

Sessió 8 – Ferran Conde i Sergi Ibàñez

Objectiu: segmentació de cèl·lules

L'objectiu de la sessió és segmentar una imatge que conté cèl·lules. La imatge original és la següent:

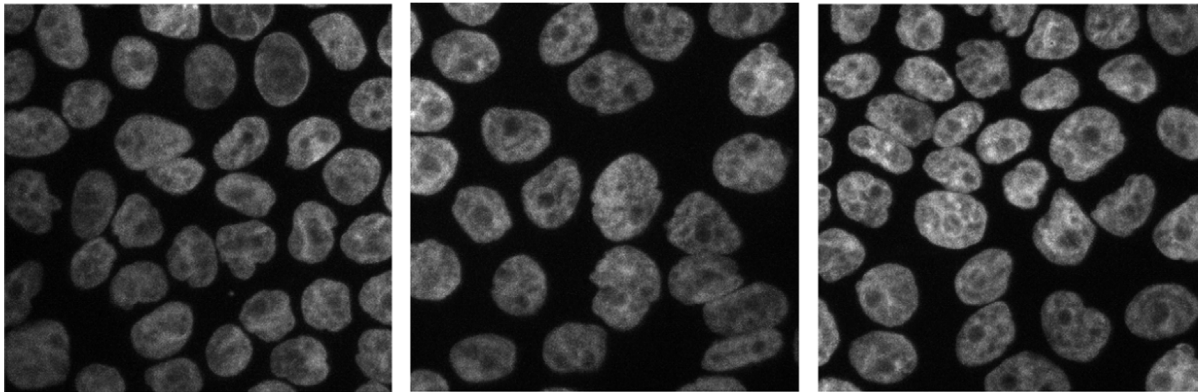


Figura 1. Imatge original

La imatge final s'hauria d'aproximar a la imatge següent:

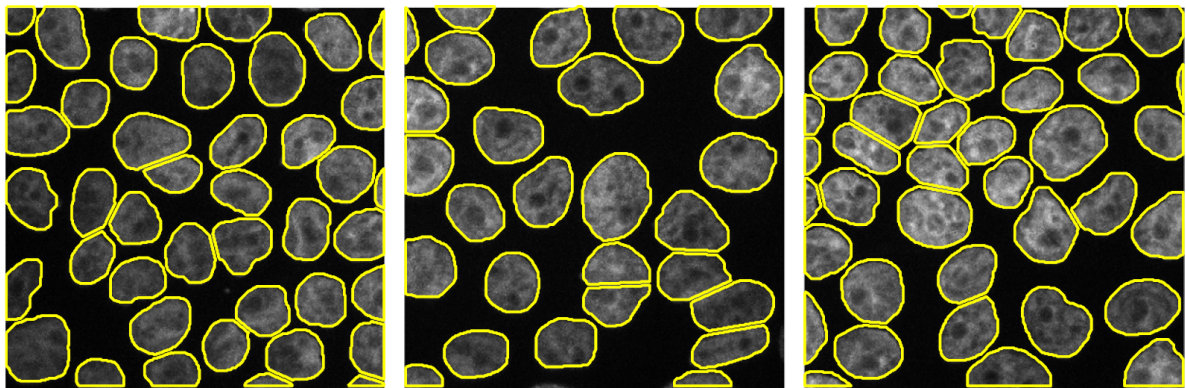


Figura 2. Resultat ideal

Pre-processament de la imatge

La imatge té una vora blanca que actua com un marc. Aquesta vora dificulta el processament; per tant, l'eliminem.

Per fer-ho, hem creat una imatge marc que hem aplicat amb **imreconstruct** sobre la imatge binaritzada per quedar-nos només el marc blanc.

Posteriorment, li hem aplicat dues dilatacions amb sengles elements estructurants en forma de línia (un en horitzontal i l'altre en vertical). Aquesta dilatació la restem a la imatge original per obtenir una nova imatge, sense la vora blanca, sobre la qual treballarem la segmentació.

