



ELABORAZIONE DELLE IMMAGINI

SEGMENTAZIONE DELLE IMMAGINI: SOGLIATURA

SOGLIATURA

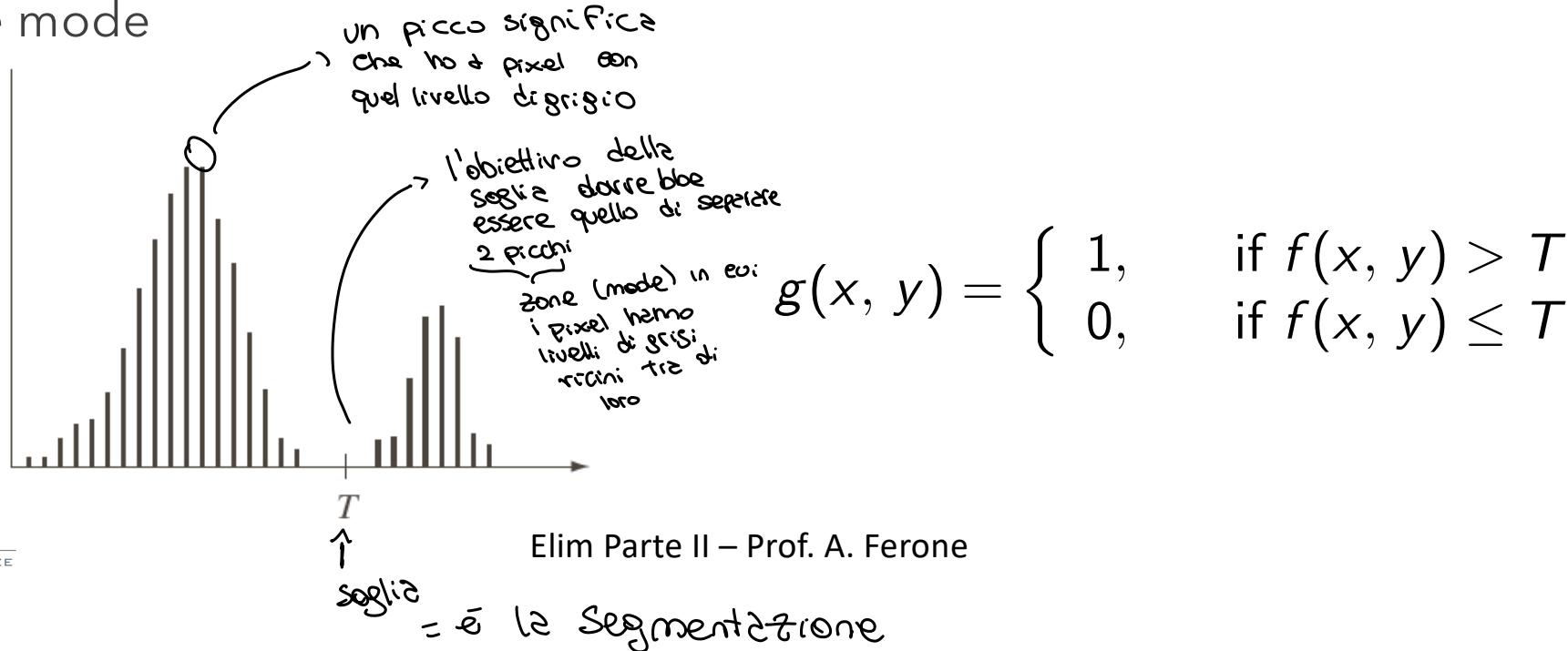
- Lo **sogliatura** è una delle tecniche più **importanti** per la segmentazione delle immagini
 - **Intuitiva**
 - **Semplice** da implementare
 - **Velocità** computazionale
- Le tecniche di sogliatura si basano sull'**analisi** dei valori di **intensità** e su alcune **proprietà** per la partizione delle immagini in regioni
- Tipicamente l'analisi viene effettuata sull'**istogramma dei valori di intensità**

L'idea è stabilire una soglia σ (è un livello di grigio) e i pixel sotto queste soglie appartengono ad un oggetto e i pixel sopra, ad un altro oggetto

Elim Parte II – Prof. A. Ferone

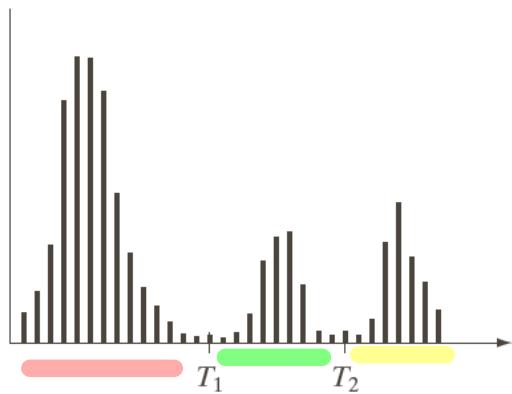
SOGLIATURA

- Si consideri l'**istogramma** di un'immagine $f(x,y)$ composta di **oggetti chiari su sfondo scuro**
- I valori sono raggruppati in **due mode** principali
- Per segmentare l'immagine è possibile scegliere una **soglia T** che separa le due mode



TIPI DI SOGLIATURA

- Sogliatura **globale**: la soglia è applicata a tutta l'immagine
- Sogliatura **varaibile**: la soglia viene modificata durante il processo (esecuzione dell'algoritmo)
- Sogliatura **locale**: la soglia dipende dall'intorno di ogni pixel
- Sogliatura **dinamica**: la soglia dipende dalla posizione del pixel
- Sogliatura **multipla**: si utilizzano più soglie



$$g(x, y) = \begin{cases} a, & \text{if } f(x, y) > T_2 \\ b, & \text{if } T_1 < f(x, y) \leq T_2 \\ c, & \text{if } f(x, y) \leq T_1 \end{cases}$$

Elim Parte II – Prof. A. Ferone

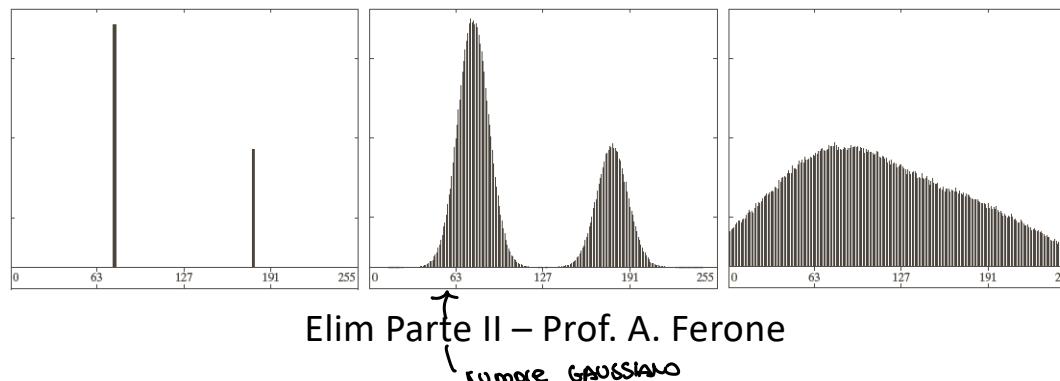
FATTORI CHE INFLUENZANO LA SOGLIATURA

- I **fattori** che influenzano la sogliatura sono
 - Distanza tra i picchi
 - Rumore nell'immagine
 - Dimensione degli oggetti rispetto allo sfondo
 - Uniformità della fonte di illuminazione
 - Uniformità delle proprietà di riflettanza nell'immagine

