediani degli elementi che compongono il cluster. La procedura è iterativa fino a quando la somma delle distanze dei dati rispetto ai centroidi dei *cluster* non aumenta. L'output della *function* è un vettore di interi di dimensione pari al numero di dati iniziali, dove il numero rappresenta il cluster di appartenenza.

## 5.5 Salvataggio e visualizzazione

Alla fine della classificazione si ottiene un vettore di interi che va da 1 a *K* di dimensione pari allo spazio dei dati. Per visualizzare i risultati, sottoforma di matrice, è necessario disporli nella loro forma originale. Poiché i dati iniziali sono stati acquisiti scandendo la matrice per colonne allo scopo di risistamarli è necessario utilizzare il seguente comando:

---

```
% Trasformazione vettore -> matrice
O=reshape(cluster_membs,size(matIdx));
```

$O$  contiene la matrice che è possibile visualizzare sotto forma di immagine, ma non ha la stessa dimensione dell'immagine di partenza. Per poter confrontare la classificazione è necessario ridimensionare  $O$  in modo opportuno;

```
% Ridimensionamento
O_RESIZE=zeros(size(mosaicoIMG(:,:,1)));
O_RESIZE(dimensione+1:end-dimensione,dimensione+1:end-
dimensione)=imresize(O,size(mosaicoIMG(:,:,1))-
2*dimensione,'nearest');
```

il comando *imresize* permette di modificare le dimensioni di un'immagine e l'opzione *'nearest'* definisce il metodo di interpolazione.

Infine, ho creato una *function* chiamata *contorno\_bw\_multi\_color.m* che prende in ingresso un'immagine ed una matrice di interi di pari dimensioni e fornisce come output l'immagine iniziale e i contorni delle classi ritrovate. Fig. 5.5 è un esempio di output dell'algorithmo sviluppato.

---

```
% Visualizzazione contorni  
contorno_bw_multi_color(mosaicoIMG,O_RESIZE)
```

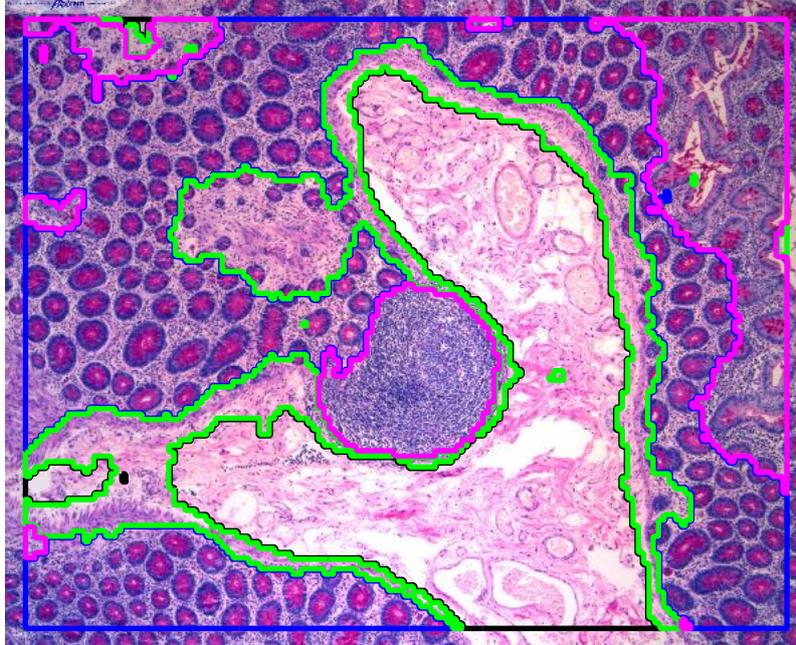


Fig. 5.5 Esempio di immagini “contornata”, evidenziando la suddivisione in classi

La procedura si conclude con il salvataggio nella matrice *O\_RESIZE* all’interno della cartella *Risultati* creata appositamente.



# Capitolo 6

## Esperimenti

In questo Capitolo sono contenute tutte le prove condotte sull'algoritmo, sia per misurare le sue performance, in termini di tempo impiegato per ottenere una classificazione, che per quanto riguarda la bontà della classificazione.

### 6.1 Performance dell'algoritmo

Il parametro  $p$  permette di classificare agevolmente immagini molto grandi senza necessariamente ridimensionarle, questo perché, come detto in precedenza indica la distanza tra il centro di una sotto immagine e la successiva. Tale parametro è scelto in base alla potenza di calcolo a disposizione dell'utente: se, ad esempio, si esamina un'immagine con dimensione pari a  $300 \times 300$  *pixel* e si imposta  $p=1$  si otterranno circa 90.000 sotto-immagini, per un'immagine  $600 \times 600$  circa 360.000 e per una  $1200 \times 1200$  circa 1.440.000. Per valutare le performance del software tutte le prove sono state condotte sullo stesso hardware: un computer portatile Acer Aspire 5738G con processore Intel Core Duo P7450 2.13 GHz e 4 GB di Ram DDR3. Le prove contenute in tabella presentano gli stessi parametri indicati nella prima riga eccetto  $p$ , indicato nella seconda. Ogni tabella contiene i risultati

espressi in secondi del tempo necessario alla classificazione. Nella colonna viola chiaro è indicata la dimensione dell'immagine su cui è eseguito l'algoritmo.

Risultati ottenuti con $f=25$ , $nbin=16$ , $cluster=4$			
$p$	2	4	8
<b>1273x1056</b>	129,6	42,5	8,1
<b>1000x800</b>	113,6	27,7	5,3
<b>1000x1250</b>	226,1	54,6	10,7

La dimensione dello spazio dei dati al si può ottenere conoscendo: la dimensione dell'immagine,  $f$  e  $p$  dalla Eq. 6.1 qui di seguito riportata:

$$spazio\ dati = round\left(\frac{(W - 2 * f) * (H - 2 * f)}{p}\right) \quad 6.1$$

Dove  $W$  è la larghezza dell'immagine e  $H$  la sua altezza espressa in pixel.

Sperimentalmente si può notare che l'algoritmo esegue una classificazione relativamente veloce (circa 20 secondi) su uno spazio dati di circa 250.000 sotto immagini, ma per *data set* più grandi i tempi aumentano in maniera esponenziale di seguito è riportato un grafico dove sull'ascisse è riportato il parametro  $p$  e sulle ordinate il tempo in secondi della classificazione dell'immagine di dimensione 1273x1056 con  $f=25$ ,  $nbin=16$ ,  $cluster=4$ .

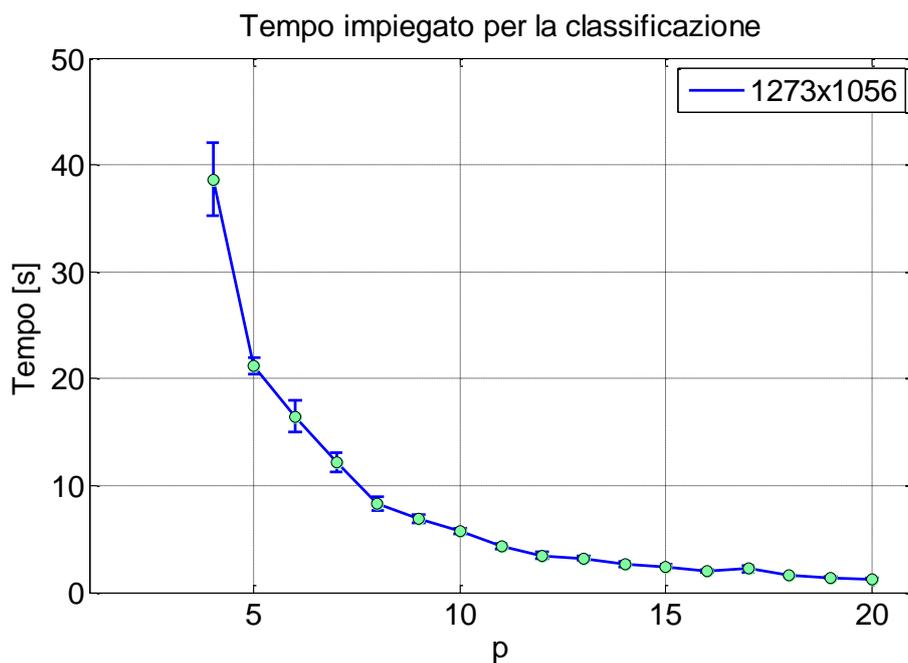


Fig. 6.1 Andamento del tempo impiegato per la classificazione in funzione del parametro  $p$

