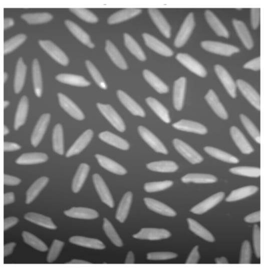
Visión por Computador - Sessió 9

Ferran Velasco Olivera

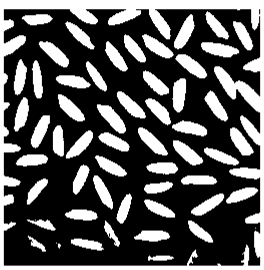
Joaquín Gómez Sánchez

Processat + Segmentació + Etiquetat + Descripció de regions

```
orig = imread('arros.tif');
imshow(orig), title('imatge original')
```

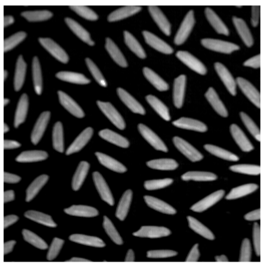


```
% segmentem per binarització
bw=im2bw(orig, graythresh(orig)); % Binarització per Otsu
figure, imshow(bw),title('Threshold automatic')
```

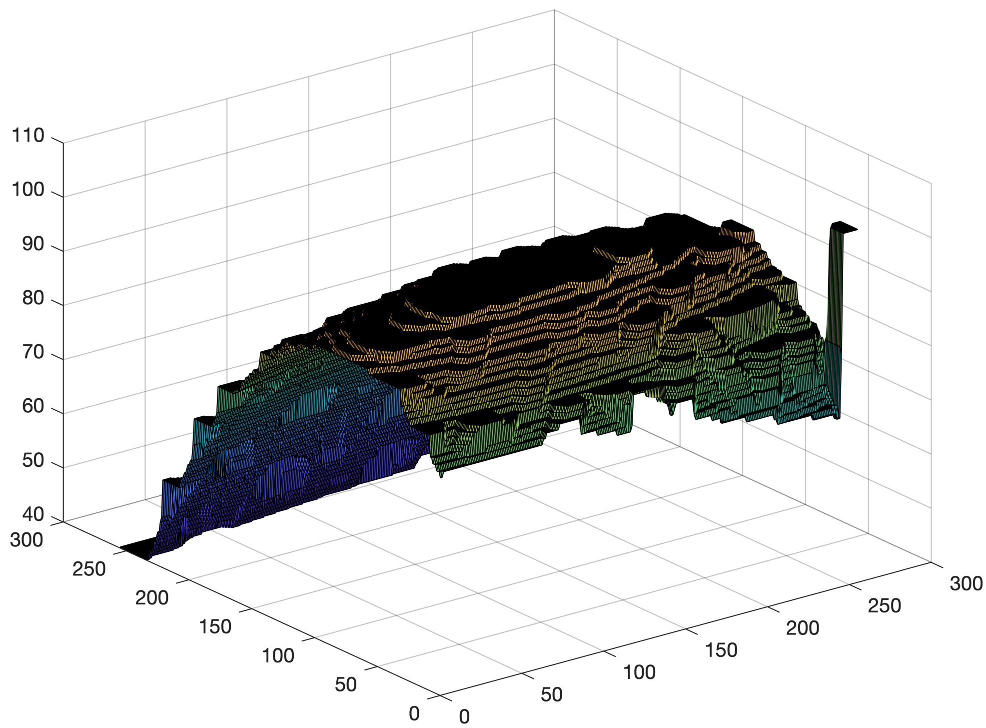


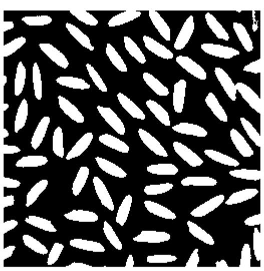
```
% No funciona, perdem grans. No confieu mai en els detectors automatics de llindar.
% El problema està en que la iluminació no és gens homogènea

% Fem un filtratge morfològic
bg = imopen(orig,strel('disk',10));
figure, imshow(bg),title('background')
```

```
figure, surf(bg) % mostrem el background com si fos una superfície
%Li restem el background a la imatge
y = imsubtract(orig,bg);
figure, imshow(y, []), title( 'top hat')
```



```
% tornem a segmentar per Otsu
bw=im2bw(y, graythresh(y));
figure, imshow(bw),title('Threshold automatic')
```





```
%Ara si!

% etiquetem la imatge segmentada
[eti num] = bwlabel(bw,4);
figure,imshow(label2rgb(eti)), title('imatge etiquetada')
```

```
% Comptem ara el nombre d’objectes a la imatge
num
```

```
num = 81
```

```
%o be:
max(eti(:))
```

```
ans = 81
```

```
% obtenim descriptors de les regions
Dades = regionprops(eti,'all')
```

```
Dades = 81×1 struct
```

...

