

Imaging in Biologia e Medicina

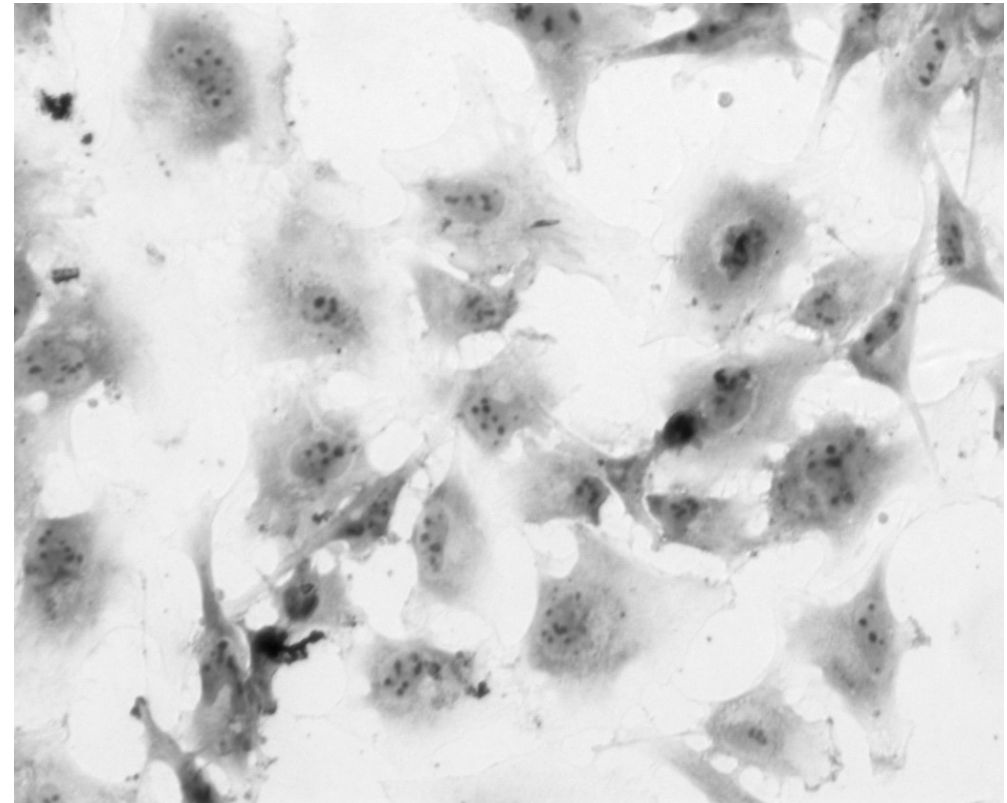
Trasformazioni Morfologiche

Trasformazioni Morfologiche

- Immagini reali sono organizzate secondo “gerarchie” di elementi e forme
 - Forma degli oggetti osservati
 - Strutture e pattern
 - Dettagli
- Di questa gerarchia è talvolta interessante l'essenza del livello più alto

Trasformazioni Morfologiche

- Cultura di cellule in un immagine al microscopio ottico
 - Nuclei
 - Strutture di adesione
 - Impurità del preparato



Trasformazioni Morfologiche

- Le trasformazioni morfologiche sono dedicate alla manipolazione dell'immagine per estrarre da essa i contenuti rilevanti
- La principale applicazione delle trasformazioni morfologiche è sulle immagini binarie
 - Miglioramento dell'identificazione di regioni a partire da operazioni di *thresholding*
 - Rimozione dell'effetto del “rumore”
 - Rimozione di “ponti” e altre strutture spurie
 - Riempimento di “buchi” e imperfezioni
 - Ristabilimento della continuità di strutture che per difficoltà nella loro identificazione appaiono interrotte

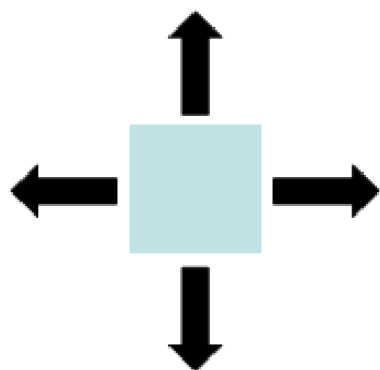
Operazioni Fondamentali

- Molte operazioni, anche complesse, possono essere decomposte in una successione opportuna dell'applicazione delle operazioni fondamentali delle T.M.
 - Erosione: operazione volta a rimuovere dettagli di minore entità (riduce marginalmente la dimensione delle strutture di maggior dimensione)
 - Dilatazione: espansione sui bordi delle regioni per inglobare altre strutture minori (espande leggermente la dimensione delle dimensioni più grandi)