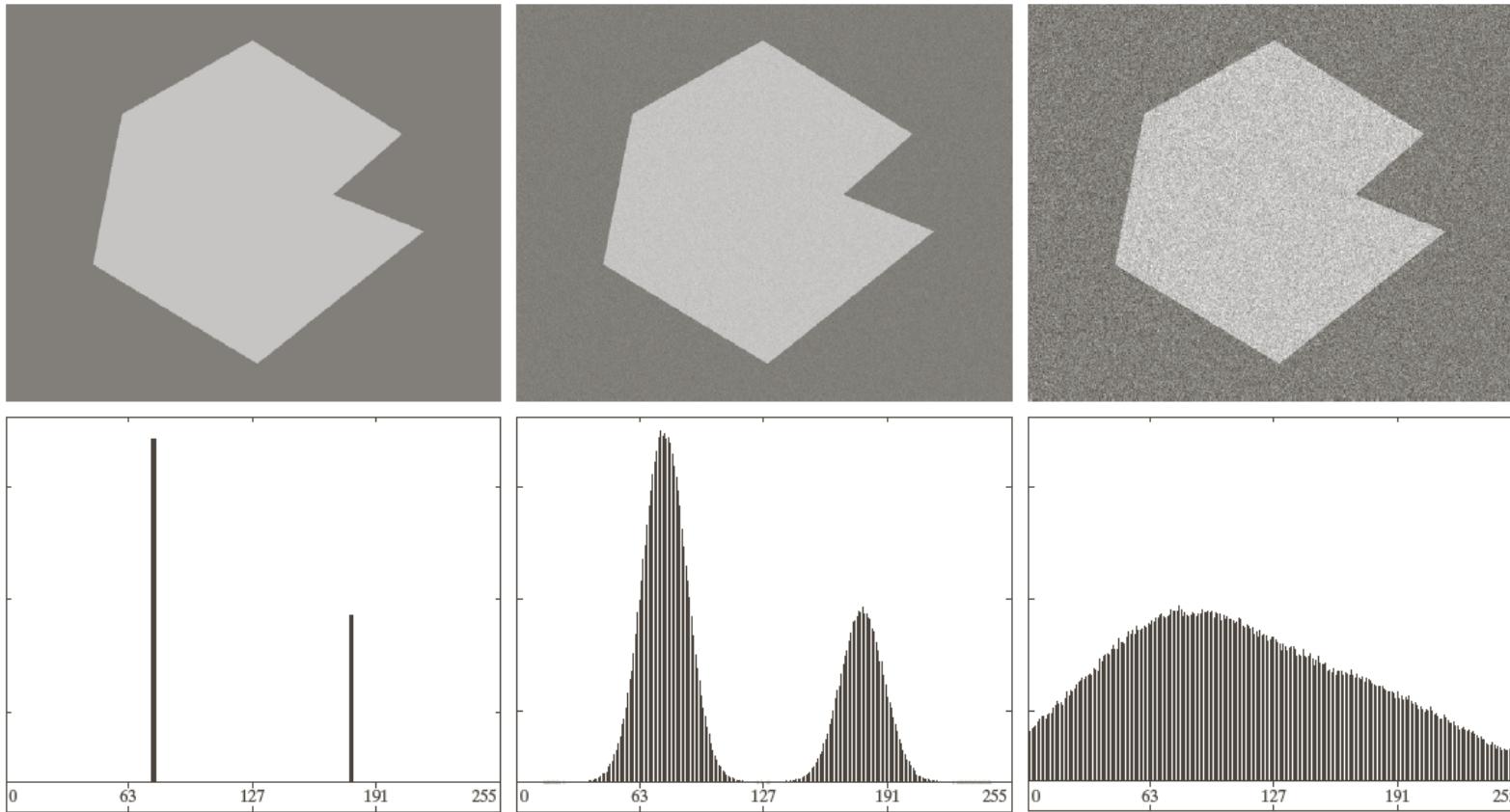
(Se l'oggetto è molto piccolo rispetto allo sfondo i suoi livelli di grigio potrebbero appaixs. da qualche parte non lo vedo)

RUOLO DEL RUMORE

- Si consideri un **istogramma** con **due mode** a "chiodo", **qualsiasi** valore di **soglia** tra le due mode consente di ottenere la **segmentazione**
- Nel caso di un'immagine corrotta da **rumore Gaussiano** con media nulla e deviazione standard pari a 10 livelli di intensità, le **due mode** risultano più **ampie** ma ancora facilmente **separabili**
- Nel caso di un'immagine corrotta da **rumore Gaussiano** con media nulla e deviazione standard pari a 50 livelli di intensità, **non** è possibile **trovare** una **soglia** che consenta di segmentare l'immagine



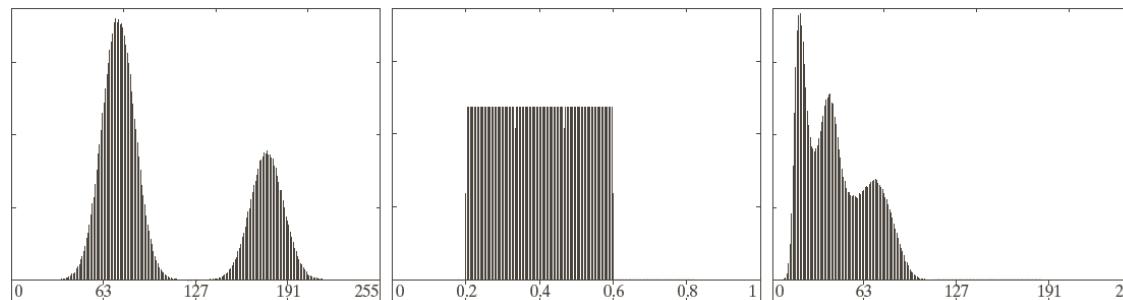
RUOLO DEL RUMORE



Elim Parte II – Prof. A. Ferone

RUOLO DELL'ILLUMINAZIONE

- Per illustrare gli effetti di un'**illuminazione non uniforme**, si **moltiplica l'immagine** da segmentare con una **funzione di intensità** variabile
- Si può notare come la **valle** che separava i due picchi **non** è più **visibile** per cui non è più possibile trovare una soglia per segmentare l'immagine