Fields	Area	Centroid	BoundingBox	SubarrayIdx	MajorAxis...	MinorAxis...	Eccentric...	Orientation
1	80	[4.9500...	[0.5000...	1×2 cell	11.6630	9.3483	0.5979	30.7742
2	240	[10.462...	[0.5000...	1×2 cell	27.9231	11.3004	0.9145	36.4357
3	24	[1.8333...	[0.5000...	1×2 cell	10.0118	3.3478	0.9424	86.4153
4	28	[2.6429...	[0.5000...	1×2 cell	7.4739	5.1280	0.7275	72.3760
5	34	[2.5588...	[0.5000...	1×2 cell	9.0735	5.2263	0.8174	-84.2532
6	118	[7.6102...	[0.5000...	1×2 cell	17.6286	9.0561	0.8580	-2.7905
7	134	[7.7910...	[0.5000...	1×2 cell	17.9600	9.9834	0.8313	-11.1770
8	102	[5.4412...	[0.5000...	1×2 cell	13.0454	10.4306	0.6006	49.3510
9	123	[6.5122...	[0.5000...	1×2 cell	16.8383	9.9027	0.8088	35.0677
10	105	[5.3905...	[0.5000...	1×2 cell	15.7145	8.9702	0.8211	53.5471
11	269	[16.728...	[11.500...	1×2 cell	34.1454	10.2338	0.9540	82.8446
12	278	[18.194...	[11.500...	1×2 cell	33.4122	10.7958	0.9464	75.4553
13	284	[28.707...	[12.500...	1×2 cell	32.3537	11.3416	0.9365	6.5124
14	236	[34.131...	[18.500...	1×2 cell	30.3216	10.1916	0.9418	0.8335
15	252	[31.384...	[19.500...	1×2 cell	30.0630	10.8257	0.9329	-34.8038
16	271	[37.649...	[21.500...	1×2 cell	32.5210	10.9652	0.9414	-21.7028
17	272	[32.838...	[22.500...	1×2 cell	32.9782	10.8624	0.9442	51.5705
18	306	[36.957...	[27.500...	1×2 cell	36.3770	10.8760	0.9543	66.1907
19	250	[33.984...	[29.500...	1×2 cell	38.0818	8.4392	0.9751	-84.1484
20	339	[51.395...	[35.500...	1×2 cell	36.3565	12.0595	0.9434	-40.4248
21	206	[51.723...	[43.500...	1×2 cell	25.3071	10.6524	0.9071	62.2859
22	264	[55.015...	[49.500...	1×2 cell	31.3634	11.1410	0.9348	-86.9591
23	290	[69.696...	[52.500...	1×2 cell	34.0698	10.9740	0.9467	-11.5906