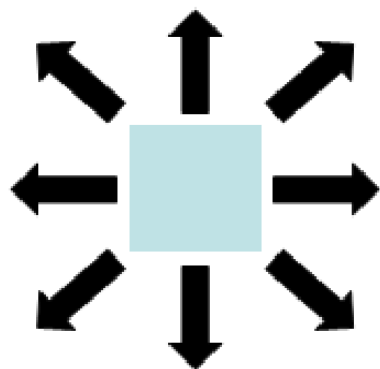
Immagini binarie

- Immagini con pixel aventi valore 0/1
- Convenzionalmente i pixel aventi valore 1 sono il *foreground* mentre i valori 0 definiscono il *background*
- Gli *oggetti* sono definiti in un immagine binaria come ogni regioni di pixel contigui aventi valore 1

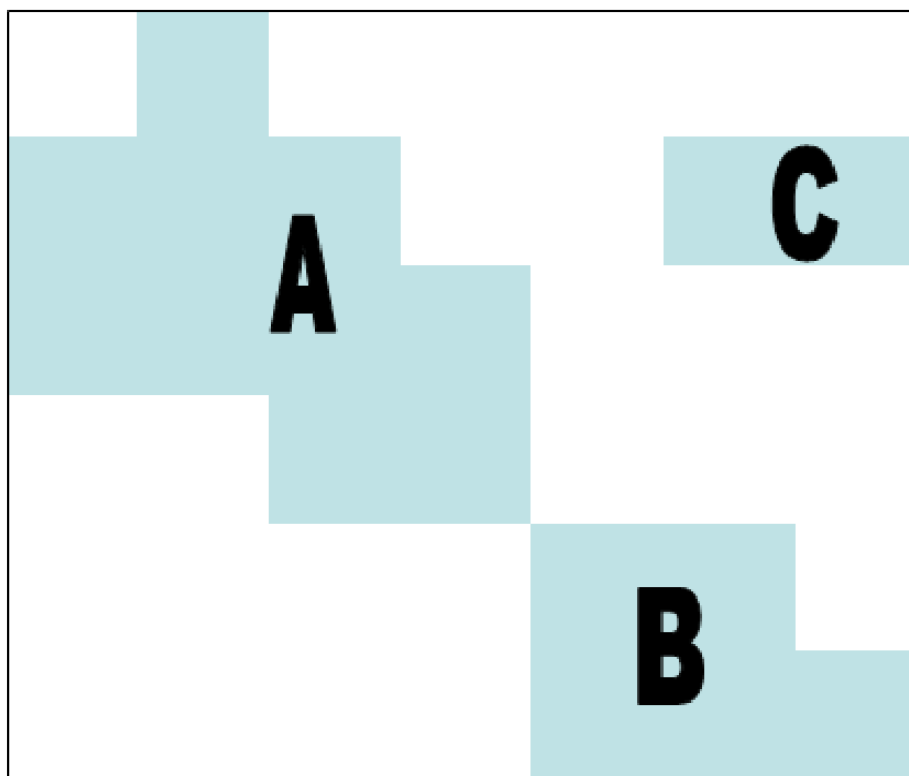
Criteri di Connettività



4-connected



8-connected



4-connected: 3 objects - A, B and C are distinct

8-connected: 2 objects - A and B form one object.

Trasformazioni

- Le immagini binarie non hanno colore o scale di grigio
 - Le uniche trasformazioni possibili sono quelle che determinano transizione $0 \rightarrow 1$ o $1 \rightarrow 0$ di ciascun pixel
 - In altre parole le trasformazioni morfologiche determinano come *foreground* e *background* devono essere cambiati