Elim Parte II – Prof. A. Ferone

queste funzioni
sono applicate
a questa

l'istogramma ha perso le sue mode

SOGLIATURA GLOBALE

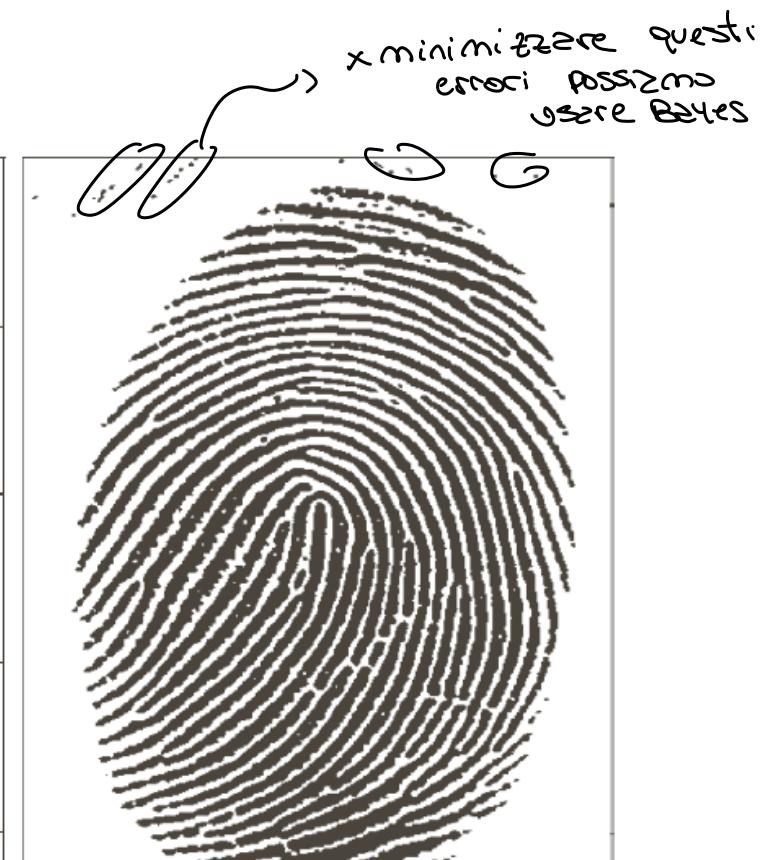
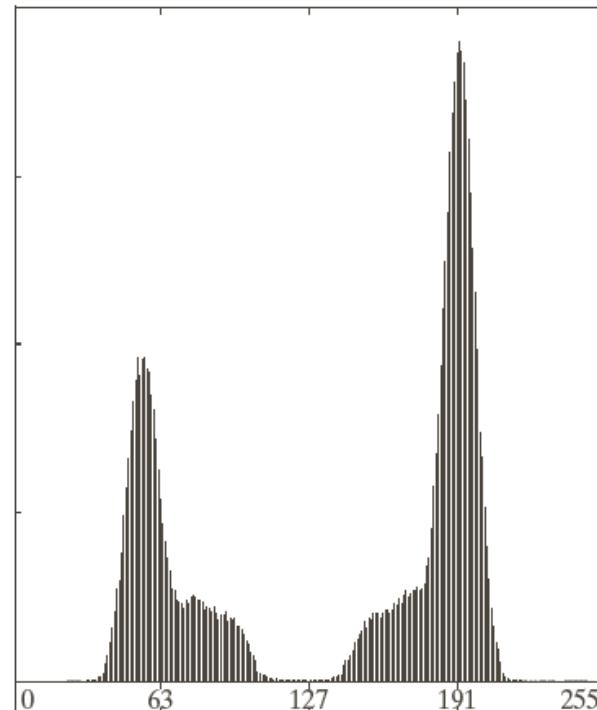
- Quando le **distribuzioni** di **intensità** dei pixel del **background** e quelli di **foreground** sono sufficientemente **distinte** è possibile utilizzare una **soglia singola globale**
- Un semplice algoritmo in 5 passi per trovare il valore della soglia
 1. Stimare il **valore iniziale** di T
 2. **Segmentare** l'immagine utilizzando T in modo da ottenere **due gruppi** di pixel: \mathbf{G}_1 e \mathbf{G}_2
 3. **Calcolare** l'**intensità media** m_1 e m_2 rispettivamente di \mathbf{G}_1 e \mathbf{G}_2
 4. **Ricalcolare** la soglia $T = (m_1 + m_2) / 2$
 5. **Ripetere** i passi **2-4** finchè le soglie in due iterazioni successive non sono minori di un valore: $|T_i - T_{i-1}| < \Delta T$

$$|T_i - T_{i-1}| < \Delta T$$

nuova soglia threshold

Elim Parte II – Prof. A. Ferone

SOGLIATURA GLOBALE



METODO DI OTSU

- La **sogliatura** può essere vista come un **problema** di **statistica** in cui si vuole **minimizzare l'errore medio** dividendo i pixel in **due classi**
- La **soluzione** a questo problema è conosciuta come **regola di decisione di Bayes**
- L'**implementazione** può risultare **complessa e non adatta** ad applicazioni pratiche
- Il metodo di **Otsu** è un'**alternativa ottimale** nel senso che **massimizza** la **varianza interclasse** (separazione tra valori di intensità delle due classi)
- Questo metodo **opera** esclusivamente sull'**istogramma** dell'immagine

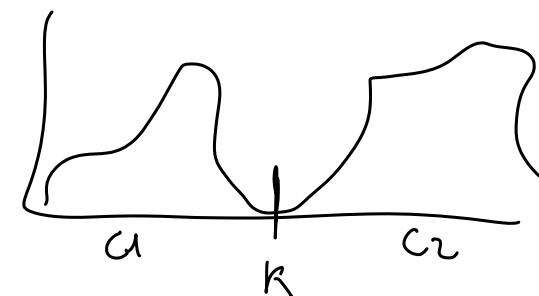
↗ avere gruppi lontani tra di loro, significa che il valore delle loro medie devono essere lontani tra di loro, quindi significa considerare gruppi + compatti (in cui i livelli di grigio sono vicini tra di loro)

Elim Parte II – Prof. A. Ferone

METODO DI OTSU

- Immagine $M \times N$ pixel
- $\{0, 1, 2, \dots, L-1\}$ livelli di intensità
- n_i è il **numero** di **pixel** dell'immagine con **intensità i** ($MN = \sum_{i=0}^{L-1} n_i$)
- L'**istogramma normalizzato** si ottiene **dividendo** ogni n_i per il numero di pixel MN , ovvero $p_i = \frac{n_i}{MN}$, da cui deriva $\sum_{i=0}^{L-1} p_i = 1$ *(è la probabilità che un certo valore d'
intensità si trove nell'immagine)*
- Selezionando una **soglia k** ($0 < k < L - 1$) è possibile segmentare l'immagine ottenendo **due classi C_1 e C_2** contenenti, rispettivamente, i pixel con intensità minore e maggiore della soglia
- La **probabilità** che un **pixel appartenga** alla classe C_1 è $P_1(k) = \sum_{i=0}^k p_i$
- La **probabilità** che un **pixel appartenga** alla classe C_2 è $P_2(k) = \sum_{i=k+1}^{L-1} p_i = 1 - P_1(k)$

NB Pi: può assumere valori
de 0 a 1
evento improbabile
↓ evento certo



METODO DI OTSU

k è la soglia

- Il **valore di intensità medio** dei pixel assegnati alla **classe C_1** è

$i = \Delta$ perché è il primo livello di grana da cui parto (se fosse 0 farebbe 0)

