

VISIÓ PER COMPUTADORS

Informe Sessió 7

Carles Llongueras

Carlos Plana García

Pablo Gámiz Nieto

Curs 2021-2022

Visualització de la imatge

Primer, llegim la imatge i comprovem que es una imatge rgb mirant el tamany de la 3a dimensió d'aquesta. El valor retornat es 3, concluint que la imatge seleccionada és rgb.

```
I = imread("bird.jpg");  
size(I, 3)
```

```
ans = 3
```

A continuació, mostrem la imatge.

```
imshow(I);
```



Amb **getrect**, obtenim els valors de *xmin*, *ymin*, *width*, *height* d'un rectangle que emmarca l'objecte que volem, en aquest cas l'ocell.

```
rect = getrect;
```

```
rect = 1x4  
    21.0000    56.0000   178.0000    83.0000
```

Kmeans

Convertim la imatge de *RGB* a *HSV* i guardem les seves components per usar-les més endavant.

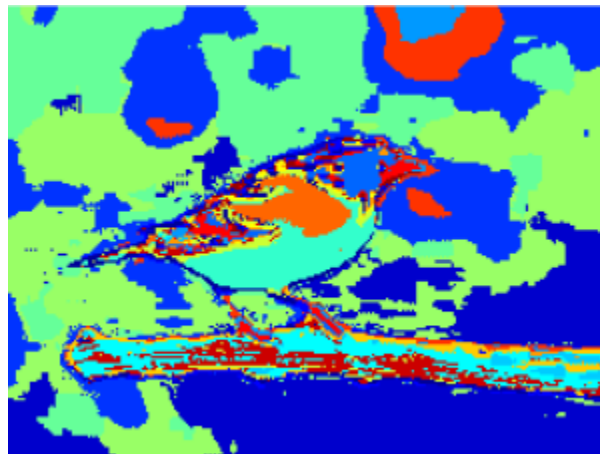
```
I2 = rgb2hsv(I);  
[x, y, z] = size(I2);  
Ha = I2(:,:,1)*2*pi;
```

Després obtenim la H_x i la H_y fent el cosinus i el sinus respectivament del perímetre del HSV . I després obtenim la *saturació* i la *lluentor* per poder crear un vector amb totes les components per poder fer el Kmeans.

```
Hx = reshape(cos(Ha), [x*y,1]);  
Hy = reshape(sin(Ha), [x*y,1]);  
S = reshape(I2(:, :, 2), [x*y,1]);  
V = reshape(I2(:, :, 3), [x*y,1]);  
O = [Hx, Hy, S, V];  
k = 20;  
C = kmeans(O, k);  
IC = reshape(C, x, y);
```

Fem un *labeling* amb la imatge.

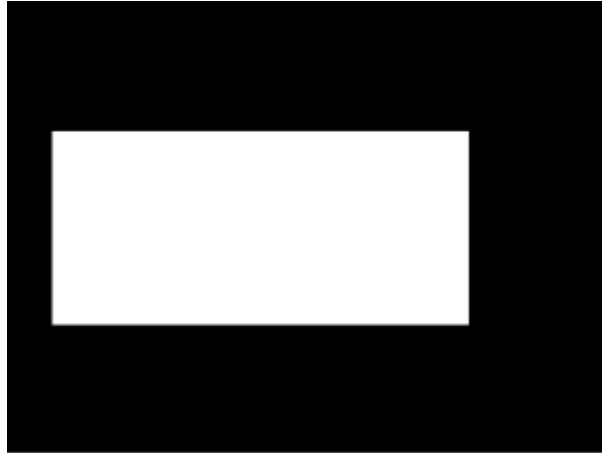
```
rgb = label2rgb(IC);  
imshow(rgb);
```



Selecció dels colors dins del rectangle

Creem una màscara binària que té el tamany i la forma del rectangle que havien creat abans per emmarcar l'objecte desitjat. Aquesta ens servirà per determinar quins píxels estan dins o fora el rectangle.

```
a = [rect(1) rect(1)+rect(3) rect(1)+rect(3) rect(1)];  
b = [rect(2) rect(2) rect(2)+rect(4) rect(2)+rect(4)];  
MASK = poly2mask(a, b, x, y);  
imshow(MASK);
```



Una vegada tenim els píxels que son dins del rectangle, hem de veure quins colors han caigut a dins d'aquest. Per fer això, fem una concatenació dels colors que ha retornat el kmeans i la màscara abans obtenida. Aquest resultat el guardem en una variable H, que indica si el color C es troba dins la màscara o no.

```
H = [C, MASK(:)];
```

A continuació, creem dos vectors que guardaran el nombre de píxels de cada un dels 20 colors del kmeans que han caigut a fora o a dins del rectangle, respectivament. Per cada un dels píxels de la matriu H, mirem si el valor de la màscara es equivalent a 1 (el píxel es troba a dins del rectangle). Si ha sigut aquest el cas, incrementem en 1 el valor en la posició del color corresponent del vector hist1. En cas contrari, fem el mateix amb hist0.

```
hist0 = linspace(0,0,20)';  
hist1 = linspace(0,0,20)';  
for i = 1:x*y  
    if (H(i,2) == 1)  
        hist1(H(i,1)) = hist1(H(i,1)) + 1;  
    else  
        hist0(H(i,1)) = hist0(H(i,1)) + 1;  
    end  
end
```

Mirem quins colors tenen més píxels a dins que fora del rectangle i guardem el resultat en RES. Seguidament, recorrem tots el píxels de la matriu H veient si el color d'aquest té un 1 al vector RES. En cas afirmatiu, posem en 1 el píxel d'una nova variable M.

```
RES = hist1 > hist0;  
  
M = zeros(x*y, 1);  
for i = 1:size(H,1)  
    if (RES(H(i,1)) == 1)
```

```
        M(i) = 1;
    end
end
```

Per últim, redimensionem el resultat que es troba a M per a que tingui les dimensions de la imatge original, i la mostregem.

```
M = reshape(M, [x y]);
imshow(M);
```



Supressió del soroll

Definim un element estructurant de tipus disk que fem servir per eliminar petites parts de l'objecte i del fons.

```
ee = strel('disk', 2);
M = imclose(M, ee);
M = imopen(M, ee);
```

La imatge final és:

```
imshow(M);
```



Comparant aquesta imatge amb la de dalt, podem veure que han desaparegut petites regions de la blanca i, a més, petits trossos de fons que es trobaven a l'ocell.