Elementi della Trasformazione

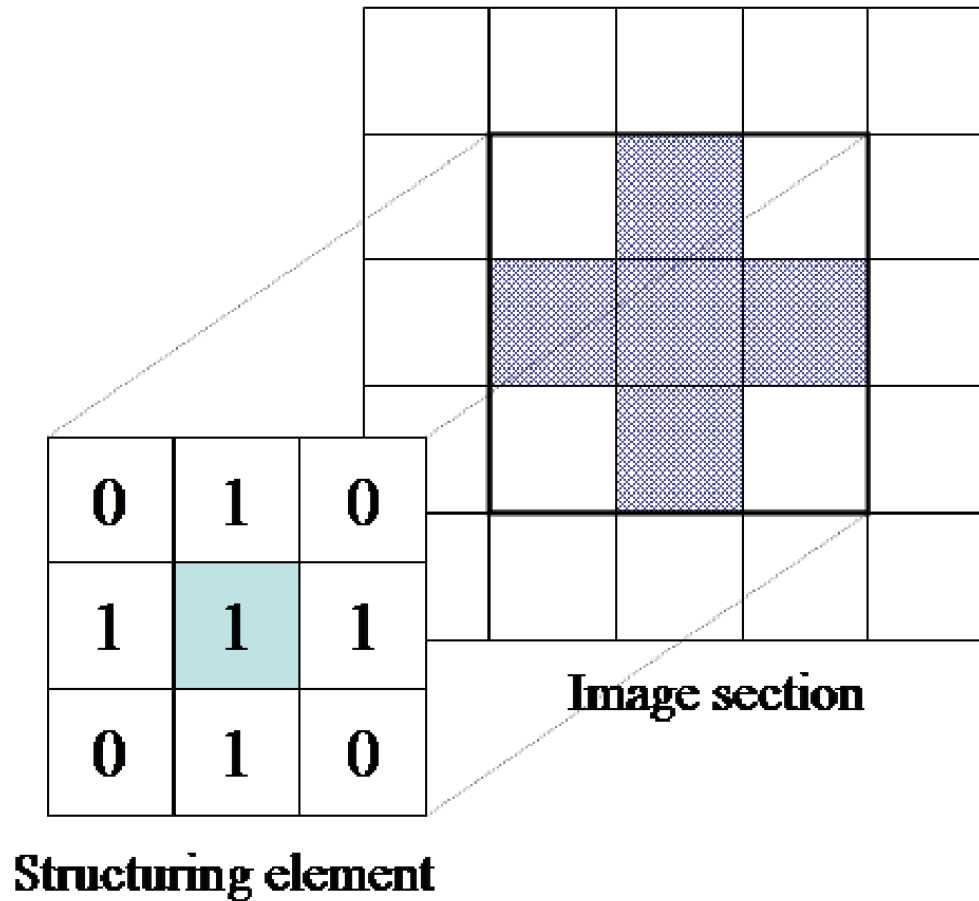
1) Immagine

2) Tipo di trasformazione

3) Elemento strutturante

- Matrice binaria che determina i parametri di calcolo
- Analogo di matrice che implementa un filtro lineare
- Descrive quali pixel rispetto ad uno dato sono da considerare vicini (neighbourhood pixels)
- Possiede un *hot-spot* come per le matrici dei filtri lineari
- L'elemento strutturante viene sovrapposto a ciascun pixel dell'immagine binaria e viene applicato il criterio determinato dal tipo di trasformazione

Elemento Strutturante



Erosione

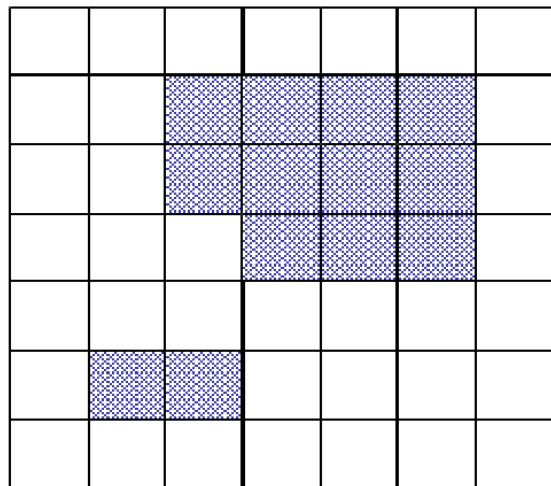
- L'erosione determina quali pixel del foreground dell'immagine binaria devono essere commutati a pixel del *background*
 - L'elemento strutturante viene sovrapposto a ciascun pixel del *foreground*.
 - Se uno qualsiasi dei pixel *vicini* appartiene al *background* allora anche il pixel corrispondente allo *hot-spot* viene posto a 0

Dilatazione

- La dilatazione determina quali pixel del *background* dell'immagine binaria devono essere commutati a pixel del *foreground*
 - L'elemento strutturante viene sovrapposto a ciascun pixel del *background*
 - Se uno qualsiasi dei pixel *vicini* appartiene al *foreground* allora anche il pixel corrispondente allo *hot-spot* viene posto a 1