Fields	Area	Centroid	BoundingBox	SubarrayIdx	MajorAxis...	MinorAxis...	Eccentric...	Orientation
24	322	[67.260...	[54.500...	1×2 cell	35.4319	11.7302	0.9436	-45.0976
25	304	[78.559...	[60.500...	1×2 cell	33.6085	11.6939	0.9375	-4.2333
26	283	[76.356...	[62.500...	1×2 cell	37.5361	9.7797	0.9655	-44.6195
27	317	[77.066...	[62.500...	1×2 cell	36.1336	11.4212	0.9487	46.5021
28	73	[75.109...	[62.500...	1×2 cell	23.6975	4.5976	0.9810	-1.1281
29	240	[81.587...	[66.500...	1×2 cell	29.0056	10.8037	0.9280	-25.8726
30	300	[83.920...	[66.500...	1×2 cell	37.0459	10.4514	0.9594	-11.8054
31	281	[77.010...	[68.500...	1×2 cell	31.8524	11.4167	0.9336	67.5990
32	220	[93.036...	[82.500...	1×2 cell	34.5131	8.2618	0.9709	-56.1381
33	331	[99.770...	[89.500...	1×2 cell	38.4465	11.1100	0.9573	-62.3274
34	305	[107.22...	[91.500...	1×2 cell	35.0086	11.2794	0.9467	-30.4602
35	357	[108.74...	[93.500...	1×2 cell	37.9577	12.2842	0.9462	-47.2962
36	307	[110.62...	[95.500...	1×2 cell	33.7683	11.7664	0.9373	-25.7948
37	242	[108.58...	[99.500...	1×2 cell	32.6218	9.6997	0.9548	-58.5434
38	2	[101.50...	[100.50...	1×2 cell	2.3094	1.1547	0.8660	0
39	275	[112.26...	[102.50...	1×2 cell	33.6722	10.6200	0.9490	-55.1772
40	300	[120.42...	[104.50...	1×2 cell	34.6939	11.1857	0.9466	20.2589
41	281	[120.68...	[112.50...	1×2 cell	32.5825	11.2345	0.9387	70.6183
42	316	[127.23...	[113.50...	1×2 cell	36.9406	11.0964	0.9538	-43.7388
43	275	[130.73...	[120.50...	1×2 cell	33.1179	10.7380	0.9460	-51.9086
44	74	[126.50...	[121.50...	1×2 cell	12.9244	7.8927	0.7919	79.3483
45	339	[148.89...	[129.50...	1×2 cell	36.4467	12.1623	0.9427	1.5157
46	339	[148.72...	[134.50...	1×2 cell	38.5445	11.3185	0.9559	-43.9575
47	331	[153.83...	[139.50...	1×2 cell	36.3317	11.8749	0.9451	-34.2374
48	291	[151.97...	[139.50...	1×2 cell	33.2645	11.3673	0.9398	-41.7872
49	239	[154.36...	[140.50...	1×2 cell	30.3035	10.3280	0.9401	9.8093
50	305	[155.86...	[140.50...	1×2 cell	35.1045	11.2148	0.9476	-25.6703
51	296	[155.39...	[142.50...	1×2 cell	33.4333	11.5046	0.9389	-47.8218
52	283	[159.42...	[143.50...	1×2 cell	31.8017	11.4791	0.9326	-1.8220
53	203	[157.53...	[143.50...	1×2 cell	31.1438	8.4082	0.9629	2.1874
54	315	[156.73...	[145.50...	1×2 cell	35.7782	11.3928	0.9479	-56.6152
55	317	[186.65...	[168.50...	1×2 cell	35.3175	11.7111	0.9434	-10.9410
56	328	[175.70...	[169.50...	1×2 cell	34.8753	12.2969	0.9358	83.5706
57	355	[183.82...	[169.50...	1×2 cell	36.3663	12.6942	0.9371	-38.9496

Fields	Area	Centroid	BoundingBox	SubarrayIdx	MajorAxis...	MinorAxis...	Eccentric...	Orientation
58	354	[181.38...	[170.50...	1×2 cell	37.2839	12.3661	0.9434	64.7149
59	354	[188.05...	[172.50...	1×2 cell	36.8032	12.3857	0.9417	-36.1485
60	281	[195.25...	[177.50...	1×2 cell	37.6251	9.7787	0.9656	-0.5020
61	151	[189.82...	[181.50...	1×2 cell	19.9604	10.7372	0.8430	66.4054
62	281	[196.51...	[183.50...	1×2 cell	34.3811	10.6768	0.9506	-38.4346
63	338	[204.40...	[186.50...	1×2 cell	38.6849	11.3622	0.9559	30.9818
64	365	[215.91...	[197.50...	1×2 cell	38.6120	12.2885	0.9480	-11.3472
65	279	[217.73...	[200.50...	1×2 cell	37.7482	9.5230	0.9677	-21.9400
66	292	[212.81...	[201.50...	1×2 cell	34.8991	10.9113	0.9499	-50.8978
67	299	[212.39...	[201.50...	1×2 cell	34.4504	11.4433	0.9432	-55.9024
68	692	[234.44...	[210.50...	1×2 cell	63.5988	17.7302	0.9604	-37.1596
69	64	[217.90...	[211.50...	1×2 cell	12.3831	7.1794	0.8148	-2.1919
70	297	[238.01...	[221.50...	1×2 cell	34.2019	11.2179	0.9447	20.7580
71	264	[242.71...	[225.50...	1×2 cell	32.6771	10.6585	0.9453	-4.9936
72	324	[237.28...	[226.50...	1×2 cell	34.0888	12.3606	0.9319	-54.0957
73	289	[239.64...	[229.50...	1×2 cell	33.9883	11.0280	0.9459	-60.9361
74	7	[235.14...	[232.50...	1×2 cell	5.1257	2.1156	0.9108	-4.3078
75	202	[241.94...	[235.50...	1×2 cell	25.2405	10.7626	0.9045	84.9135
76	242	[247.50...	[236.50...	1×2 cell	33.9498	9.2991	0.9618	-55.5549
77	158	[249.82...	[239.50...	1×2 cell	18.1335	12.4404	0.7276	19.7132
78	136	[249.57...	[240.50...	1×2 cell	17.5357	10.3733	0.8063	19.7048
79	65	[250.67...	[248.50...	1×2 cell	23.9697	5.0098	0.9779	-84.8954
80	192	[253.18...	[249.50...	1×2 cell	33.8614	7.3735	0.9760	-86.9613
81	1	[254,11]	[253.50...	1×2 cell	1.1547	1.1547	0	0

```
% Comproveu en el workspace la variable Dades. Quines propietats hem extret de cada reg
% Consulteu el help de la funció regionprops.
 % Podem obtenir una dada de l'objecte 50 de la forma:
Dades(50).Area
```

```
ans = 305
```