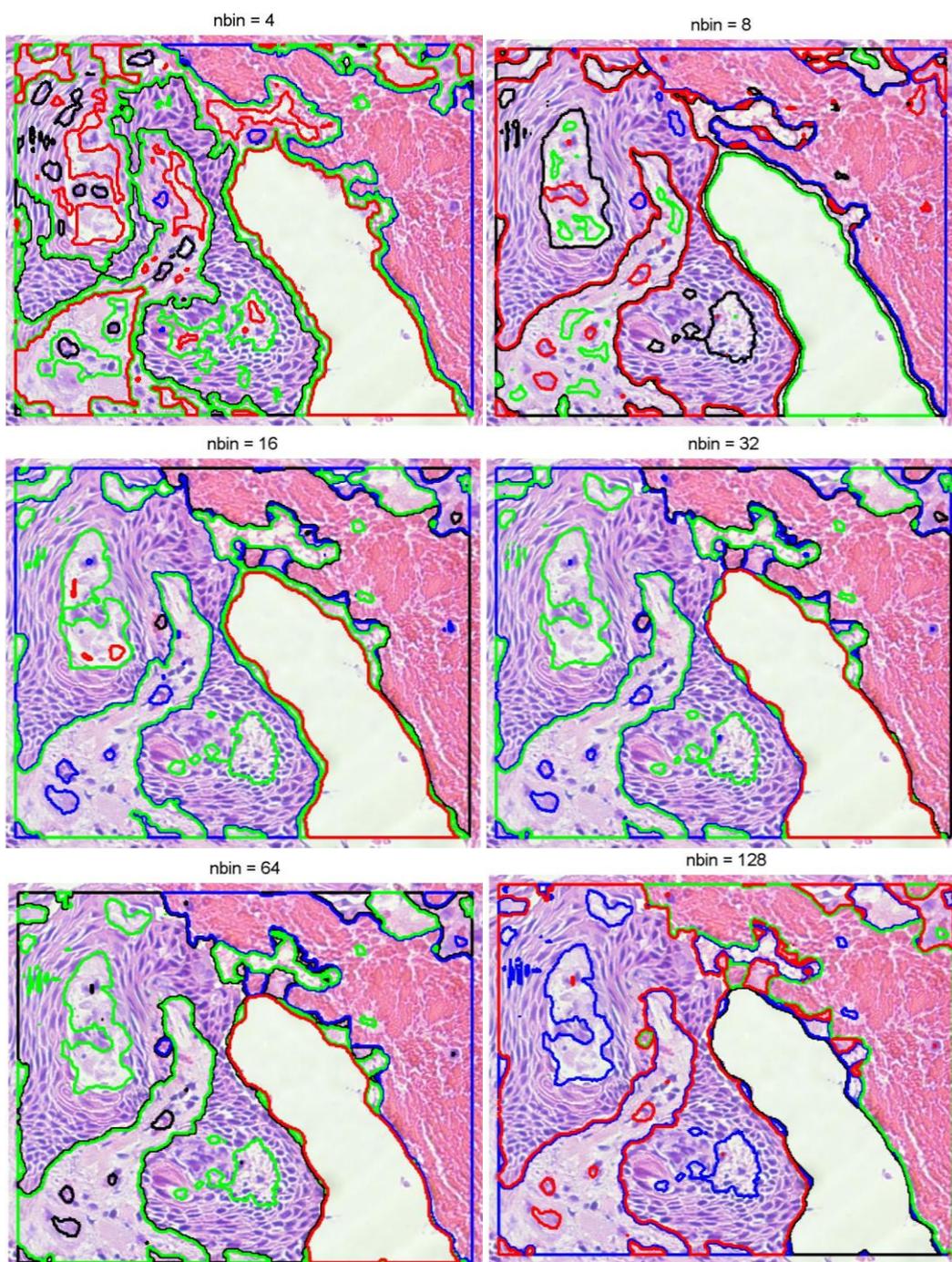
Potendo agire su  $p$ , è possibile indagare immagini molto grandi in tempi ragionevoli e ciò risulta molto utile sia per ottenere risultati preliminari sulla classificazione sia perché non viene persa alcuna informazione, cosa che invece accade se le immagini vengono ridimensionate. Inoltre non dover ricorrere al ridimensionamento dell'immagine per diminuire lo spazio dei dati permette di ottenere un'informazione statistica, sotto forma di istogramma, più robusta.

## 6.2 Il parametro $nbin$

L'algoritmo di classificazione presentato in questa Tesi, si basa esclusivamente sull'individuazione di cluster ottenuti partendo dagli istogrammi delle sottoimmagini. Tutte le immagini classificate sono codificate ad 8 bit ciò significa che ogni pixel può presentare, su ogni canale, un valore di grigio che va da 0 a 255. L'istogramma, ottenuto su ogni canale, perciò, va da un valore minimo pari a 0 ad un massimo di 255. Classificare le sottoimmagini con 3 istogrammi che presentano 256 valori ognuno risulta alquanto oneroso in termini di tempi di

---

calcolo. Per ottenere un compromesso tra tempo di calcolo e bontà della classificazione è stata classificata la medesima immagine mantenendo fissi i valori di  $p=4$  e  $f=25$ , e si è proceduto con un'analisi visiva dei risultati mostrati in Fig. 6.2.



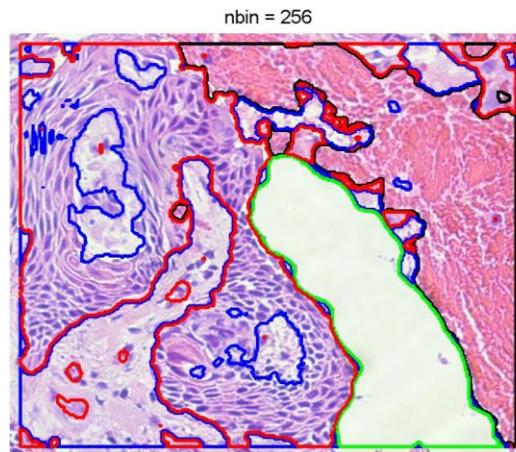


Fig. 6.2 Classificazione della medesima immagine con  $p=4$   $f=25$  al variare di  $nbin$

I risultati ottenuti suggeriscono che utilizzare valori troppo piccoli, 4 e 8, per la discretizzazione dell'istogramma portano ad un cattivo riconoscimento delle regioni simili, mentre con valori superiori si ha un riconoscimento sostanzialmente corretto. La scelta di utilizzare solo 16 *bin* per la discretizzazione deriva sia dal tempo impiegato per ottenere il risultato ma anche della dimensione della finestra utilizzata. La dimensione delle sotto immagini è un fattore molto importante per ottenere un informazione statistica robusta, infatti, nell'esempio di classificazione in Fig. 6.2,  $f$  ha valore 25. Il numero di *pixel* che costruiscono l'istogramma per la classificazione è facilmente ricavabile dall'Eq. 6.2.

$$n_{pixel} = (2 * f + 1)^2 \quad 6.2$$

Questo significa che ogni sotto immagine è descritta da un tre istogrammi di 2601 punti distribuiti su di un intervallo di  $nbin$ . Utilizzare un elevato numero per  $nbin$  può portare, per finestre piccole ad intaccare l'informazione statistica dell'istogramma generato e per questo in tutte le prove  $nbin$  è stato impostato come un parametro fisso e pari a 16.

---

### 6.3 *Testbed*

A seguito dell'applicazione dell'algoritmo di classificazione ad un *data set* di 39 immagini sono emersi risultati interessanti ed i più significativi verranno qui di seguito commentati. Tutti i risultati ottenuti e presentati sono stati realizzati con un Acer Aspire 5738G con processore Intel Core Duo P7450 2.13 GHz e 4 GB di Ram DDR3, il numero di regioni percepite è stato ottenuto visualizzando le immagini alla massima luminosità offerta dal monitor, il parametro  $p$  è stato scelto in funzione della dimensione dell'immagine originale per quelle non ridimensionate ed è invece fisso, pari a 4, per le immagini di test,  $nbin$  è sempre fisso a 16 e il programma Matlab 2011.

Le immagini sono state ottenute da vetrini istologici di numerosi tessuti sia sani che patologici, tutte però sottostanno ai seguenti criteri:

- sono ottenute a basso ingrandimento;
- contengono almeno 2 aree percettivamente differenti;
- non devono essere presenti dettagli di piccole dimensioni;

I requisiti appena citati servono per non identificare singole entità classificandole come regioni, ma per individuare un *pattern* che si ripresenti. Inoltre classificare immagini che presentano un'unica area non risulta infatti di interesse.

### 6.4 Validazione della classificazione

Per validare la classificazione è stato realizzato un programma Matlab che sottopone al giudizio dell'utente un *data set* di immagini che soddisfano i seguenti criteri:

- ogni immagine presenta almeno due classi distinguibili;
- le immagini siano diverse e rappresentative di una'ampia varietà;
- il *pattern* da individuare non sia banale.

All'utente vengono presentate, in sequenza, tutte le 39 immagini del *data set*. Per ciascuna di esse egli deve, innanzitutto, ridimensionare la finestra (*drag box*) che l'algoritmo utilizza per identificare l'area dell'immagine, in termine di numero di

---

pixel, a cui viene associato l'istogramma necessario al software per procedere con la classificazione. Si noti che questo è un passo estremamente importante per un corretto utilizzo dell'algoritmo di classificazione, infatti la scelta di un errato dimensionamento della finestra  $f$  può generare una somiglianza inesatta tra le classi. L'algoritmo presentato in questa Tesi deve poter lavorare con un'ampia casistica di immagini che possono variare sia in contenuto che in dimensioni, due di queste, presentate nel corso del test, sono visibili in Fig. 6.3.

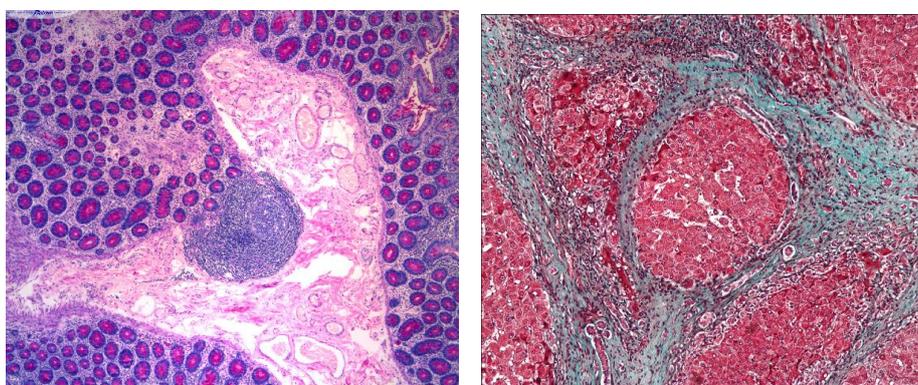


Fig. 6.3 Esempio di immagini istologiche di test

L'ampia variabilità di colore, dimensione e risoluzione dei soggetti presenti nelle immagini di test non consente di definire, a priori, una dimensione della finestra che possa estrarre i dati necessari ad un corretto funzionamento del classificatore. Qualora ci fosse, all'interno di tutte le immagini del *data set*, la stessa tipologia di soggetto di indagine, sarebbe possibile invece definire una finestra di dimensione fissa.