Erosione/Dilatazione

 = 1  = 0

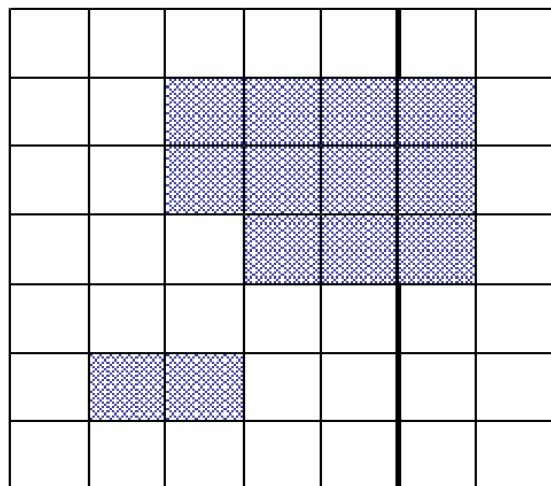
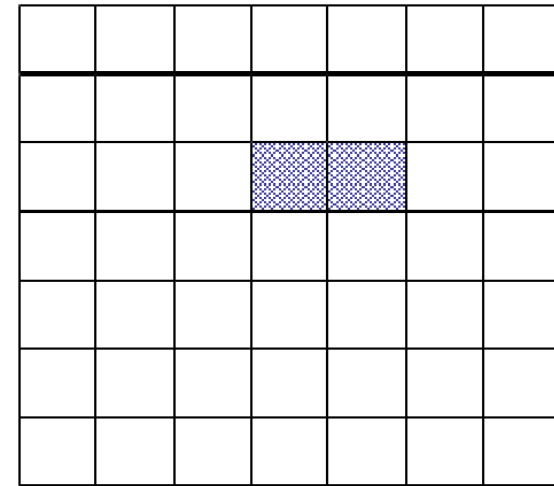


EROSION

0	1	0
1	1	1
0	1	0

⊖

=

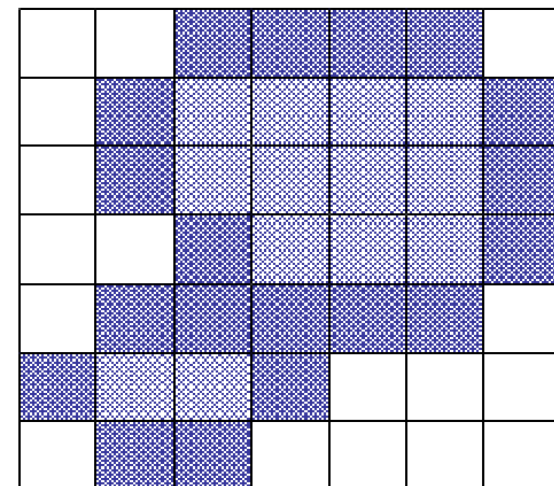


DILATION

0	1	0
1	1	1
0	1	0

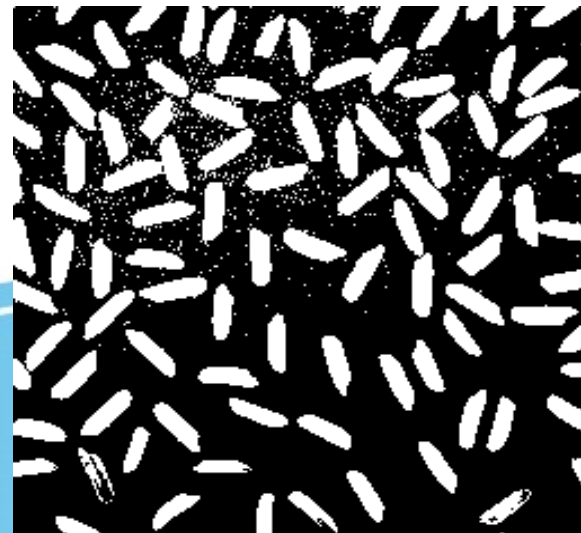
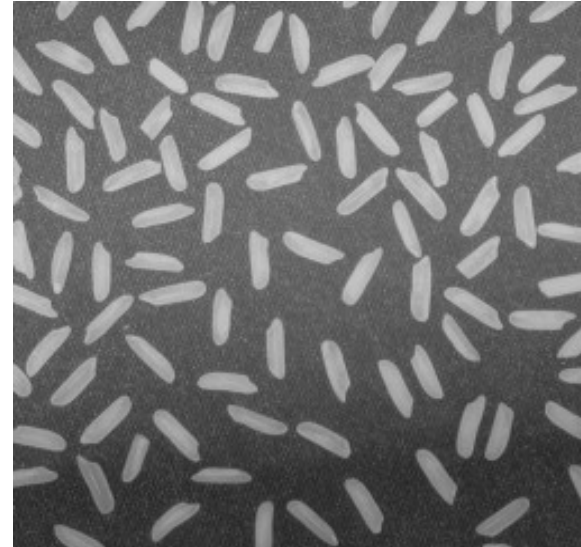
⊕