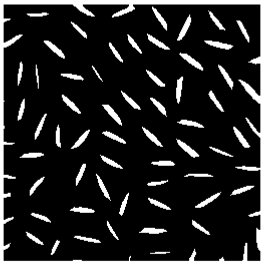
```
% O be un vector amb totes les Àrees:
Arees=[Dades.Area];
```

Exercici

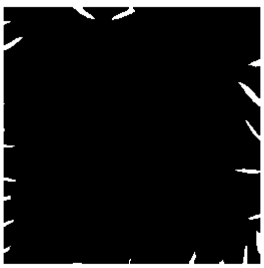
En la imatge segmentada, separeu els grans d%arròs que es toquen. Desprès elimineu els grans d'arròs que toquen les vores. Tots aquests grans falsejaven el resultat. Amb la nova imatge etiquetatda, obteniu les propietats que us semblin adients usant regionprops.

Representeu mitjançant plots o diagrames de barres aquelles propietats que us semblin interessants per a fer un control de qualitat dels grans d'arròs.

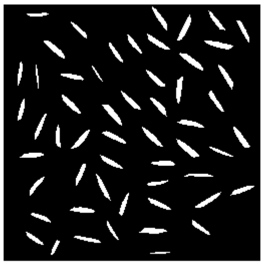
```
% Separar grans d'arros
ee = strel('disk',4);
seg = imerode(bw, ee);
imshow(seg)
```



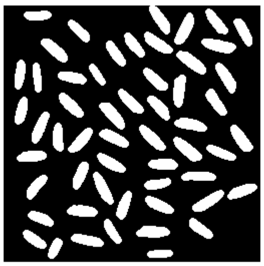
```
% eliminar grans d'arros de les vores
mark = true(size(seg));
mark(2:end-1, 2:end-1) = 0;
touchingBorders = imreconstruct(mark, seg);
imshow(touchingBorders)
```



```
noBorders = seg - touchingBorders;
imshow(noBorders)
```



```
final = imreconstruct(logical(noBorders), seg);
final = imdilate(final, ee);
imshow(final)
```



```
% labeling
[eti num] = bwlabel(final,4);
figure,imshow(label2rgb(eti)), title('imatge etiquetada')
```

```
% obtenim descriptors de les regions
Dades = regionprops(eti,'all')
```

Dades = 55×1 struct

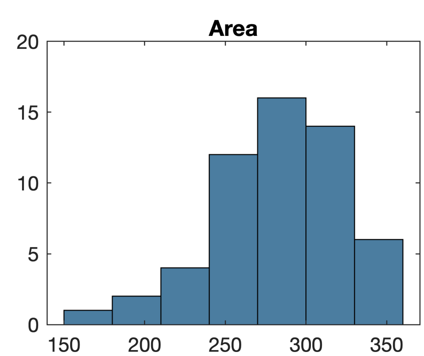
...