è la probabilità di cadere in quelle classi

$$m_1(k) = \frac{1}{P_1(k)} \sum_{i=1}^k ip_i$$

devi vederlo come il peso che ha quella probabilità

- Il **valore di intensità medio** dei pixel assegnati alla **classe C_2** è

$$m_2(k) = \frac{1}{P_2(k)} \sum_{i=k+1}^{L-1} ip_i$$

- La **media cumulativa** fino al **livello k** e l'**intensità media** dell'intera **immagine** sono

$$m(k) = \sum_{i=0}^k ip_i , m_G = \sum_{i=0}^{L-1} ip_i$$

Elim Parte II – Prof. A. Ferone

METODO DI OTSU

- Per stimare l'**efficienza** della **soglia** al livello k si utilizza il valore

$$\eta = \frac{\sigma_B^2}{\sigma_G^2}$$

σ_B^2 + è alto + ha separato bene le classi
 σ_G^2 quindi massimizzare questo valore
- Ovvero la **varianza interclasse** rispetto alla **varianza globale** (σ_G^2)

$$\sigma_G^2 = \sum_{i=0}^{L-1} (i - m_G)^2 p_i$$

I punti i livelli di grigio sono lontani dalle medie totali

$$\sigma_B^2 = P_1(m_1 - m_G)^2 + P_2(m_2 - m_G)^2$$

$$\sigma_B^2 = P_1 P_2 (m_1 - m_2)^2 = \frac{(m_G P_1 - m)^2}{P_1 (1 - P_1)}$$

essendo globale, non dipende da k , e quindi costante

è questo il valore che voglio massimizzare

METODO DI OTSU

- Al **crescere** della **distanza** delle **due medie**, la varianza interclasse σ_B^2 **aumenta**
- La **varianza interclasse** è una misura della **separabilità** tra le **classi**
- Nella formula $\eta = \frac{\sigma_B^2}{\sigma_G^2}$ il **denominatore** è **costante** (fattore di normalizzazione) per cui anche η è una **misura** della **separabilità** tra le **classi**
- **Massimizzando η si massimizza anche σ_B^2**
- Trovare un k che **massimizzi σ_B^2**

METODO DI OTSU

- **Esplicitando** il valore della soglia k

$$\eta(k) = \frac{\sigma_B^2(k)}{\sigma_G^2}$$

$$\sigma_B^2(k) = \frac{[m_G P_1(k) - m(k)]^2}{P_1(k)[1 - P_1(k)]}$$

- La **soglia ottimale** è il valore k^* che **massimizza** $\sigma_B^2(k)$

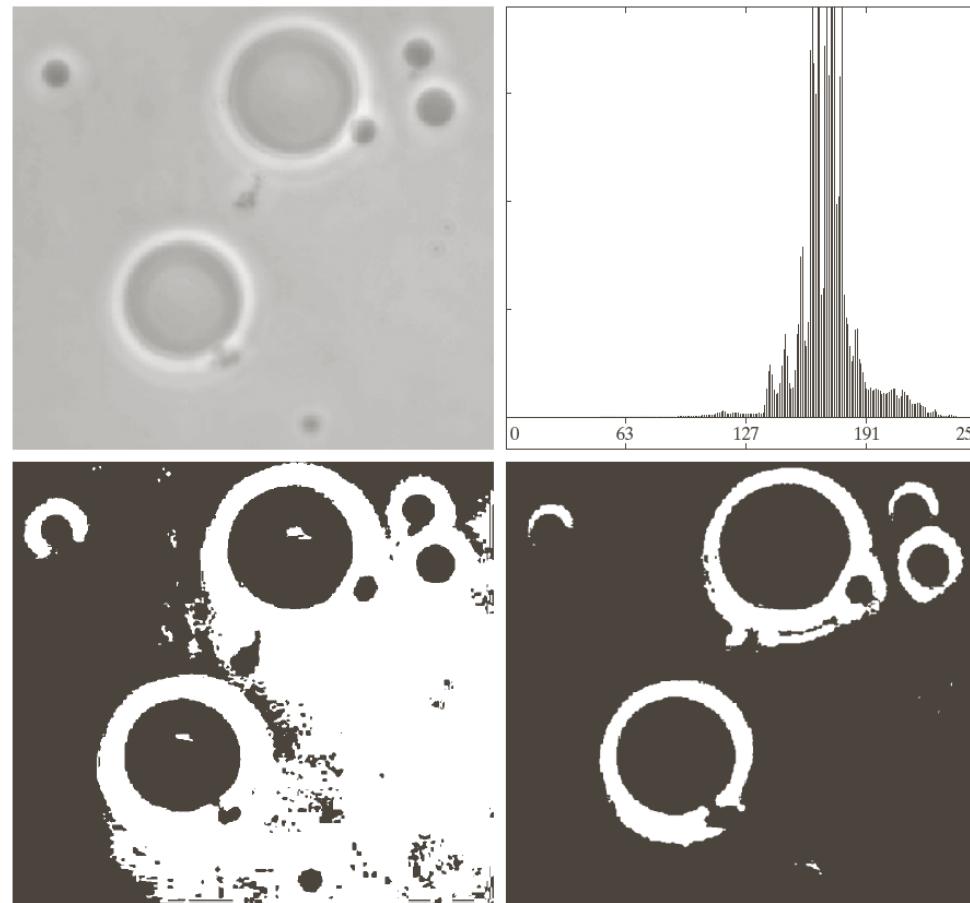
$$\sigma_B^2(k^*) = \max_{0 \leq k \leq L-1} \sigma_B^2(k)$$

- Se $\sigma_B^2(k)$ è **massimo** per **più** di un **valore** si calcola la **media** dei valori di k
- Una volta trovato k^* il valore può essere usato per **segmentare** l'immagine

ALGORITMO DI OTSU

- Calcolare l'istogramma normalizzato dell'immagine
- Calcolare le somme cumulative $P_1(k)$ per $k=0, 1, \dots, L-1$
- Calcolare le medie cumulative $m(k)$ per $k=0, 1, \dots, L-1$
- Calcolare l'intensità globale media m_G
- Calcolare la varianza interclasse $\sigma_B^2(k)$ per $k=0, 1, \dots, L-1$
- Calcolare la soglia k^* , ovvero il valore k per cui $\sigma_B^2(k)$ è massimo
- Calcolare il valore di separabilità $\eta(k^*)$