=



Applicazione dell'erosione

- Rimozione di strutture spurie
- Risultato dell'identificazione degli oggetti usando la soglia calcolata secondo il metodo della *massima entropia*
- Applichiamo all'immagine binaria il metodo dell'erosione



Structuring Element

- Gli *structuring elements* possono essere specificati come matrici
- Octave/Matlab offre la funzione **strel** che produce varie forme di *structuring elements* come “oggetti”, strutture di dati aggregati aventi determinate proprietà di trasformazione
- Tra le proprietà degli *structuring elements* c'è anche la matrice che ne determina la forma e l'azione
- '**help strel**' is your friend

Structuring Elements

```
octave:18> se=strel("disk",2)
```

```
se =
```

```
Flat STREL object with 13 neighbors
```

```
Neighborhood:
```

```
0 0 1 0 0
0 1 1 1 0
1 1 1 1 1
0 1 1 1 0
0 0 1 0 0
```


Erosione+Dilatazione

```
% da immagine RGB a grayscale
```

```
rice=rgb2gray(img);
```

```
% conversione all'intervallo normalizzato [0-1]
```

```
rice=mat2gray(rice);
```

```
imshow(rice)
```

```
% determinazione della soglia con Octave
```

```
thresh=graythresh(rice,"MaxEntropy")
```

```
→ thresh=0.48235
```

```
% immagine binaria
```

```
ricebw=im2bw(rice,thresh);
```

```
figure;imshow(ricebw);
```

```
% erosione e dilatazione con
```

```
% elemento strutturante a disco
```

```
se=strel("disk",2);
```

```
eroded=imerode(ricebw,se);
```

```
dilated=imdilate(eroded,se);
```

```
% display
```