Fields	Area	Centroid	BoundingBox	SubarrayIdx	MajorAxis...	MinorAxis...	Eccentric...	Orientation
1	251	[16.808...	[11.500...	1×2 cell	31.2657	10.3360	0.9438	82.1038
2	254	[18.315...	[11.500...	1×2 cell	29.5360	11.0580	0.9273	75.5370

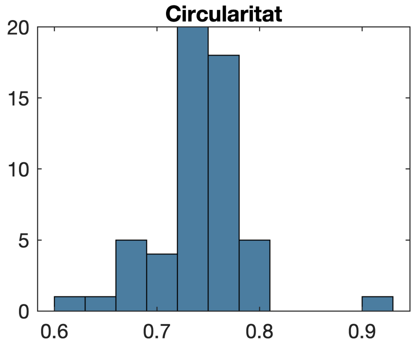
Fields	Area	Centroid	BoundingBox	SubarrayIdx	MajorAxis...	MinorAxis...	Eccentric...	Orientation
3	274	[28.799...	[13.500...	1×2 cell	31.0037	11.3707	0.9303	5.5293
4	241	[31.215...	[19.500...	1×2 cell	28.6737	10.8327	0.9259	-35.1751
5	227	[34.555...	[20.500...	1×2 cell	29.2617	10.1139	0.9384	1.8423
6	260	[33.084...	[23.500...	1×2 cell	30.9512	11.0082	0.9346	50.8396
7	242	[37.756...	[24.500...	1×2 cell	28.5881	10.9563	0.9236	-22.8072
8	301	[36.976...	[28.500...	1×2 cell	35.7896	10.8512	0.9529	66.5302
9	216	[34.064...	[29.500...	1×2 cell	31.9847	8.7298	0.9620	-84.5744
10	324	[51.506...	[37.500...	1×2 cell	34.4719	12.1262	0.9361	-40.8236
11	192	[51.817...	[44.500...	1×2 cell	23.3978	10.6133	0.8912	63.3888
12	256	[54.929...	[49.500...	1×2 cell	30.1958	11.1126	0.9298	-87.1932
13	266	[69.090...	[54.500...	1×2 cell	30.5977	11.2122	0.9304	-11.5549
14	309	[67.563...	[55.500...	1×2 cell	33.4572	11.8878	0.9347	-44.3990
15	291	[79.048...	[63.500...	1×2 cell	31.6518	11.8615	0.9271	-4.2815
16	241	[75.958...	[64.500...	1×2 cell	30.4171	10.3064	0.9408	-44.6110
17	300	[77.676...	[64.500...	1×2 cell	33.4108	11.6214	0.9376	47.2778
18	287	[84.090...	[67.500...	1×2 cell	35.0833	10.5643	0.9536	-11.5335
19	272	[77.154...	[69.500...	1×2 cell	30.8527	11.3608	0.9297	67.2949
20	176	[93.795...	[85.500...	1×2 cell	26.5366	8.6120	0.9459	-55.9430
21	324	[99.654...	[89.500...	1×2 cell	37.4850	11.1370	0.9548	-62.4315
22	294	[107.53...	[93.500...	1×2 cell	33.2269	11.3983	0.9393	-29.9558
23	341	[109.28...	[95.500...	1×2 cell	35.6451	12.4351	0.9372	-48.1960
24	304	[110.67...	[95.500...	1×2 cell	33.3566	11.8093	0.9352	-25.7125
25	223	[108.76...	[100.50...	1×2 cell	29.5241	9.8655	0.9425	-58.1581
26	260	[112.21...	[102.50...	1×2 cell	31.3423	10.7166	0.9397	-54.9496
27	270	[120.65...	[112.50...	1×2 cell	31.1357	11.2578	0.9323	70.2458
28	298	[127.09...	[114.50...	1×2 cell	34.3910	11.2370	0.9451	-43.8780
29	260	[130.67...	[120.50...	1×2 cell	30.7506	10.9362	0.9346	-51.6952
30	324	[149.43...	[132.50...	1×2 cell	34.4078	12.1903	0.9351	0.6966
31	320	[148.57...	[135.50...	1×2 cell	35.5864	11.5554	0.9458	-44.1058
32	285	[151.88...	[139.50...	1×2 cell	32.4089	11.4331	0.9357	-42.0517
33	319	[154.08...	[140.50...	1×2 cell	34.6377	11.9851	0.9382	-33.5396
34	229	[153.98...	[140.50...	1×2 cell	28.7579	10.3737	0.9327	9.9048
35	288	[155.49...	[141.50...	1×2 cell	32.5057	11.4086	0.9364	-25.6694
36	285	[155.59...	[143.50...	1×2 cell	31.8079	11.6239	0.9308	-48.1762