---

Modifica la dimensione del rettangolo e doppio click per confermare

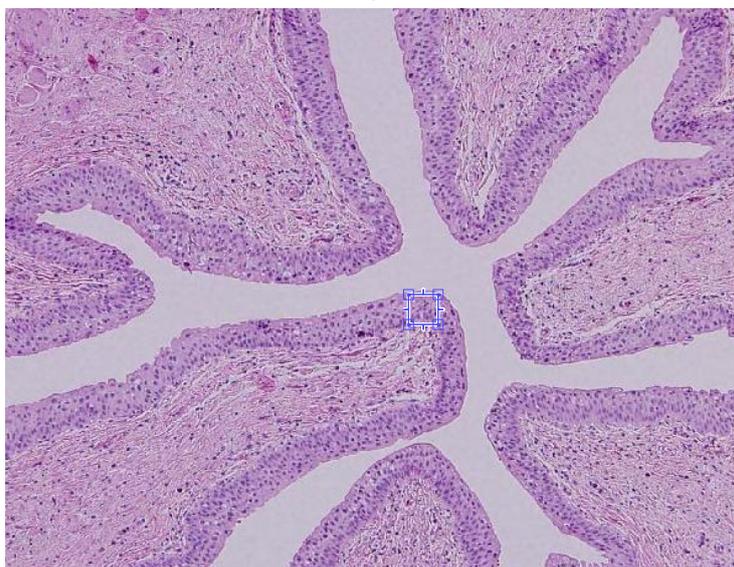


Fig. 6.4 Corretto dimensionamento della *drag box*

Identificata la dimensione della finestra (Fig. 6.4), facilmente modificabile con il mouse, l'utente deve specificare il numero di classi percettivamente differenti all'interno dell'immagine. Fatto ciò l'algoritmo propone la sua classificazione e, se il risultato è soddisfacente ed in linea con quello dell'utente, quest'ultimo deve fornire una valutazione numerica, da 1 a 10, in merito alla bontà della classificazione (di cui si parlerà in seguito) ed, eventualmente, lasciare un commento. Qualora la classificazione non risultasse soddisfacente l'utente può provare ad ottenerne una migliore cambiando la dimensione della finestra congiuntamente o separatamente al numero di classi percepite e verificando come si comporta l'algoritmo. Per svolgere l'intera procedura è stato realizzato un protocollo guida per l'utente al fine di chiarire qualsiasi dubbio in merito all'utilizzo del test che sarà illustrato qui di seguito.

## 6.5 Guida al test

La guida allegata al test si pone l'obiettivo di guidare l'utente nella scelta della dimensione più opportuna da attribuire alla finestra al fine di ottenere un buon

---

risultato nel lavoro di classificazione in regioni di immagini istologiche. A titolo di esempio sono riportate differenti tipologie di immagini che saranno classificate.

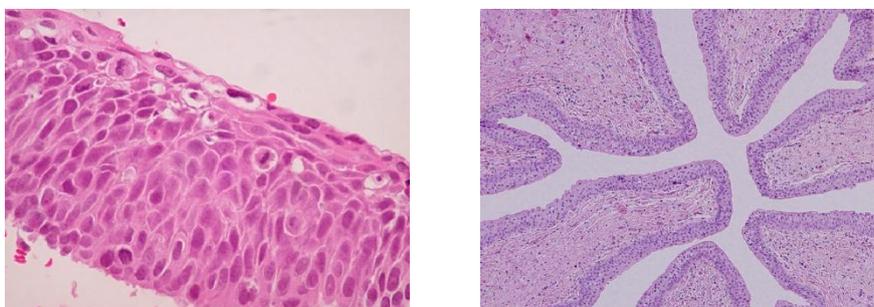


Fig. 6.5 Immagini di training

Come si può notare esse sono molto differenti, ma entrambe possono essere suddivise in macro-aree: due regioni sono osservabili nell'immagine di sinistra mentre in quella di destra tre.

Nel corso della guida sono riportati alcuni esempi significativi per chiarire qual è il modo di procedere per una corretta impostazione delle dimensioni della finestra e come modificarla per rendere la classificazione il migliore possibile.

L'immagine riportata in Fig. 6.6 mostra le tre classi percettivamente differenti tramite 3 frecce colorate. Si noti che il numero di classi può variare a seconda dell'utente, tuttavia, si ritiene che, in questo caso, tre sia il numero più adatto e condivisibile da un vasto numero di utenti. La freccia rossa indica, qualitativamente, la regione di colore uniforme 'bianco', la freccia nera indica le cellule rosa e quella blu quelle più viola.

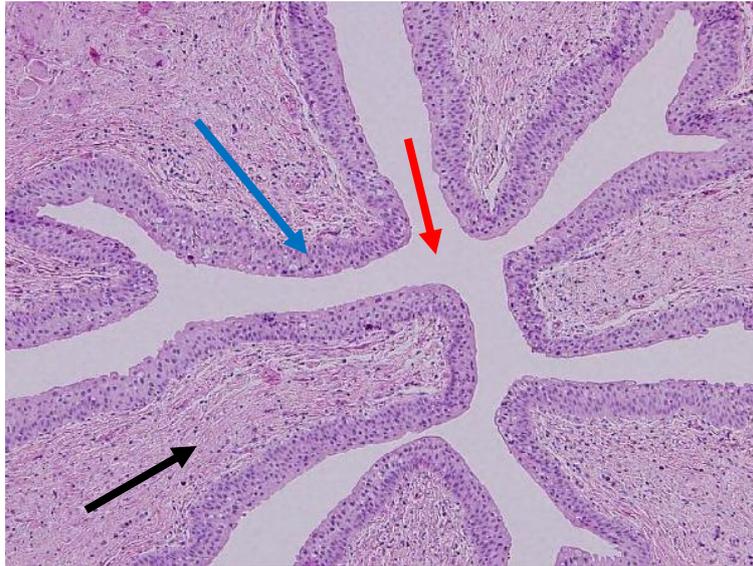


Fig. 6.6. Immagine di *training* 1 - Regioni percettivamente differenti

La finestra deve riuscire a contenere un campione rappresentativo della regione più piccola che si vuole andare ad individuare. Ciò significa che una finestra di dimensione pari a quella mostrata in Fig. 6.7 è troppo grande poiché al suo interno è presente più di una classe.

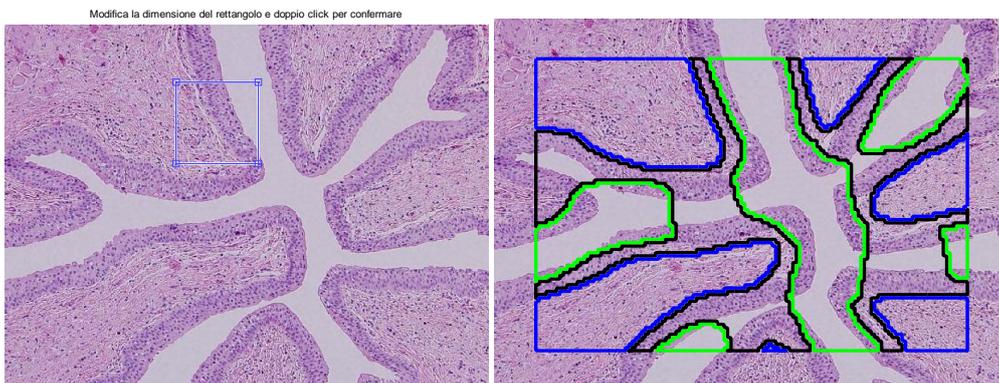


Fig. 6.7 Finestra e relativa classificazione (Grande)

In Fig. 6.8 è invece riportato un esempio di sotto dimensionamento della finestra, che conduce ad una errata classificazione, in quanto il campione individuato contenuto nella *drag box* non risulta rappresentativo dell' area considerata.

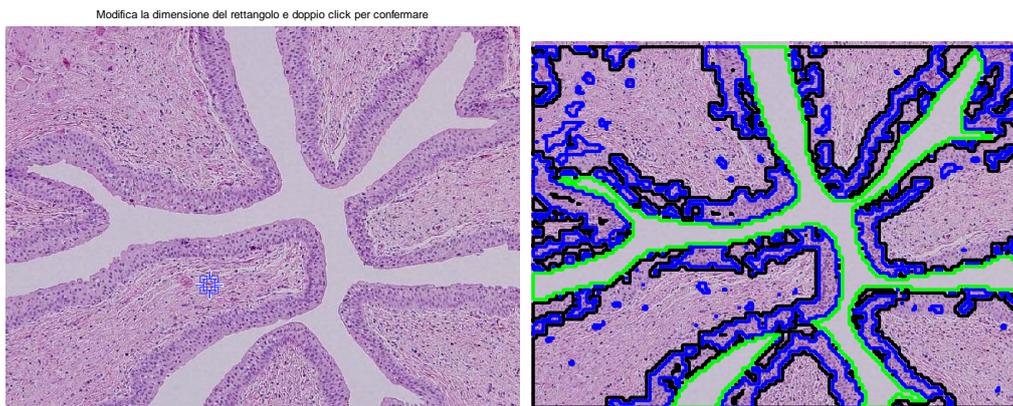


Fig. 6.8 Finestra e relativa classificazione (Piccola)

Un dimensionamento corretto potrebbe essere quello presentato in Fig. 6.9. In questo caso la finestra ha una dimensione tale da riuscire a contenere, in tutte le regioni percepite, una porzione rappresentativa della classe di appartenenza.

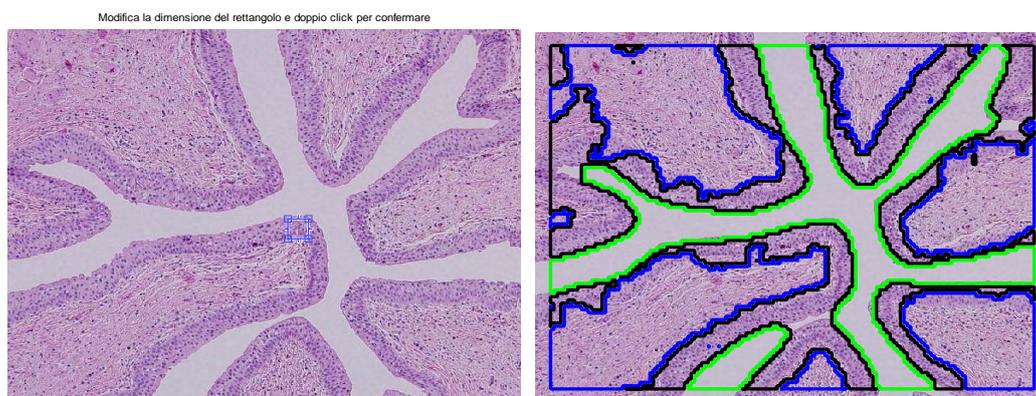


Fig. 6.9 Finestra e relativa classificazione (Giusta)

In Fig. 6.10 è riportato secondo esempio di immagine in cui risulta un po' più complicato definire una dimensione della finestra che riesca ad estrarre correttamente tutte le classi.