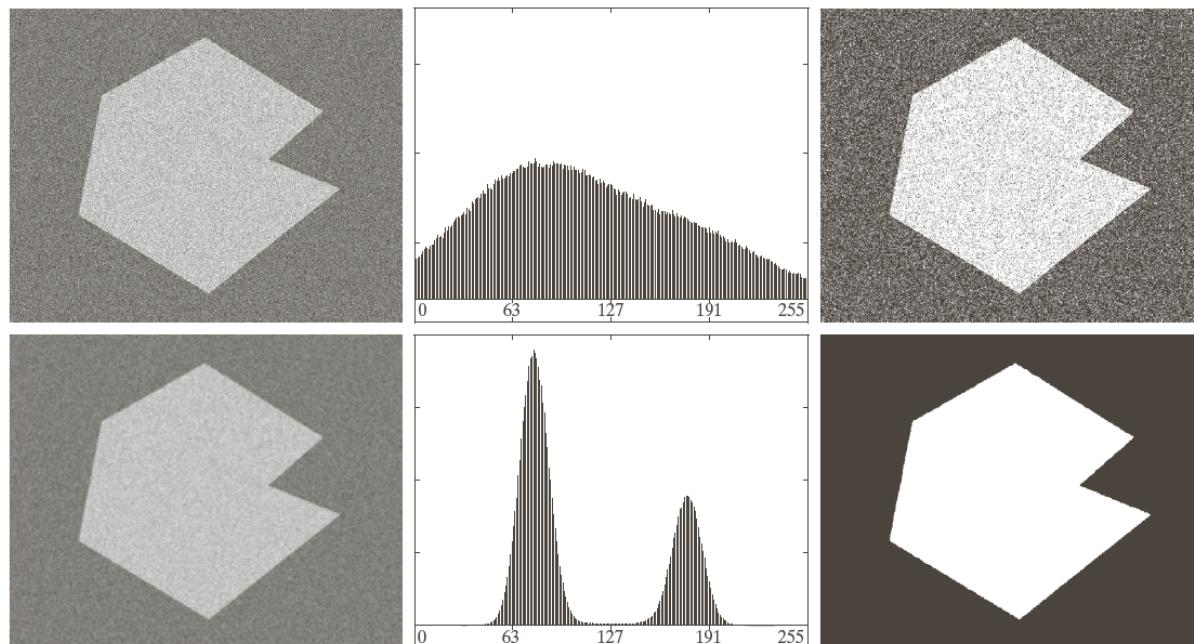
ALGORITMO DI OTSU



Elim Parte II – Prof. A. Ferone

SMOOTHING

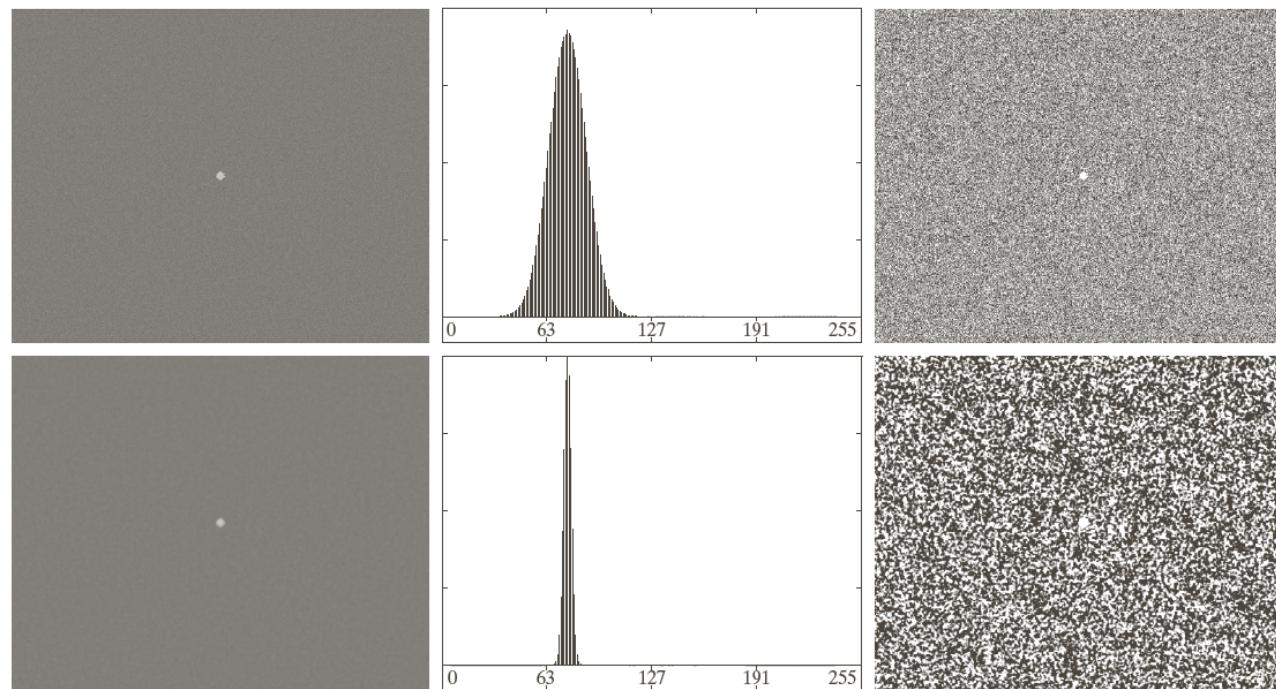
- La presenza di **rumore** può rendere **difficile** il problema della **sogliatura**
- Effettuare lo **smoothing** dell'immagine prima della sogliatura può migliorare le prestazioni



Elim Parte II – Prof. A. Ferone

SMOOTHING

- Nel caso in cui la **regione** da segmentare è così **piccola** che il suo **contributo** all'istogramma è **trascurabile rispetto** a quello del **rumore**, lo **smoothing non risolve** il problema