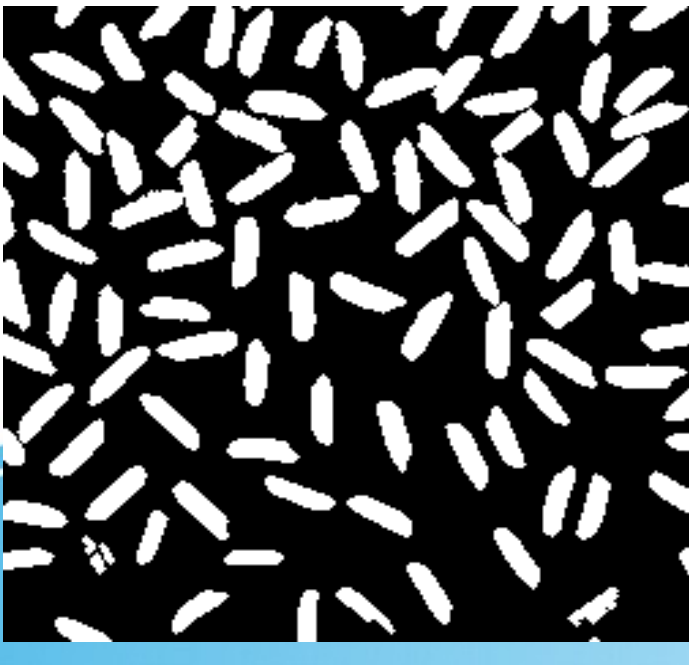
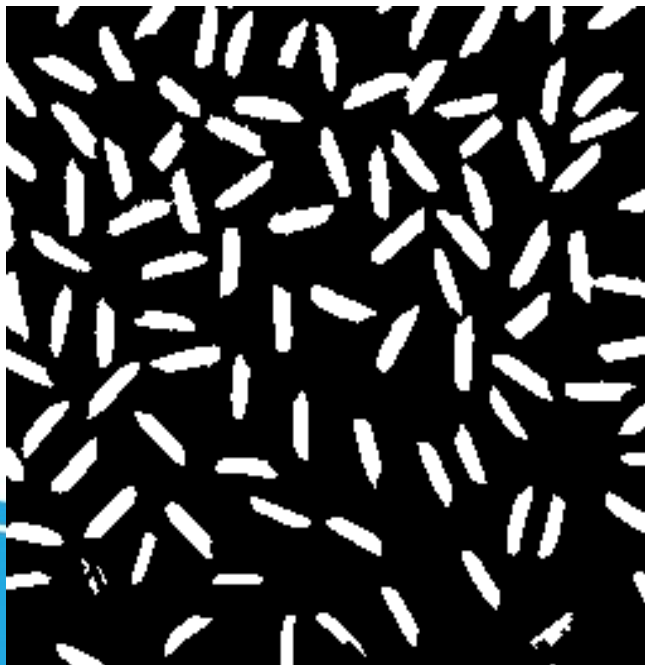
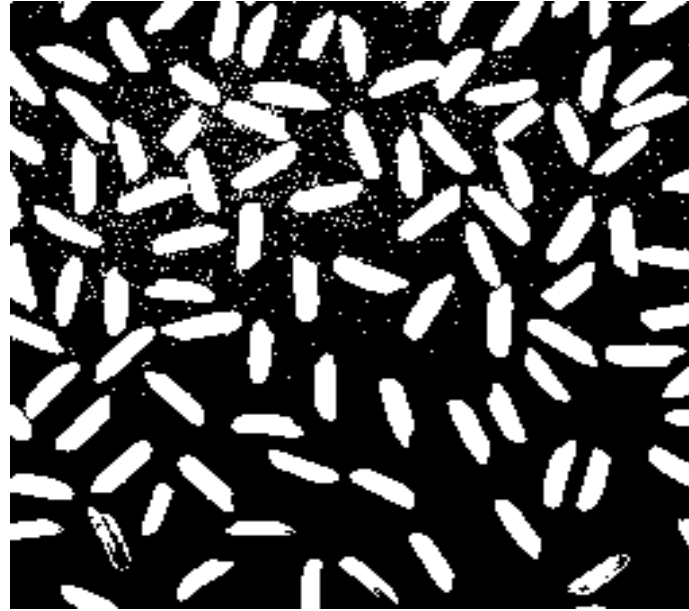
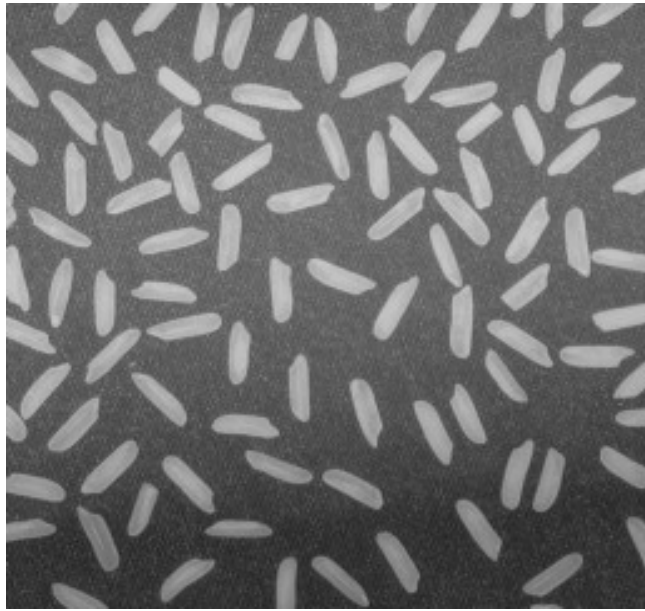
```
subplot(2,2,1);imshow(rice);
```

```
subplot(2,2,2);imshow(ricebw);
```

```
subplot(2,2,3);imshow(eroded);
```

```
subplot(2,2,4);imshow(dilated);
```