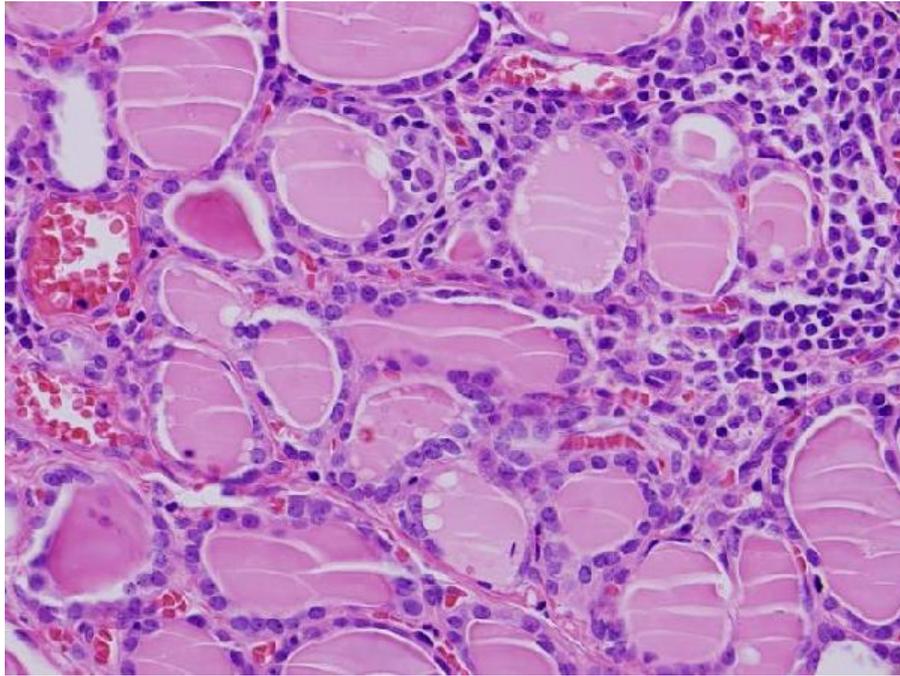


Fig. 6.10 Immagini di *training 2*

Il numero di classi percettivamente differenti non è pressoché certo come nel caso precedente. In questo esempio, utenti diversi, potrebbero identificare un numero di classi significativamente differente.

Nel caso in cui l'utente percepisca come classe quella composta dai nuclei viola, è necessario utilizzare una finestra piccola come quella proposta in Fig. 6.11.

---

Modifica la dimensione del rettangolo e doppio click per confermare

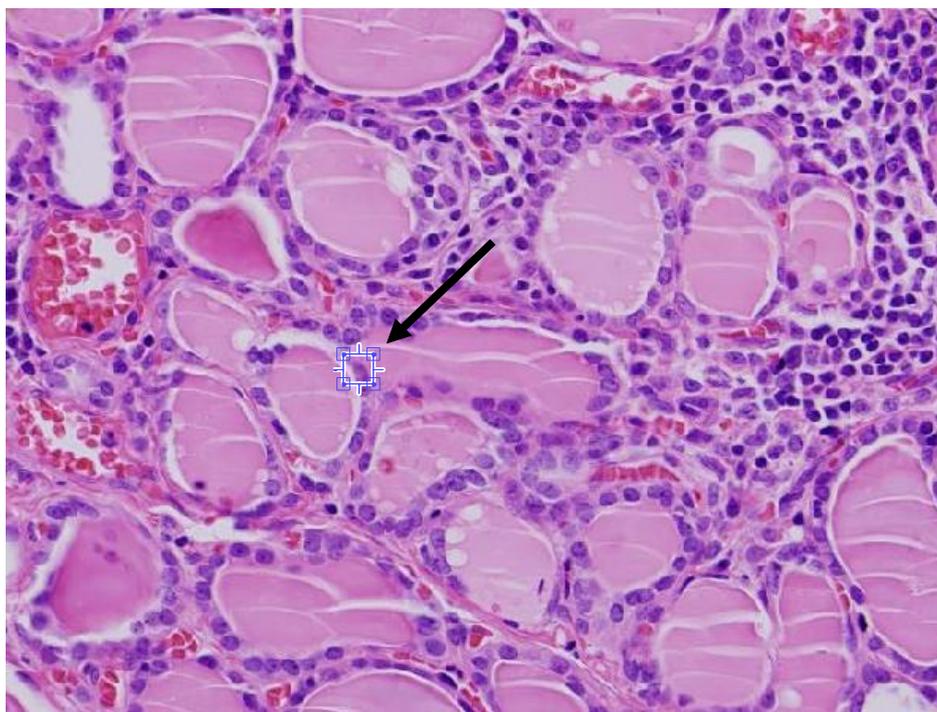


Fig. 6.11 Dimensione della finestra corretta per discriminare i nuclei

Il risultato della classificazione con 2 classi è visibile in Fig. 6.12: in una classe sono presenti i nuclei mentre nell'altra sono presenti le aree uniformi viola.

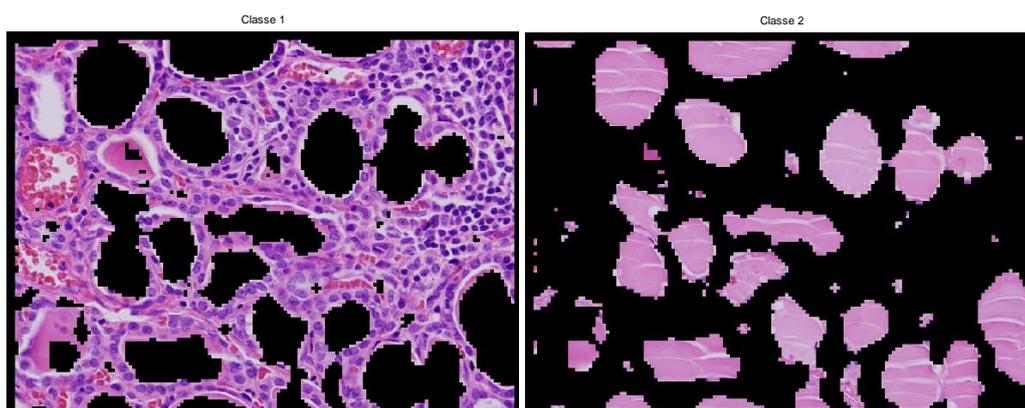


Fig. 6.12 Risultato della classificazione

---

In Fig. 6.13 è visibile un ultimo esempio di immagine di test. Anche in questo caso il numero di classi percepite non risulta univoco, infatti la maggior parte degli utenti indica in due o tre il numero di macro-aree differenti.

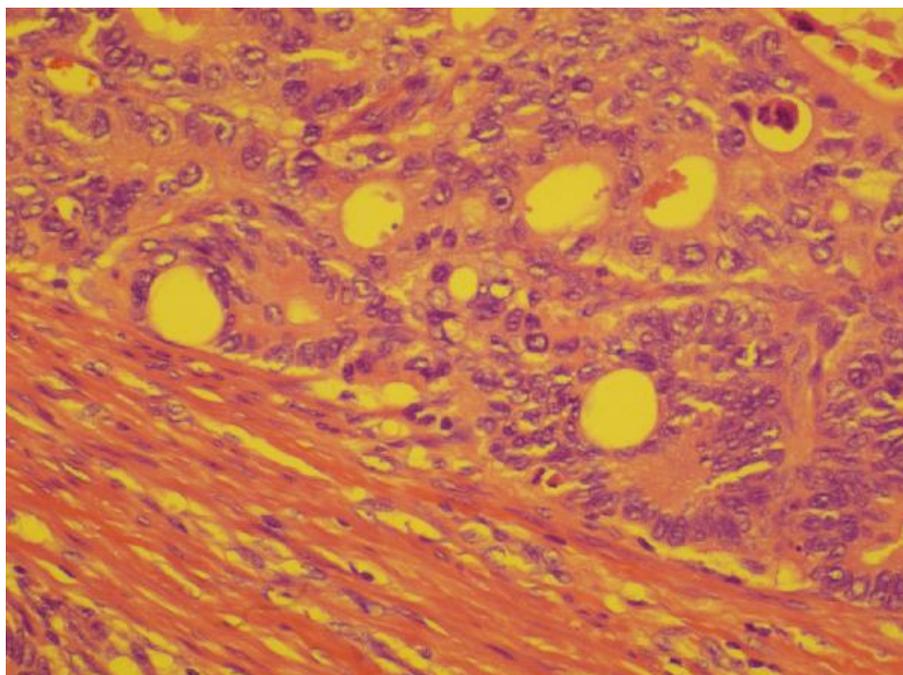


Fig. 6.13 Immagine di *training* 3

Nel caso se ne individuino solo 2 è meglio utilizzare una finestra grande come mostrato in Fig. 6.14, in modo da evitare una sovra segmentazione delle regioni.

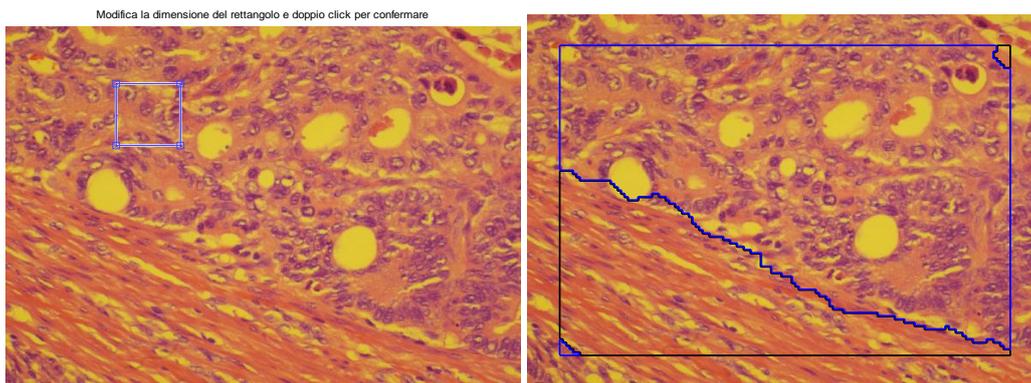


Fig. 6.14 Classificazione mediante 2 classi

Nel caso in cui la terza classe individuata sia quella dei ‘gialli’ allora sarà necessario impostare una finestra un po’ più piccola per non rischiare che questa classe venga trascurata, un esempio di finestra è mostrato in Fig. 6.15.