Elim Parte II – Prof. A. Ferone

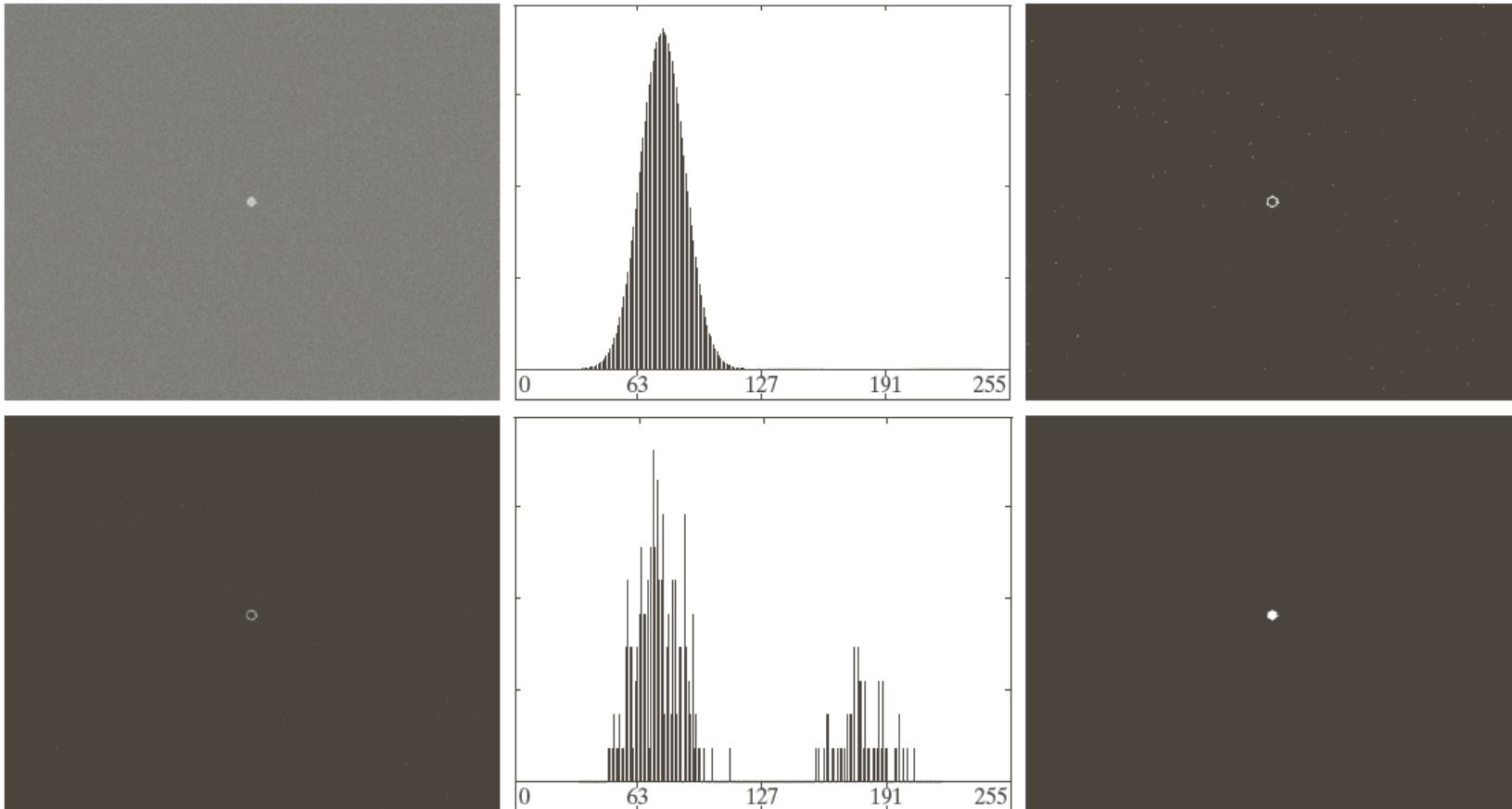
UTILIZZO DEGLI EDGE

- Per **migliorare l'istogramma** dell'immagine al fine della segmentazione è possibile **considerare** solo i **pixel** che si trovano sugli **edge**, o in **prossimità** di un edge, tra gli **oggetti** e il **background**
- Applicando questa idea all'immagine precedente (aree piccole rispetto al background), l'**istogramma** sarà caratterizzato da picchi più o meno della stessa altezza
- La **probabilità** che un **pixel** appartenga al **foreground** è pressochè **uguale** alla **probabilità** che appartenga al **background** per cui le mode saranno simili
- Per **estrarre** i pixel di **edge** è possibile utilizzare il **Laplaciano** o la magnitudo del **gradiente**

UTILIZZO DEGLI EDGE

1. Calcolare un'immagine di edge *con un Laplaciano*
 2. Individuare un valore di soglia $T \rightarrow$ *individuare i pixel di edge e farli:*
 3. Applicare la soglia all'immagine di edge ottenendo un'immagine binaria g_T
 4. Calcolare l'istogramma utilizzando solo i pixel dell'immagine di input che corrispondono alle posizioni dei pixel con valore 1 in g_T
 5. Utilizzare l'istogramma ottenuto per la segmentazione con il metodo di Otsu
- Per la scelta della T iniziale di solito si sceglie un percentile alto (90-esimo)

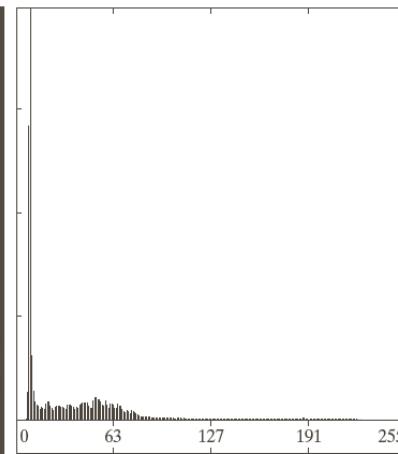
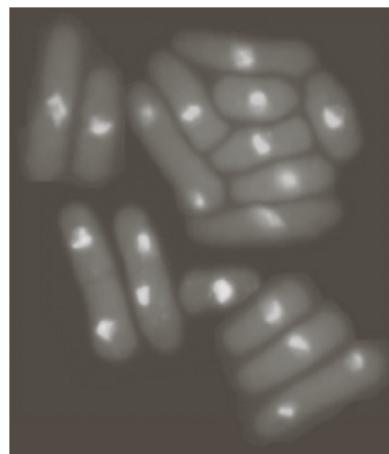
UTILIZZO DEGLI EDGE (GRADIENTE)



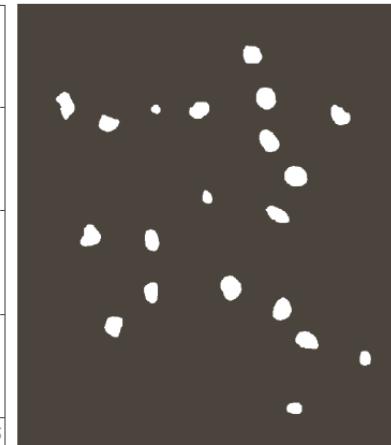
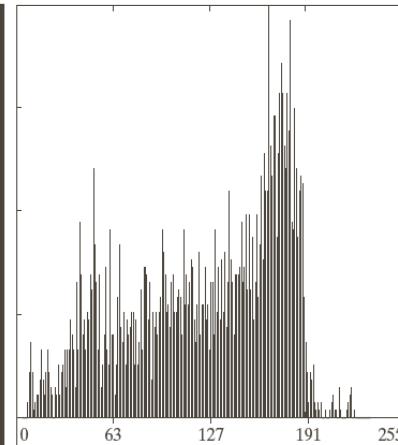
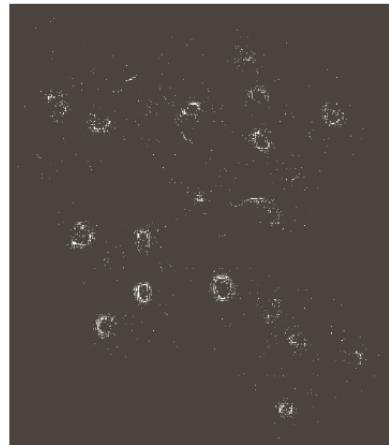
Elim Parte II – Prof. A. Ferone

UTILIZZO DEGLI EDGE (LAPLACIANO)

otsu



LAPLACIANO →



Elim Parte II – Prof. A. Ferone

SOGLIE MULTIPLE

- Il metodo di **Otsu** può essere **generalizzato** al caso di **K classi**
- Tuttavia nel caso dell'istogramma dei valori di intensità con **troppe classi** il metodo **perde di significato**
- Per **K=2** si vogliono separare **tre classi**, ovvero **tre intervalli di intensità**
- La varianza interclasse è data da

$$\sigma_B^2(k_1, k_2) = P_1(m_1 - m_G)^2 + P_2(m_2 - m_G)^2 + P_3(m_3 - m_G)^2$$