Erosione+Dilatazione



Opening

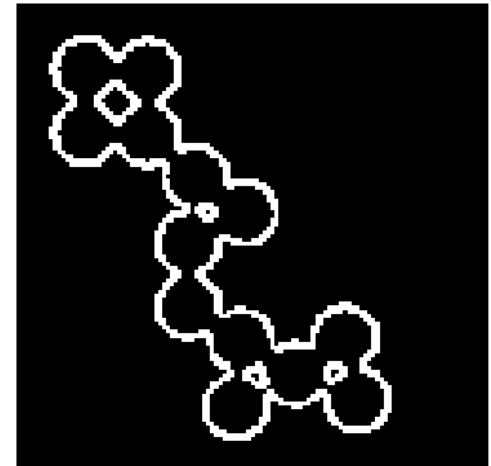
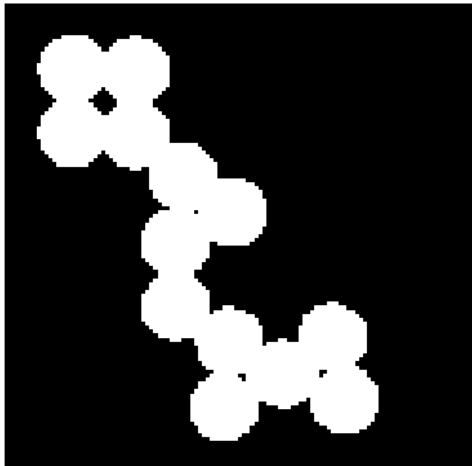
- *Opening* è l'operazione di *erosione* seguita da *dilatazione* fatta con lo stesso elemento strutturante
 - L'effetto è quello di eliminare le strutture più piccole preservando approssimativamente quelle che non vengono eliminate
 - Il ripristino delle strutture più grandi è approssimato. Il grado di approssimazione dipende anche dalla forma e dimensione dell'elemento strutturante scelto

Closing

- Il *Closing* è l'operazione di *dilatazione* seguita da un'operazione di *erosione* fatta con lo stesso elemento strutturante
 - L'operazione di *closing* tende a rimuovere buchi nel foreground più piccoli dell'elemento strutturante usato

Boundary Extraction

- Le operazioni morfologiche possono talvolta essere usate anche per la determinazione di bordi
- La procedura calcola l'immagine binaria A_p ottenuta sottraendo all'immagine originaria A la stessa immagine dopo aver subito un processo di erosione



Erosione/Dilatazione di immagini *grayscale*

- Per le immagini grayscale si può definire analoghe operazioni di erosione e dilatazione, con alcune differenze
 - L'elemento strutturante è definito da
 - Una struttura binaria che ne definisce il dominio di applicazione
 - Valori che attribuiscono pesi diversi ad ogni elemento della struttura binaria

Dilatazione/Erosione per immagini grayscale

- Per elementi strutturanti *piatti* le due operazioni corrispondono ad un filtraggio con fitri di max e min sugli intorni definiti dall'elemento strutturante
- L'*erosione* di un immagine grayscale A tramite l'elemento strutturante B si ottiene assegnando al pixel corrispondente l'hot-spot il valore minimo dell'intensità dei pixel e valori dell'intorno (neighbourhood) definito dall'elemento strutturante.
- La *dilatazione* si ottiene assegnando al pixel corrispondente all'hot-spot il valore massimo dell'intensità dei pixel e valori dell'intorno (neighbourhood) definito dall'elemento strutturante

Correzione background

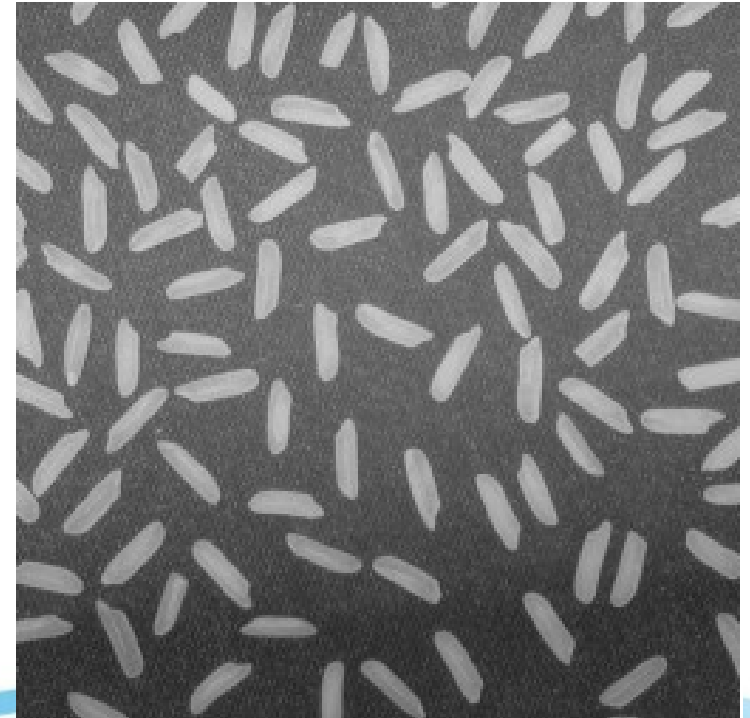
- Metodo alternativo per determinare una soglia corretta per l'immagine grani-riso.png

```
img = imread('grani-riso.png');  
img = mat2gray(rgb2gray(img));
```