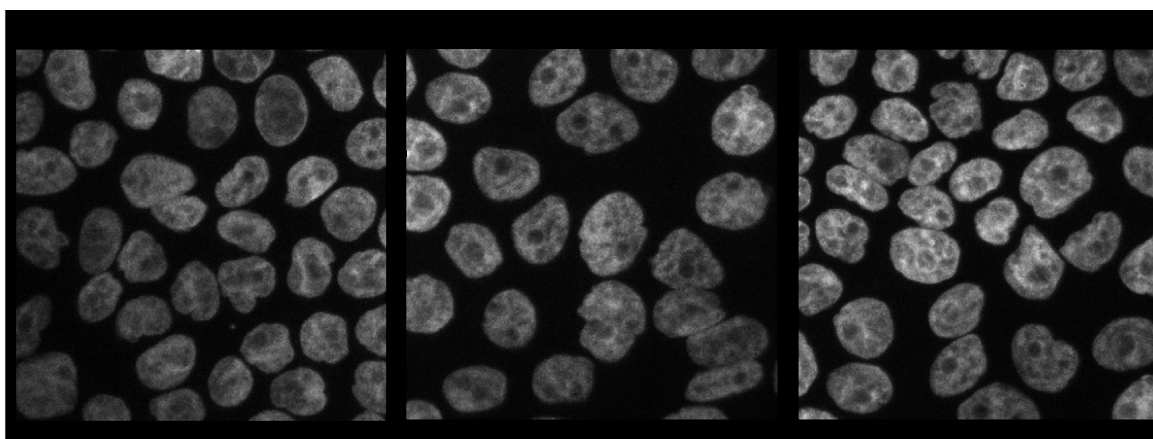


Figura 3: Imatge sobre la qual treballarem

Tractaments pre-binaritzat

Per començar, apliquem filtratge gaussià a la imatge per suavitzar-la. Això es fa per intentar reduir el nombre de màxims locals, atès que necessitarem uns marcadors per efectuar el watershed. Li apliquem tancament i obertura:

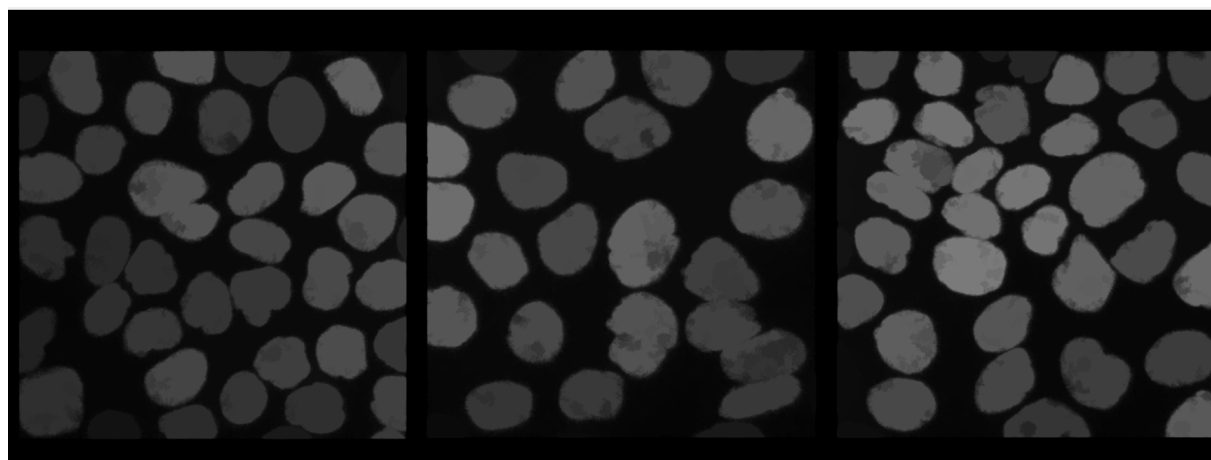


Figura 4: Imatge suavitzada amb tancament i obertura

Binarització i markers

Després de binaritzar, tenim els següents marcadors:

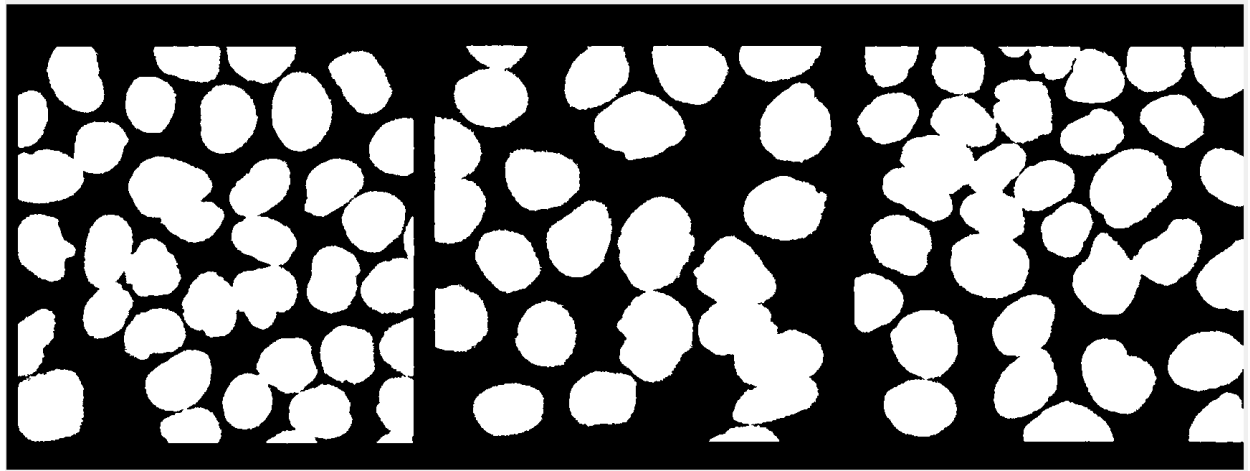


Figura 5: Marcadors

Calculem la transformada de la distància i li apliquem 3 filtres de mediana.
A aquesta transformada li apliquem el **watershed**.