- Le soglie ottimali sono

$$\sigma_B^2(k_1^*, k_2^*) = \max_{0 < k_1 < k_2 < L-1} \sigma_B^2(k_1, k_2)$$

- Il grado di separabilità

$$\eta(k_1^*, k_2^*) = \frac{\sigma_B^2(k_1^*, k_2^*)}{\sigma_G^2}$$

Elim Parte II – Prof. A. Ferone

SOGLIE MULTIPLE

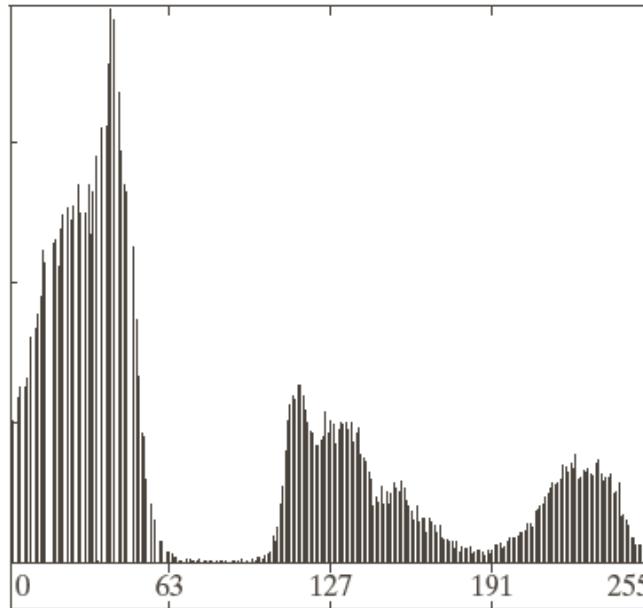
- La procedura inizia selezionando la **prima soglia** $k_1=1$ e poi si **incrementa** la **soglia k_2** a partire dai valori maggiori di k_1 e minori di $L-1$ ($k_2= 2, \dots, L-2$)
- Successivamente si **incrementa k_1** e si **incrementa** la soglia **k_2** a partire dai valori **maggiori** di k_1
- Al termine della procedura si ottiene una **matrice** $\sigma_B^2(k_1, k_2)$ all'interno della quale si trova il valore **massimo** in corrispondenza di k_1^* e k_2^*
- L'immagine **segmentata** si ottiene da

$$g(x, y) = \begin{cases} a & \text{se } f(x, y) \leq k_1^* \\ b & \text{se } k_1^* < f(x, y) \leq k_2^* \\ c & \text{se } f(x, y) > k_2^* \end{cases}$$

→ sono etichette = livelli di grigio

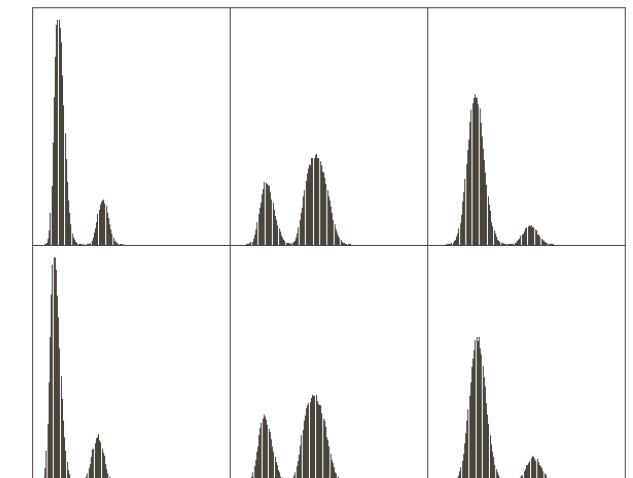
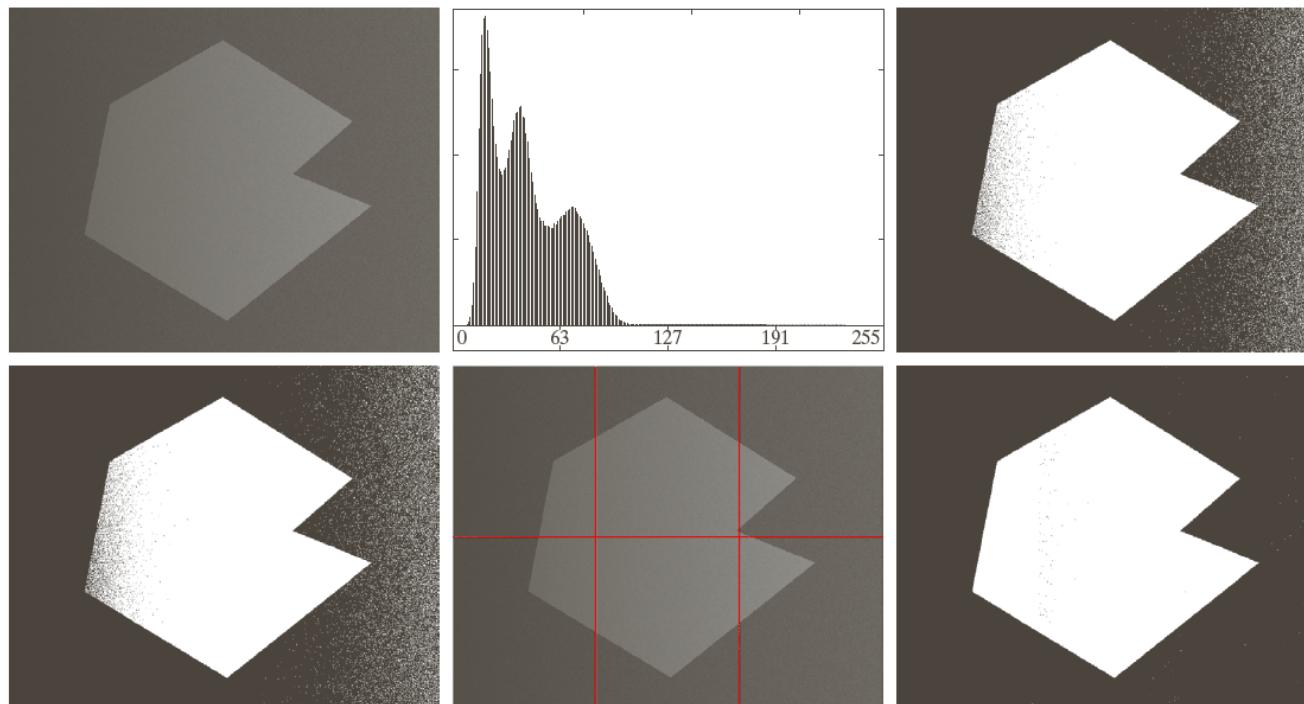
Elim Parte II – Prof. A. Ferone

SOGLIE MULTIPLE



SOGLIATURA VARIABILE

- Uno dei metodi più semplici di **sogliatura variabile** consiste nel suddividere un'immagine in **rettangoli non sovrapposti**



Elim Parte II – Prof. A. Ferone

SOGLIATURA VARIABILE

- Un approccio più generale consiste nel calcolare una **soglia** considerando una o più **proprietà** dell'**intorno** di ogni **pixel**
- Ad esempio si possono considerare la **media** (m_{xy}) e la **deviazione standard** (σ_{xy}) dei valori di **intensità** nell'intorno (S_{xy}) di ogni pixel