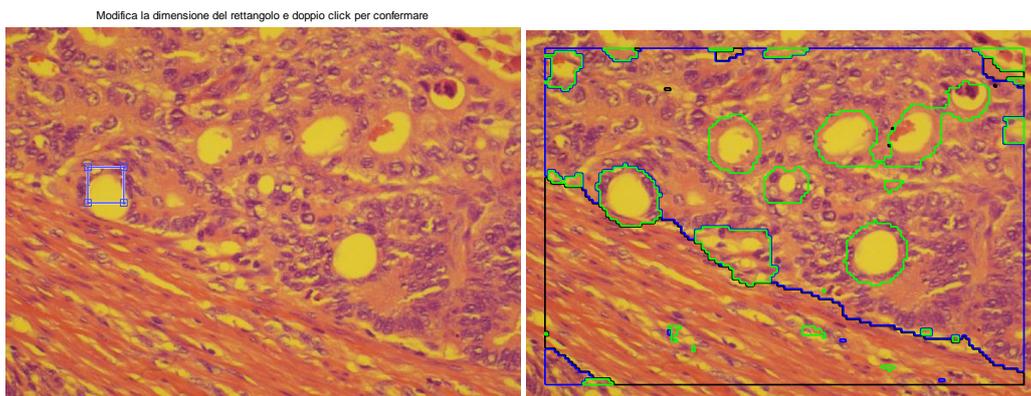


Fig. 6.15 Classificazione mediante 3 classi

Infine vale la pena spiegare in che modo l’operatore deve esprimere il giudizio sulla bontà della classificazione. Per effettuare la valutazione delle performance dell’algoritmo all’utente è richiesto di esprimere un giudizio mediante un punteggio da 0 a 10. Attribuire 0 significa che la classificazione proposta non rispecchia in alcun modo quella attesa o non è, secondo l’utente accettabile, mentre la valutazione 10 indica un’ottima classificazione in linea con le aspettative oppure una classificazione inattesa ma altrettanto convincente. Ai fini

---

della valutazione è necessario considerare che si richiede di impostare un'unica dimensione della finestra per la classificazione, ciò comporta che non sempre tale dimensione può essere ottimale per ogni parte dell'immagine. Questo aspetto va preso in considerazione quando si classificano immagini che presentano 2 classi identiche intervallate da una classe differente come cerchiato in rosso in Fig. 6.16.

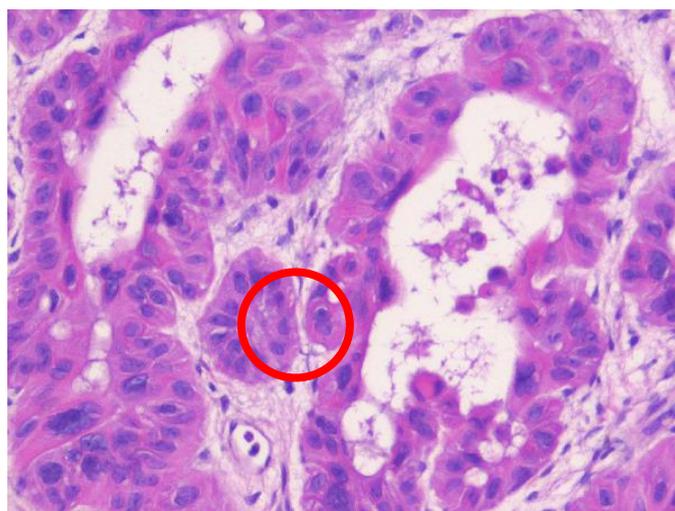


Fig. 6.16 Esempio di classi inseparabili dall'algoritmo

In Fig. 6.17 è presente la classificazione dell'immagine presentata in Fig. 6.16. A sinistra è proposta la suddivisione in 3 classi mentre a destra in 4, si nota che con 3 regioni la classificazione risulta condivisibile mentre con 4 si genera una classe che si può definire 'classe dei bordi'. Quest'ultima si presenta quando si ha una transizione da una classe ad un'altra. In tale contesto le classi, precedentemente identificate, con una classe in meno, risultano più selettive.

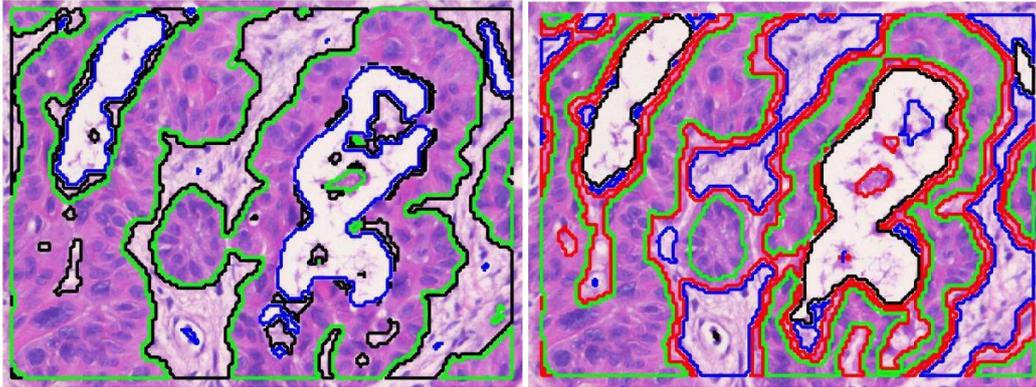


Fig. 6.17 Classificazione con 3 regioni a sinistra e 4 a destra

Tale comportamento è in linea con la logica del classificatore, consentendo di avere una maggiore specificità all'aumentare del numero delle classi a parità di dimensione della finestra. Il classificatore è progettato infatti per suddividere l'immagine in aree e non per una segmentazione di oggetti di piccole dimensioni, tuttavia su immagini ad alta risoluzione questo risulta possibile.

Nel test le immagini presentano una dimensione massima di 600 pixel sul lato maggiore è quindi non consigliato ricercare il dettaglio.

## 6.6 Presentazione delle regioni identificate

Per comprendere in modo veloce e chiaro se le immagini sono correttamente classificate sono state create due *function*. La prima chiamata *mask\_RGB\_toni.m* permette di visualizzare un'unica classe all'interno dell'immagine e tutto il resto in nero, la seconda chiamata *contorno\_bw\_multi\_color.m* genera un contorno di differente colore per ciascuna delle regioni individuate dall'algoritmo fornendo così una visione globale.

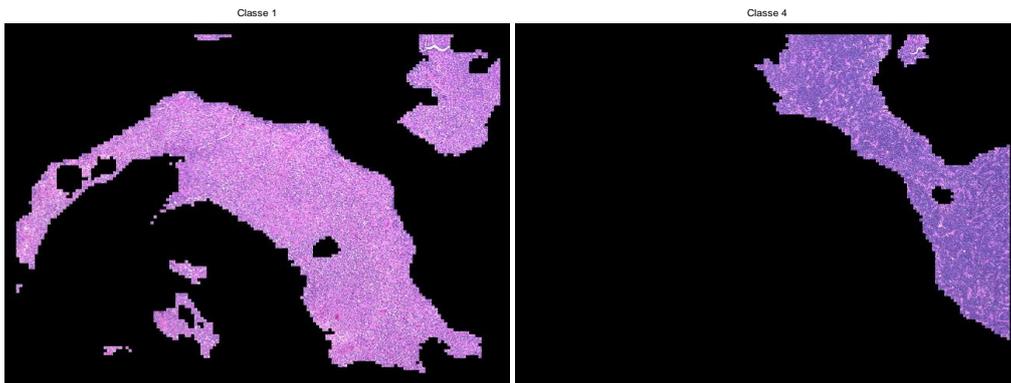


Fig. 6.18 Visualizzazione ottenuta con la *function mask\_RGB\_toni.m*

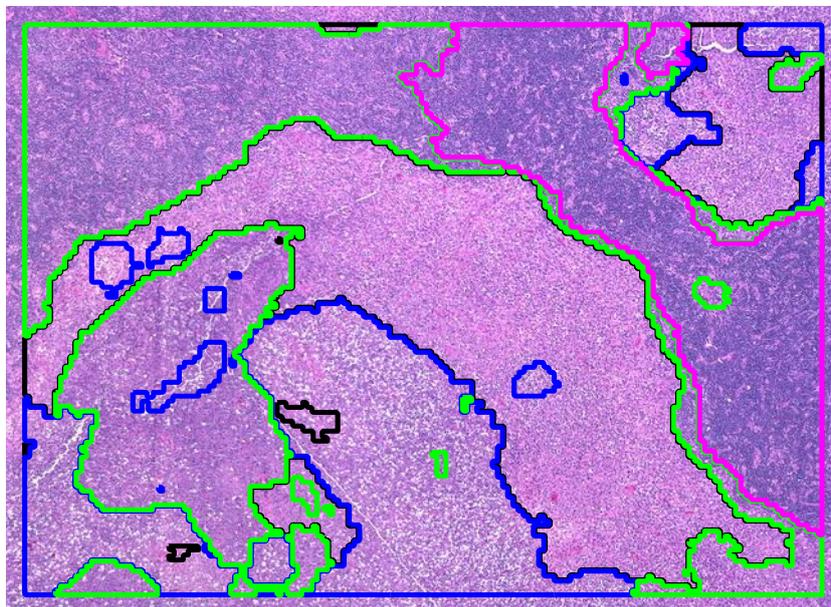


Fig. 6.19 Visualizzazione ottenuta con *contorno\_bw\_multi\_color.m*

## 6.7 Esempio di classificazione

Una delle immagini più significative dal punto di vista della classificazione è quella mostrata in Fig. 6.20. Si tratta di una porzione di un mosaico di un vetrino istologico rappresentativo di un tumore al polmone. Si possono distinguere quattro aree di interesse biologico in cui suddividere l'immagine, tutte identificabili da *pattern* caratteristici che corrispondono a:

- sfondo in bianco;

- 
- componente cellulare, alta densità cellulare;
  - componente stromale, bassa densità cellulare;
  - globuli rossi, componente tendente al rosso;

L'algoritmo identifica queste aree in maniera chiara anche se i *pattern* da riconoscere sono molto differenti gli uni dagli altri. Questa suddivisione è stata fatta senza apportare nessuna manipolazione all'immagine originale; la scelta di questo modo procedere deriva da:

- l'istogramma usato per la classificazione contiene già un filtraggio per il rumore;
- non esiste un metodo generale per la manipolazione applicabile al vasto data set di immagini.