Fields	Area	Centroid	BoundingBox	SubarrayIdx	MajorAxis...	MinorAxis...	Eccentric...	Orientation
37	183	[156.53...	[143.50...	1×2 cell	27.5423	8.5670	0.9504	2.7082
38	276	[159.79...	[144.50...	1×2 cell	30.7375	11.5709	0.9264	-1.6807
39	307	[156.57...	[145.50...	1×2 cell	34.5406	11.4898	0.9431	-56.2939
40	325	[175.70...	[169.50...	1×2 cell	34.5087	12.2929	0.9344	83.3505
41	351	[183.81...	[169.50...	1×2 cell	35.8386	12.6989	0.9351	-39.1780
42	345	[181.21...	[170.50...	1×2 cell	35.9154	12.4739	0.9377	64.2355
43	310	[187.02...	[170.50...	1×2 cell	34.2456	11.7773	0.9390	-11.4663
44	350	[188.06...	[172.50...	1×2 cell	36.2484	12.4441	0.9392	-36.1139
45	278	[195.06...	[177.50...	1×2 cell	37.1472	9.8030	0.9646	-0.4128
46	255	[196.72...	[185.50...	1×2 cell	30.3585	10.9585	0.9326	-37.0106
47	331	[204.55...	[188.50...	1×2 cell	37.6663	11.3948	0.9531	31.3157
48	349	[215.21...	[197.50...	1×2 cell	36.3872	12.4360	0.9398	-12.0873
49	280	[212.63...	[201.50...	1×2 cell	33.0507	11.0362	0.9426	-51.1172
50	288	[212.24...	[201.50...	1×2 cell	32.8453	11.5292	0.9364	-55.5427
51	252	[217.32...	[202.50...	1×2 cell	33.2511	9.7873	0.9557	-22.1733
52	322	[222.51...	[211.50...	1×2 cell	35.5024	11.8246	0.9429	-58.8647
53	285	[238.21...	[223.50...	1×2 cell	32.2693	11.3817	0.9357	20.3450
54	319	[237.35...	[226.50...	1×2 cell	33.3789	12.4084	0.9283	-54.0445
55	279	[239.76...	[229.50...	1×2 cell	32.5890	11.1165	0.9400	-61.7474

```
% Observem les distribucions de diferents atributs
Arees=[Dades.Area];
histogram(Arees), title('Area')
```




```
Circularities = [Dades.Circularity];
histogram(Circularities), title('Circularitat')
```



Codis de cadena

```
clear all, close all
im=imread('head.png');
% no cal tan gran
im=imresize(im,1/4);
imshow(im), title('imatge original')
```

```
% obtenim el contorn
ero=imerode(im,strel('disk',1));
cont=xor(ero,im);
figure,imshow(cont), title('contorns')
```

```
% obtenim les coordenades del contorn
[fila col] = find(im,1); % Busquem el primer píxel
B = bwtraceboundary(im,[fila col],'E'); %direccio est a l'atzar
% B conté les coordenades

% Ho comprovem mostrant el resultat
aux=zeros(size(im));
aux(sub2ind(size(aux),B(:,1),B(:,2)))=1;
figure,imshow(aux),title ('contorns a partir de coordenades')
```

Exercici

Trobar els codis de cadena incrementals a partir de B

```
incrementalChain = codiCadenaIncremental(B);
```

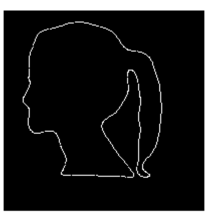
```
incrementalChain = 753×1
     0
     0
     0
     0
     0
     1
     7
     0
     1
     7
     ⋮
```