$$T_{xy} = a\sigma_{xy} + bm_{xy}$$

a e b sono parametri che pesano le varianze e le medie in un intorno

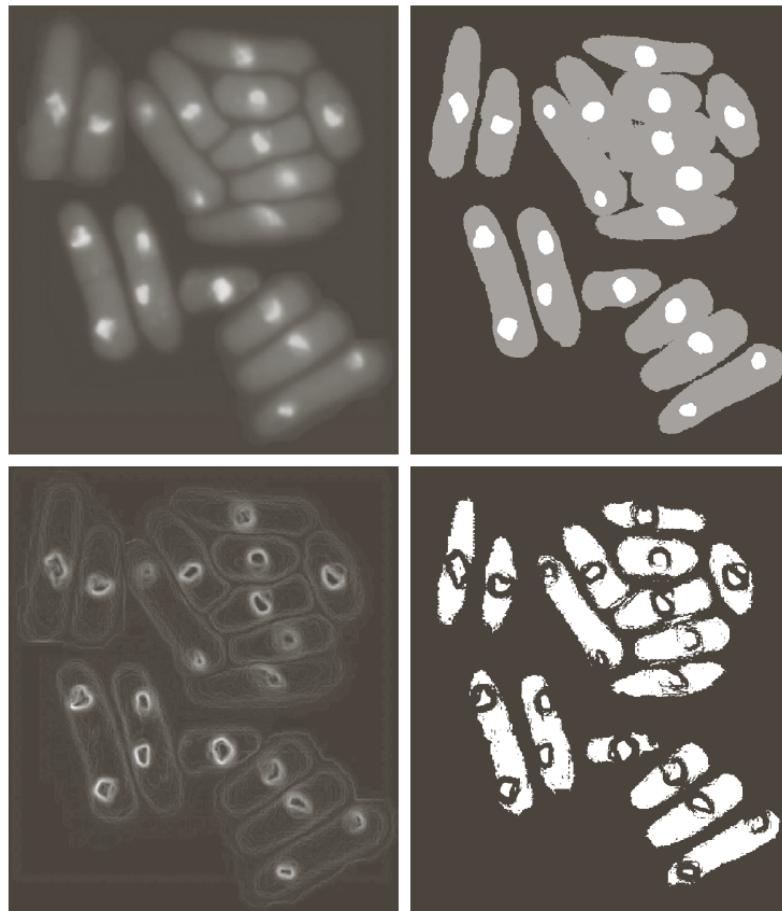
$$T_{xy} = a\sigma_{xy} + bm_G$$

→ poss anche considerare le medie globali x vedere le distanze dei pixel nell'intorno

- In alternativa è possibile usare un **predicato** sui parametri locali

$$Q(\sigma_{xy}, m_{xy}) = \begin{cases} \text{vero(1)} & \text{se } f(x,y) > a\sigma_{xy} \text{ AND } f(x,y) > bm_{xy} \\ \text{falso(0)} & \text{altrimenti} \end{cases}$$

SOGLIATURA VARIABILE



Elim Parte II – Prof. A. Ferone

THRESHOLDING OPENCV

- Per effettuare la sogliatura (thresholding) in OpenCV

```
double cv::threshold(  
    cv::InputArray    src,           // Input image  
    cv::OutputArray   dst,           // Result image  
    double           thresh,         // Threshold value  
    double           maxValue,       // Max value for upward operations  
    int              thresholdType // Threshold type to use  
);
```

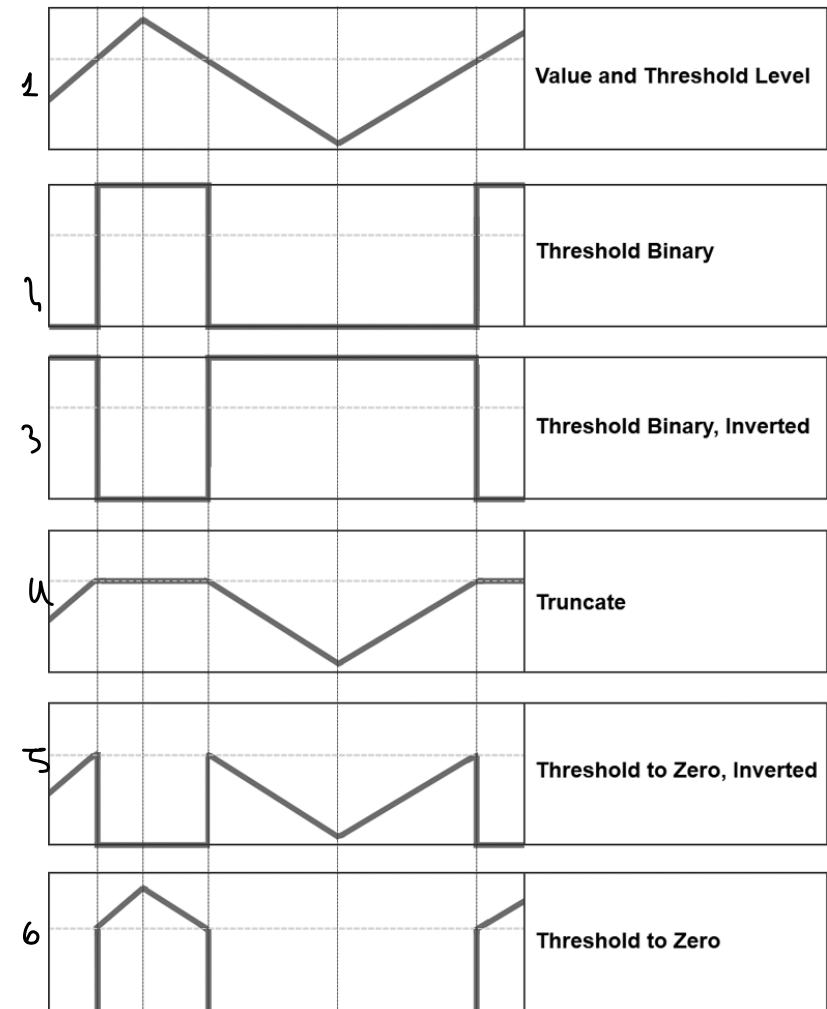
THRESHOLDING OPENCV

Threshold type	Operation
<code>cv::THRESH_BINARY</code>	$DST_I = (SRC_I > thresh) ? MAXVALUE : 0$
<code>cv::THRESH_BINARY_INV</code>	$DST_I = (SRC_I > thresh) ? 0 : MAXVALUE$
<code>cv::THRESH_TRUNC</code>	$DST_I = (SRC_I > thresh) ? THRESH : SRC_I$
<code>cv::THRESH_TOZERO</code>	$DST_I = (SRC_I > thresh) ? SRC_I : 0$
<code>cv::THRESH_TOZERO_INV</code>	$DST_I = (SRC_I > thresh) ? 0 : SRC_I$

`cv::THRESH_OTSU`

Metodo di Otsu

6



Elim Parte II – Prof. A. Ferone

THRESHOLDING OPENCV

- Per effettuare la sogliatura adattiva in OpenCV

```
void cv::adaptiveThreshold(  
    cv::InputArray    src,           // Input image  
    cv::OutputArray   dst,           // Result image  
    double           maxValue,       // Max value for upward operations  
    int              adaptiveMethod, // mean or Gaussian  
    int              thresholdType, // Threshold type to use  
    int              blockSize,      // Block size dell'intorno  
    double           C              // Constant  
)
```

- adaptiveMethod può essere
 - cv::ADAPTIVE_THRESH_MEAN_C (tutti i pixel hanno lo stesso contributo)
 - cv::ADAPTIVE_THRESH_GAUSSIAN_C (come GAUSSIANA)
- blockSize è la dimensione dell'intorno in cui viene calcolata la media pesata
- C è una costante che viene sottratta alla media calcolata

ESERCIZI

- Es. di utilizzo delle funzioni OpenCV a pag. 255-261 del manuale di OpenCV
- Implementare le tecniche di sogliatura descritte nella lezione