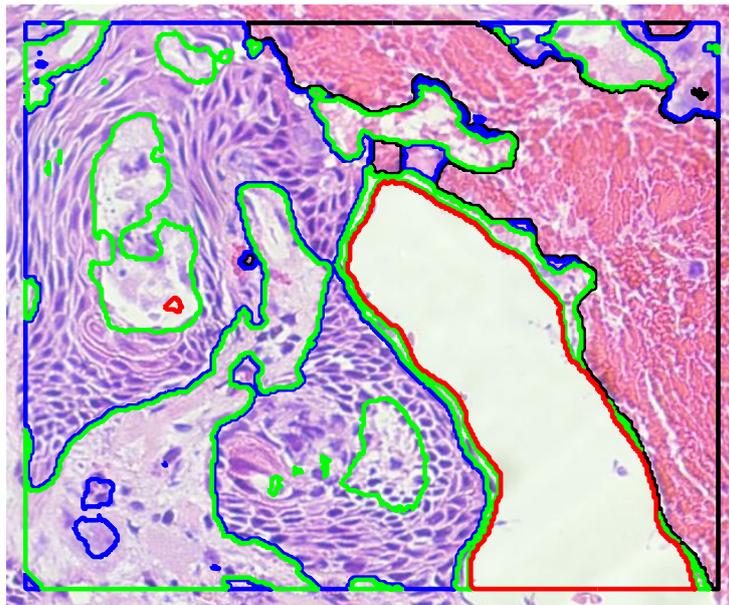


Fig. 6.20 Classificazione polmone

L'immagine in Fig. 6.20 è stata elaborata con una dimensione della finestra tale da consentire l'unione della parte cellulare, di conseguenza all'interno delle macro aree sono state individuate piccole zone associate a classi diverse. Queste, se confrontate con la classe suggerita dal software, si nota una maggiore somiglianza

---

con quest'ultima che con l'area circostante e ciò prova la bontà della discriminazione. Tuttavia, poiché lo scopo del classificatore è identificare macroaree, queste micro aree possono essere interpretate come un errore. L'eliminazione delle microaree rimane un problema aperto in quanto prima bisogna rispondere alla domanda di quali siano le aree maggiormente interessanti per ogni tessuto.

## 6.8 Risultati del test

Le immagini presentate nei test sono state tutte precedentemente ridimensionate, per rendere più veloce la classificazione, in modo che il lato maggiore abbia una risoluzione di 600 *pixel*. L'impiego di un vasto *data set* di immagini da valutare fa emergere un cattivo comportamento del classificatore quando il numero di classi distinte è molto elevato ed quindi necessario usare una dimensione della finestra ridotta.

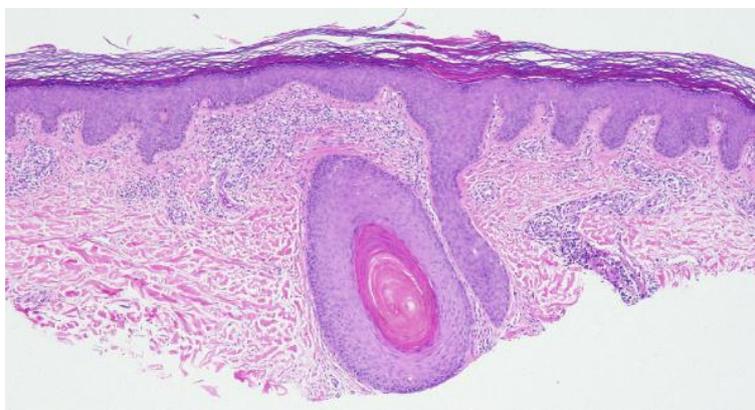


Fig. 6.21 Esempio di immagine che viene classificata male

Ciò è direttamente imputabile alla risoluzione dell'immagine, infatti il classificatore, per lavorare egregiamente, ha bisogno di una finestra che consenta di ottenere un'informazione statistica. Empiricamente è stato dimostrato che, per ottenere una classificazione robusta, la finestra deve avere, almeno, una

---

dimensione pari alla metà del numero di *bin* utilizzati per ottenere l'istogramma. Disporre di immagini ad alta risoluzione permette di lavorare con finestre piccole, relativamente all'immagine trattata, ma contenenti un numero di *pixel* elevato. Tuttavia, l'immagine in Fig. 6.21, anche ad una risoluzione più elevata, non viene classificata bene. Infatti, all'aumentare del numero delle regioni la zona centrale rosa dell'immagine non viene trovata e bisogna impostare il parametro  $K=15$  per riconoscerla correttamente. Questo errore è frutto della distanza utilizzata per confrontare gli istogrammi, la distanza *city-block* genera molte classi spurie soprattutto quando si presenta una transizione tra le regioni, generando le classi che sono state denominate 'classe dei bordi'.

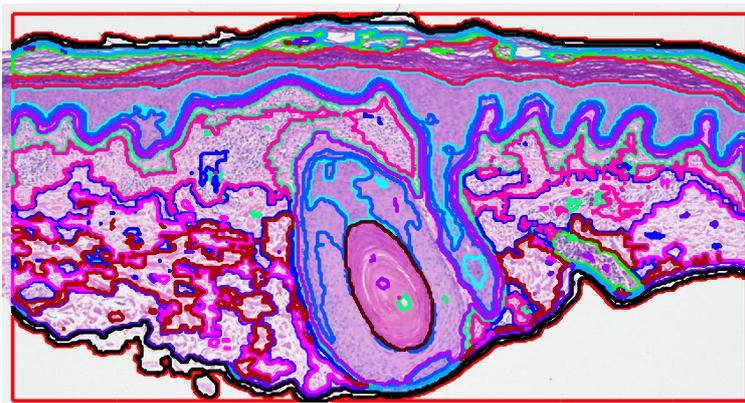


Fig. 6.22 Immagine classificata con  $K=15$

In Fig. 6.22 si può vedere bene il risultato ottenuto con  $K=15$  e notare che si ha una eccessiva frammentazione dell'immagine, tuttavia l'ovale rosa al centro viene riconosciuto correttamente.

Il classificatore si comporta molto bene quando è necessario classificare regioni che si distribuiscono in modo uniforme sull'immagine anche se non connesse fisicamente, ma che presentano lo stesso contenuto informativo. Un'immagine che viene classificata in modo ottimo è presente in Fig. 6.23.

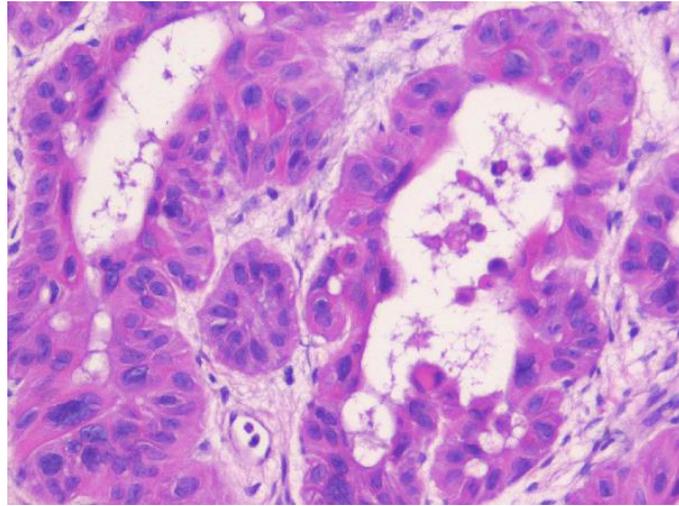


Fig. 6.23 Tipologia di immagine ben classificata

Questa immagine presenta la condizione ideale per l'impiego di questo classificatore, infatti si possono facilmente distinguere due o tre aree che presentano un *pattern* alquanto complesso che richiederebbe un algoritmo *ad hoc*. Il software realizzato, invece, sfrutta l'immagine per generare lo spazio dei dati sul quale si determina la classificazione.

In Fig. 6.24 sono riportati i risultati delle valutazioni effettuate dagli utenti, su una scala da 0 a 10, dove i cerchietti blu rappresentano il punteggio medio di ogni immagine. Si può notare come le valutazioni dipendono fortemente dalle immagini che sono state sottoposte all'utente.

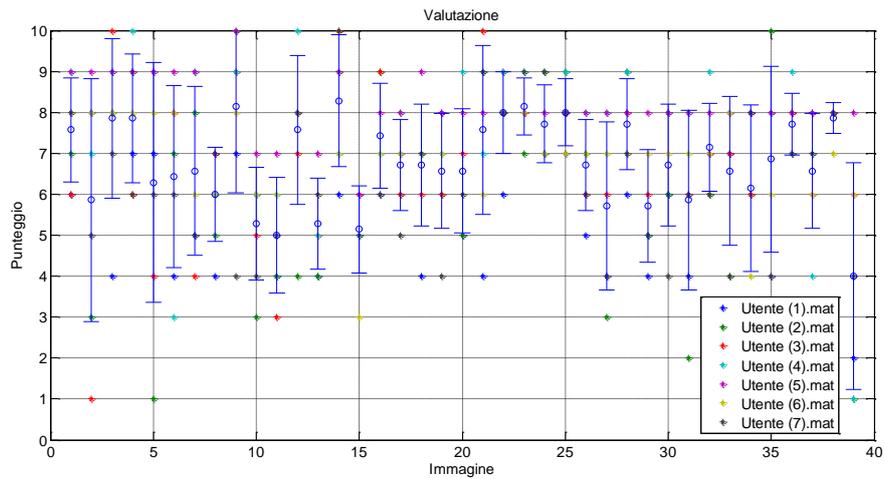


Fig. 6.24 Risultati del test

Il punteggio medio risulta di 6,5. Analizzando il grafico di Fig. 6.24 si nota subito che l'immagine n. 39 non è stata classificata in modo soddisfacente dagli utenti. L'immagine 39 del test è mostrata in Fig. 6.25.

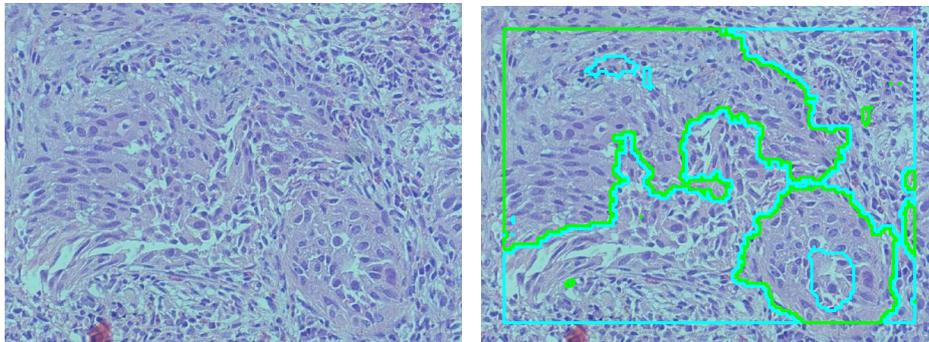


Fig. 6.25 A sinistra l'immagine 39 del test e a destra un esempio di classificazione in 2 regioni

Nell'immagine di Fig. 6.25, per un utente senza un'adeguata formazione medica, risulta molto difficile assegnare sia il numero di cluster che la dimensione della finestra, poiché non sono facilmente percepite due classi distinte.