Atès que no agafava bé les cèl·lules de les vores, hem creat un altre marc per agafar el contorn d'aquestes. L'hem aplicat amb **imreconstruct**:



Figura 6: Imatges que tocaven la vora

Finalment, apliquem color groc allà on el watershed és zero.
La imatge resultant és la següent:

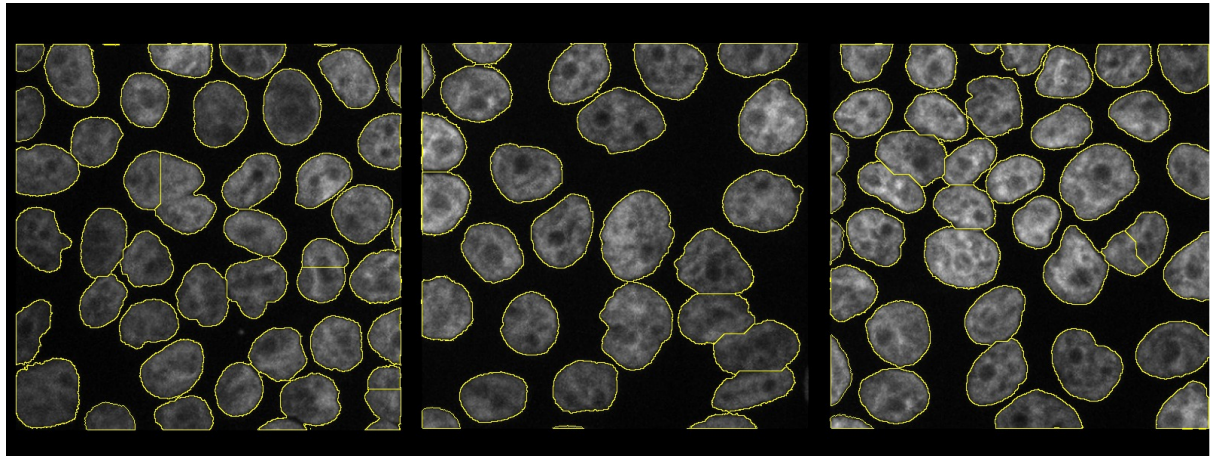


Figura 7: Imatge obtinguda

Annex: codi

```
% Primera part: eliminar el marc
I = imread('cellsegmentationcompetition.png');
I = rgb2gray(I);
figure; imshow(I);

[rows, cols] = size(I);
marc = false(rows, cols);
marc(1, 1:cols) = true;
marc(rows, 1:cols) = true;
marc(1:rows, 1) = true;
marc(1:rows, cols) = true;

blancs = I > 250;
%figure; imshow(blancs);

J = imreconstruct(marc, blancs);
%figure; imshow(J);
SH = strel('line', 6, 0);
SV = strel('line', 8, 90);
J = imdilate(J, SH);
J = imdilate(J, SV);
%figure; imshow(J);

J = not(J);
J = uint8(J);

I = I .* J;
figure; imshow(I);

marc = false(rows, cols);
marc(62, 1:cols) = true;
marc(608, 1:cols) = true;
marc(1:rows, 17) = true;
marc(1:rows, 560) = true;
marc(1:rows, 600) = true;
marc(1:rows, 1189) = true;
blancs = I > 20;
K = imreconstruct(marc, blancs);
[B,L] = bwboundaries(K, 'noholes');
figure; imshow(K);

%% Tractaments pre-binaritzat
se = strel('disk', 5);
filtrada = imgaussfilt(I);
filtrada = imgaussfilt(filtrada);

% Tancament
Iobrd = imdilate(filtrada, strel('disk', 20));
Iobrcbr = imreconstruct(imcomplement(Iobrd), imcomplement(filtrada));
tractada = imcomplement(Iobrcbr);

% Obertura
Ie = imerode(tractada, strel('disk', 20));
Iobr = imreconstruct(Ie, tractada);
Ifinal = Iobr;

% Binaritzat per maxims regionals... no acaba d'anar be
```

```

%markers = imregionalmax(tractada);
%figure; imshow(markers);

%% Binaritzat normal
Ifinal = Ifinal .* 1.6;
markers = Ifinal > 32;
%figure; imshow(markers);

%% Watershed

D = bwdist(not(markers), 'euclidean');
TDF = medfilt2(D);
TDF = medfilt2(TDF);
TDF = medfilt2(TDF);
TDF = -TDF;
TDF(not(markers)) = -Inf;

W = watershed(TDF);
%%

% Obtenim les cèl·lules dels boundaries
for k = 1:length(B)
    boundary = B{k};
    for n = 1:length(boundary)
        W(boundary(n,1), boundary(n,2)) = 0;
    end
end

% Tornem a RGB
capaRG = I;
capaB = I;
capaRG(W == 0) = 255;
capaB(W == 0) = 0;
B = cat(3, capaRG, capaRG, capaB);
figure; imshow(B);

```