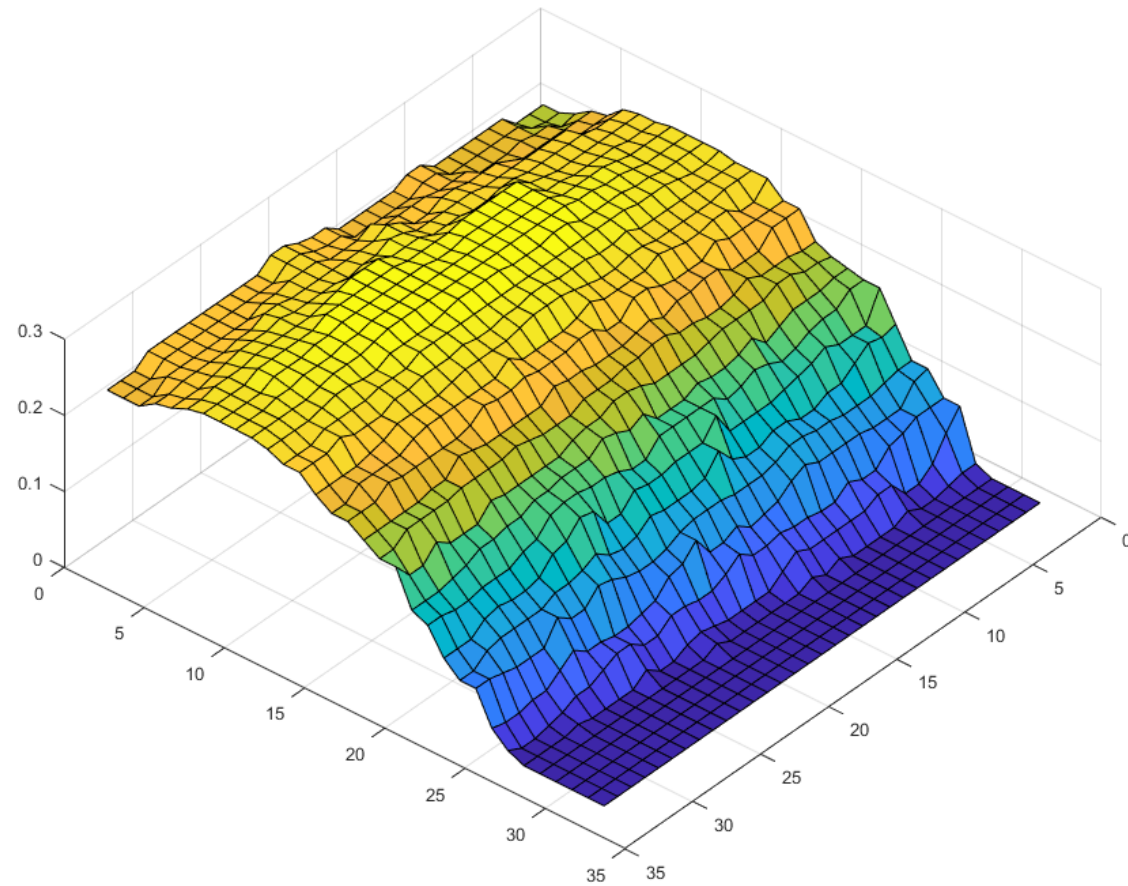
```
figure(1);  
background = imerode(img,strel('disk',15));  
figure(2);  
imshow(background);
```

```
% l'erosione tende ad eliminare i grani  
% più chiari lasciando solo l'andamento  
% approssimato del background nell'immagine
```

```
surf(background(1:8:end,1:8:end));
```



Background dell'immagine



Thresholding

% sottraiamo il background dall'immagine

```
img2 = imsubtract(img,background);  
imshow(img2);
```

% quindi calcoliamo la soglia
% semplicemente con il metodo di Otsu

```
soglia = graythresh(img2);
```

% questa la segmentazione dell'immagine

```
bwimg2 = im2bw(img2,soglia);  
imshow(bwimg2);
```