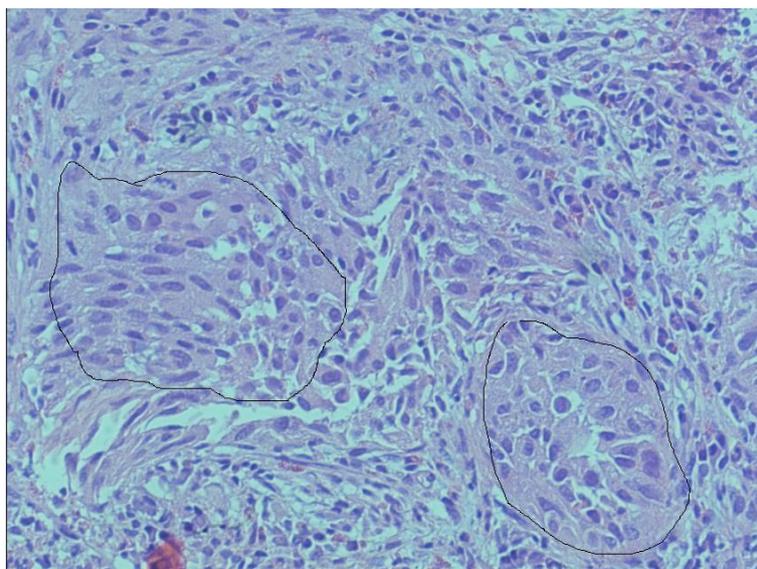


Fig. 6.26 Contornate di nero le aree classificate come tumorali dall'istopatologo

Tale immagine (Fig. 6.26) rappresenta però un vetrino istologico, proveniente da una biopsia di vescica che presenta aree tumorali. Queste aree, individuate da un patologo, sono mostrate in Fig. 6.26. Applicando l'algoritmo di segmentazione si può vedere come vengano individuate due regioni, caratterizzate da pattern differenti, più facilmente distinguibili grazie alla classificazione, che evidenziano proprio le aree tumorali individuate dal patologo.

Escludendo quindi questo dato come *outlier*, il punteggio medio sale a 7,6. La percentuale di punteggi sufficienti, considerando il punteggio medio per immagine è del 74%, mentre il 76,5% di tutti i punteggi degli utenti sono risultati almeno sufficienti.

Occorre inoltre considerare che l'utente ha a disposizione la possibilità di cambiare sia le dimensioni della finestra di calcolo degli istogrammi, sia il numero di classi, anche più volte per la stessa immagine durante il test. Questo produce l'effetto di poter disporre di più valutazioni della stessa immagine per lo stesso utente. Se aggregiamo i dati in tal senso, il valore medio del punteggio massimo dato all'algoritmo di segmentazione per immagine risulta di 8,4.

In Fig. 6.27, si riportano le distribuzioni dei voti medi a sinistra e massimi a destra per immagine.

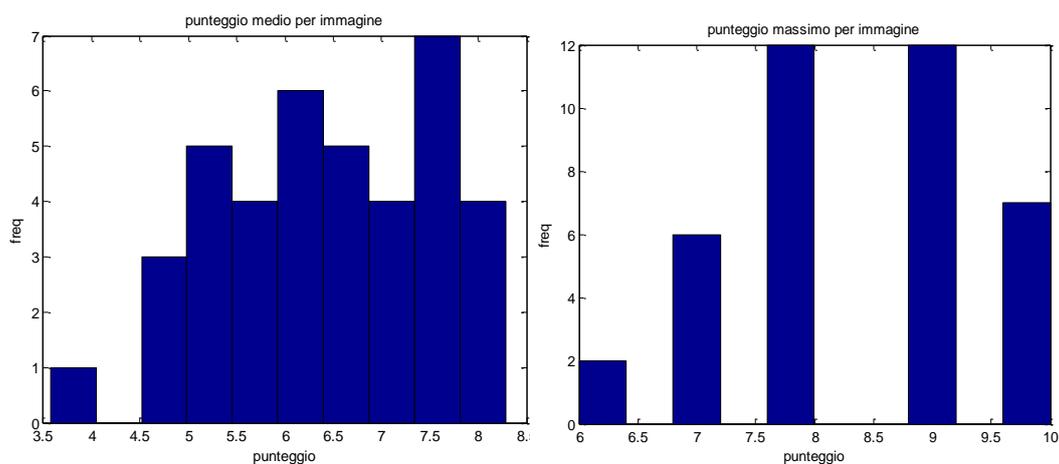


Fig. 6.27 A sinistra (a) è riportata la distribuzione dei punteggi medi assunti dalle immagini e a destra (b) il voto massimo

Per tutte le immagini considerate, esiste almeno un utente che ha dato un punteggio più che sufficiente, con punteggio minimo pari a 6 (Fig. 6.27 (b)). Ciò testimonia come, una volta individuato il set di parametri giusto, l'algoritmo produce una segmentazione in linea con quanto visivamente l'utente percepisce. Per quanto concerne il numero di classi che l'utente attribuisce visivamente alle immagini, in Fig. 6.28, si riporta il valore medio delle classi per immagine.

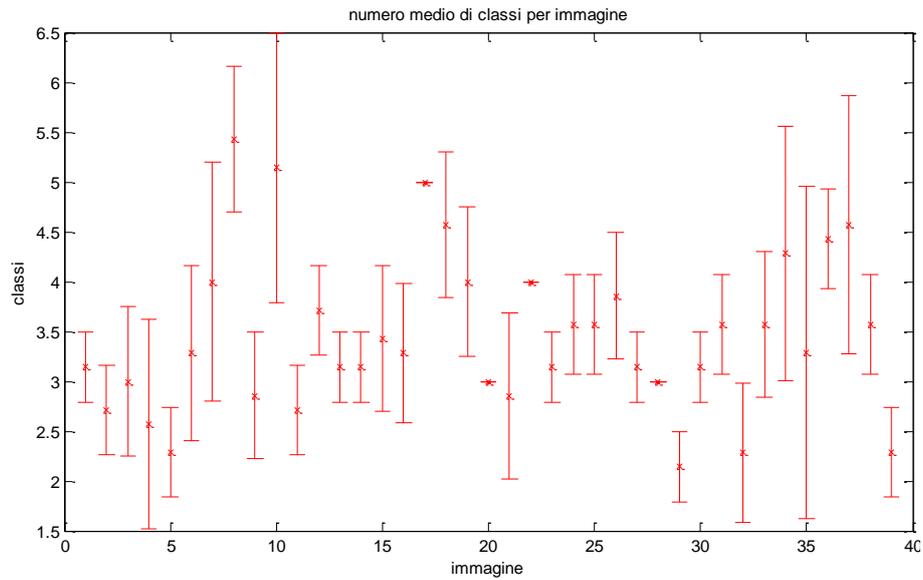


Fig. 6.28 Valor medio e varianza del numero di classi percepito dagli utenti

Come si può notare, per le immagini numero 17, 20, 22 e 28 (Fig. 6.29), vi è completo accordo per tutti gli utenti sul numero delle classi percepite. Negli altri casi, il disaccordo è modesto, ovvero limitato ad al più una sola classe di differenza per il 56% dei casi, e di oltre una classe per il restante 44%.

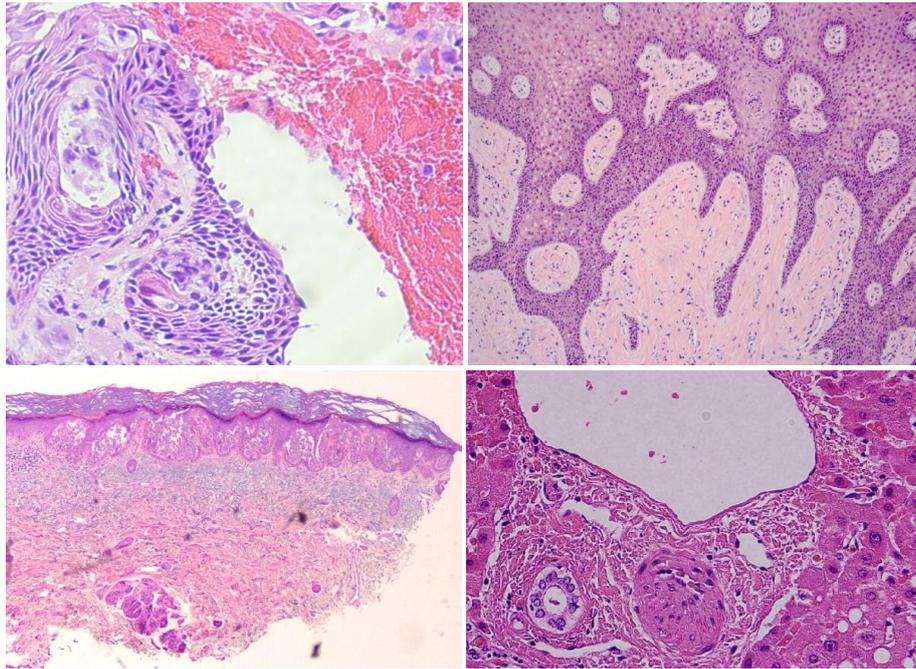


Fig. 6.29 Immagini n. 22 e n. 20 in alto, n. 17 e n. 28 in basso

Dal grafico Fig. 6.30 relativo alla dimensione della finestra utilizzata si può vedere come le immagini che hanno preso un punteggio alto, 9-22-23-25, sono state classificate con una dimensione della finestra molto simile, ciò suggerisce che esiste una tipologia di immagini che può essere trattata con questa classificazione producendo buoni risultati.