Descriptors de Fourier

```
% centrem coordenades
mig=mean(B);
B(:,1)=B(:,1)-mig(1);
B(:,2)=B(:,2)-mig(2);

% Convertim les coordenades a complexes
s= B(:,1) + i*B(:,2);

% Cal que la dimensio del vector sigui parell
[mida bobo]=size(B);
if(mida/2~=round(mida/2))
    s(end+1,:)=s(end,:); %dupliquem l'ultim
    mida=mida+1;
end

% Calculem la Fast Fourier Transform
z=fft(s);
% representem l'espectre
figure,plot(abs(z))
% ho displaiem logaritmic perque no es veu res
figure,plot(log(abs(z))), title ('descriptors de Fourier')

% Recuperem la imatge original per comprovar que el procés és reversible
```

```
ss=ifft(z); % Transformada de Fourier inversa
files= round(real(ss)+mig(1));
cols=round(imag(ss)+mig(2));
aux(:,:)=0;
aux(sub2ind(size(aux),files,cols))=1;
figure,imshow(aux), title('imatge recuperada')
```

```
%% Reduim la quantitat de descriptors de Fourier
N=30; % agafem N descriptors
tmp=z;
tmp(N+1:end-N)=0; % eliminem els del mig perque es duplica l'espectre
figure, plot(log(abs(tmp))), title ('Només 30 descriptors')

% Tornem al pla imatge a partir de l'espectre modificat
ss2=ifft(tmp);
% Les coordenades resultants poden sortir del rang de la imatge original
% Creo una imatge més gran per a que les coordenades no s'em surtin de mare
mida=200;
files= round(real(ss2)+mida/2);
cols=round(imag(ss2)+mida/2);
aux=logical(zeros(mida));
aux(sub2ind(size(aux),files,cols))=1;
figure,imshow(aux)
title(['Numero de descriptors: ',num2str(N)]);
```



Exercici

Repetiu el procés utilitzant diferents quantitats de descriptors i expliqueu quin efecte té això en el detall de la imatge obtinguda

```
N=10; % agafem N descriptors
tmp=z;
tmp(N+1:end-N)=0; % eliminem els del mig perque es duplica l'espectre
figure, plot(log(abs(tmp))), title ('Només 10 descriptors')
```

```
% Tornem al pla imatge a partir de l’espectre modificat
ss2=ifft(tmp);
% Les coordenades resultants poden sortir del rang de la imatge original
% Creo una imatge més gran per a que les coordenades no s'em surtin de mare
mida=200;
files= round(real(ss2)+mida/2);
cols=round(imag(ss2)+mida/2);
aux=logical(zeros(mida));
aux(sub2ind(size(aux),files,cols))=1;
figure,imshow(aux)
```

```
title(['Numero de descriptors: ',num2str(N)]);
```

```
N=20; % agafem N descriptors
tmp=z;
tmp(N+1:end-N)=0; % eliminem els del mig perque es duplica l'espectre
figure, plot(log(abs(tmp))), title ('Només 20 descriptors')
```

```
% Tornem al pla imatge a partir de l’espectre modificat
ss2=ifft(tmp);
% Les coordenades resultants poden sortir del rang de la imatge original
% Creo una imatge més gran per a que les coordenades no s'em surtin de mare
mida=200;
files= round(real(ss2)+mida/2);
cols=round(imag(ss2)+mida/2);
aux=logical(zeros(mida));
aux(sub2ind(size(aux),files,cols))=1;
figure,imshow(aux)
```

```
title(['Numero de descriptors: ',num2str(N)]);
```

Explicació: Quant més gran és el número de descriptors de fourier emprats, el nivell de fidelitat del contorn és més elevat. Per tant, en quant més descriptors s'utilitzin, més semblant serà el contorn a la imatge original.

Funcions

```
function incrementalChain = codiCadenaIncremental(B)
    [n m] = size(B);
    chain = zeros(n-1, 1);

    % calcul del codi de cadena
    for i=1:(n-1)
        D = B(i+1, :)-B(i, :);
        if (D(1) == -1 && D(2) == 0)
            chain(i) = 1;
        elseif(D(1) == -1 && D(2) == -1)
            chain(i) = 2;
        elseif(D(1) == 0 && D(2) == -1)
            chain(i) = 3;
        elseif(D(1) == 1 && D(2) == -1)
            chain(i) = 4;
        elseif(D(1) == 1 && D(2) == 0)
            chain(i) = 5;
        elseif(D(1) == 1 && D(2) == 1)
            chain(i) = 6;
        elseif(D(1) == 0 && D(2) == 1)
            chain(i) = 7;
        elseif(D(1) == -1 && D(2) == 1)
            chain(i) = 8;
        end

    end

    % cadena incremental
    incrementalChain = mod(diff(chain), 8)
end
```