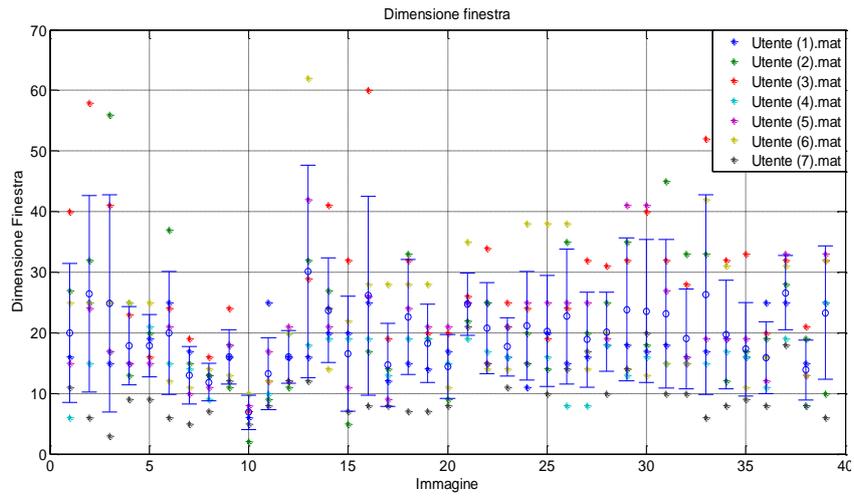


Fig. 6.30 Dimensione della finestra



# Capitolo 7

## Conclusioni

L'algoritmo che è stato realizzato nel presente lavoro di Tesi e descritto in questo elaborato permette di individuare aree percettivamente simili all'interno di un'immagine istologica. Esso trova interessanti applicazioni in ambito ospedaliero rappresentando, in particolare, un valido strumento di supporto agli istopatologi nella classificazione e nella refertazione di campioni istologici. L'algoritmo punta sulla facilità d'uso e sulla velocità della classificazione, requisiti necessari in un contesto di uso pratico. Inoltre, proponendosi di individuare le regioni percettivamente simili, si vogliono fornire le basi per un successivo sviluppo di identificazione semantica delle zone del vetrino istologico indagato. La segmentazione proposta in questa Tesi non porta però ad ottenere sempre ottimi risultati. Infatti, le immagini devono presentare regioni contenenti un *pattern* specifico e che questo sia legato alla *feature* scelta per la classificazione, il colore. L'algoritmo realizzato è stato concepito per individuare regioni e ciò lo rende poco adatto ad indagare il dettaglio. In particolare, questo deriva dall'algoritmo di classificazione utilizzato che, utilizzando gli istogrammi estratti da ogni singola componente colore come *feature*, diventa poco selettivo in caso di finestre molto piccole. Tuttavia, non aver basato la ricerca su specifiche strutture geometriche permette di generalizzare l'impiego dell'algoritmo e di trattare un'ampia tipologia

---

di immagini istologiche. La suddivisione in regioni trova, poi un riscontro biologico: infatti, le classi presentano tratti biologici distintivi, come confermato dal biologo.

Riassumendo, i punti di forza dell'algoritmo sono:

- semplicità d'impiego;
- generalità;
- velocità di esecuzione;
- utilizzo di classificazione non supervisionata.

Vale la pena anche esporre alcune limitazioni del classificatore, il quale non si comporta come atteso, nei casi in cui:

- l'utente scelga una finestra troppo piccola;
- l'utente inserisca un numero troppo elevato di classi;
- le immagini a disposizione siano a bassa risoluzione;
- siano presenti nell'immagine regioni piccole e ben distinte, tali da essere percepite dall'utente come una classe.

Infine, per rendere l'algoritmo più *user friendly*, diversi interventi sull'interfaccia grafica, sia per la scelta dei parametri, sia per la visualizzazione dei risultati possa portare ad un migliore utilizzo.

L'algoritmo è stato pensato e realizzato per ottenere, in futuro, una classificazione del tutto automatica delle immagini: infatti, la necessità di agire solo su un parametro, ovvero la dimensione della finestra, per ottenere una classificazione è sicuramente incoraggiata per un prossimo lavoro di automatizzazione, limitato alla scelta di questo parametro. Inoltre, variare questo parametro permette di indagare l'immagine virtualmente a ingrandimenti differenti, in un modo che ricorda le tecniche multi risoluzione assai più complesse presenti in letteratura. L'approccio di classificazione proposto, basato solo su istogrammi, presenta grandi potenzialità: infatti, è stato possibile comprimere tutta l'informazione presente in una sotto immagine in un unico vettore e solo questa informazione è stata utilizzata per la successiva segmentazione dell'immagine. Questo algoritmo può essere sicuramente migliorato affiancando ad esso un'analisi sui componenti

---

‘biologici’ dell’immagine, al fine di eliminare quelli non rilevanti per una classificazione biologicamente significativa. Inoltre, per alcune immagini, che presentano un numero di regioni elevate, l’algoritmo k-means nella sua versione originale può non essere sufficiente e lo studio di altri metodi di classificazione da affiancare ad esso, in futuro, sarà fondamentale per ottenere un risultato ottimale.



# Bibliografia

- [1] Di Vinay Kumar and Stanley Leonard Robbins, *Robbins Basic Pathology, cap. 6*, 8th ed. 2007.
- [2] M. P. Martano Manuela, Restucci Brunella, Paciello Orlando, "MORFOMETRIA NUCLEARE NELLE EPULIDI ACANTOMATOSE E NEI CARCINOMI SQUAMOSI DELLA GENGIVA NEL CANE."
- [3] S. L. Horowitz and T. Pavlidis, "Picture segmentation by a directed split-and-merge procedure," in *Proceedings of the 2nd Int Joint Conference on Pattern Recognition*, 1974, pp. 424-433.
- [4] K. S. Fu and J. K. Mui, "A survey on image segmentation," *Pattern Recognition*, vol. 13, no. 1, pp. 3-16, 1981.
- [5] B. A. Wandell, *Foundations of Vision*, vol. 21, no. 2. Sinauer Associates, 1995, p. 476.
- [6] M. N. Gurcan, L. Boucheron, A. Can, A. Madabhushi, N. Rajpoot, and B. Yener, "Histopathological Image Analysis: A Review.," *IEEE reviews in biomedical engineering*, vol. 2, pp. 147-171, Jan. 2009.
- [7] W. C. Allsbrook, K. A. Mangold, M. H. Johnson, R. B. Lane, C. G. Lane, and J. I. Epstein, "Interobserver reproducibility of Gleason grading of prostatic carcinoma: general pathologist.," *Human Pathology*, vol. 32, no. 1, pp. 81-88, 2001.
- [8] C. R. King and J. P. Long, "Prostate biopsy grading errors: a sampling problem?," *International journal of cancer Journal international du cancer*, vol. 90, no. 6, pp. 326-330, 2000.
- [9] L. Yang, P. Meer, and D. J. Foran, "Unsupervised segmentation based on robust estimation and color active contour models.," *IEEE transactions on information technology in biomedicine a publication of the IEEE Engineering in Medicine and Biology Society*, vol. 9, no. 3, pp. 475-486, 2005.
- [10] J. M. R. Rubin, D. Strayer, E. Rubin, *Rubin's Pathology: Clinicopathologic Foundations of Medicine*. 2007.

- 
- [11] D. F. Gleason, "Histologic grading of prostate cancer: a perspective.," *Human Pathology*, vol. 23, no. 3, pp. 273-279, 1992.
- [12] G. Alexe et al., "High expression of lymphocyte-associated genes in node-negative HER2+ breast cancers correlates with lower recurrence rates.," *Cancer Research*, vol. 67, no. 22, pp. 10669-10676, 2007.
- [13] V. R. Korde, H. Bartels, J. Barton, and J. Ranger-Moore, "Automatic segmentation of cell nuclei in bladder and skin tissue for karyometric analysis.," *Analytical and quantitative cytology and histology the International Academy of Cytology and American Society of Cytology*, vol. 31, no. 2, pp. 83-89, 2009.
- [14] P. Bamford and B. Lovell, "Method for accurate unsupervised cell nucleus segmentation," *2001 Conference Proceedings of the 23rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, vol. 3, pp. 2704-2708, 2001.
- [15] G. Begelman, E. Gur, E. Rivlin, M. Radzsky, and Z. Zalevsky, "Cell nuclei segmentation using fuzzy logic engine," *2004 International Conference on Image Processing 2004 ICIP 04*, vol. 5, pp. 2937-2940, 2004.
- [16] K. Nguyen, A. K. Jain, and R. L. Allen, "Automated Gland Segmentation and Classification for Gleason Grading of Prostate Tissue Images," *2010 20th International Conference on Pattern Recognition*, vol. 0, pp. 1497-1500, 2010.
- [17] S. Naik, S. Doyle, M. Feldman, J. Tomaszewski, and A. Madabhushi, "Gland Segmentation and Computerized Gleason Grading of Prostate Histology by Integrating Low-, High-level and Domain Specific Information," *Area*, pp. 1-8.
- [18] S. Doyle, A. Madabhushi, M. Feldman, and J. Tomaszewski, "A boosting cascade for automated detection of prostate cancer from digitized histology.," *Medical Image Computing and Computer-Assisted Intervention*, vol. 9, no. 2, pp. 504-511, 2006.
- [19] T. Chen and Y. Chen, "Fast image segmentation based on K-Means clustering with histograms in HSV color space," *Multimedia Signal Processing*, pp. 322-325, 2008.
- [20] Y. Ou and C. Sur, "An Improved Histogram-Based Image Hashing Scheme Using K-means Segmentation," *jwis2009.nsysu.edu.tw*, pp. 1-13.
- [21] Y. Peng, Y. Jiang, and L. Eisengart, "Segmentation of prostatic glands in histology images," : *From Nano to*, vol. 60611, no. c, pp. 2091-2094, 2011.

- 
- [22] K. L. Weind, C. F. Maier, B. K. Rutt, and M. Moussa, "Invasive carcinomas and fibroadenomas of the breast: comparison of microvessel distributions-- implications for imaging modalities.," *Radiology*, vol. 208, no. 2, pp. 477-483, 1998.
- [23] B. Karaçali and A. Tözeren, "Automated detection of regions of interest for tissue microarray experiments: an image texture analysis.," *BMC medical imaging*, vol. 7, p. 2, Jan. 2007.
- [24] L. He, L. Long, and S. Antani, "Distribution fitting-based pixel labeling for histology image segmentation," *Proc. SPIE Medical*, 2011.
- [25] C. Chen, J. Ozolek, and W. Wang, "A pixel classification system for segmenting biomedical images using intensity neighborhoods and dimension reduction," *Biomedical Imaging*., pp. 1649-1652, 2011.
- [26] M. Mignotte, "Segmentation by fusion of histogram-based k-means clusters in different color spaces.," *IEEE transactions on image processing : a publication of the IEEE Signal Processing Society*, vol. 17, no. 5, pp. 780-7, May 2008.
- [27] M. J. Swain and D. H. Ballard, "Color indexing," *International Journal of Computer Vision*, vol. 7, no. 1, pp. 11-32, 1991.