



**FIB**

Facultat d'Informàtica  
de Barcelona

Departament d'Enginyeria de Sistemes,  
Automàtica i Informàtica Industrial

UNIVERSITAT POLITÈCNICA DE CATALUNYA

# VISIÓ PER COMPUTADOR

## Exercici 4 de Laboratori

Facultat d'Informàtica de Barcelona

Adrian Cristian Crisan

Barcelona, Octubre de 2021

## 1. Condicions de la sessió

- Haureu d'implementar la solució amb CODI PROPI, usant comandes matlab i funcions de la 'Image Processing Toolbox'. NO ES PERMET USAR NI COPIAR IMPLEMENTACIONS DESENVOLUPADES PER ALTRES.
- La pràctica és INDIVIDUAL i no s'acceptaran solucions realitzades entre estudiants.
- Realitzeu els exercicis proposats i ho entregueu al racó en un informe en **format pdf**. En l'informe, detalleu el programa utilitzat i els resultats obtinguts.

## 2. Exercici de la sessió

Es vol tenir una útil aplicació de visió per computador que ajudi a trobar a Odlaw, utilitzant el coneixement que la samarreta d'Odlaw està composta de franges negres i grogues. Amb aquest objectiu, caldrà crear un imatge que sigui una mena de mapa tèrmic; que indiqui on hi han franges negres i grogues (horitzontals i amb prou gruix) en la imatge. Elimineu les deteccions espúries (petites deteccions) i feu una superposició del mapa tèrmic (en vermell) amb la imatge original (en gris).



### 3. Passes a seguides (son les opcionals)

1. Llegim la imatge RGB, apliquem un filtre medfilt2 [1 4] per treure soroll.

```
I = imread('Wally.png');  
R = I(:, :, 1);  
G = I(:, :, 2);  
B = I(:, :, 3);  
JR = medfilt2(R, [1 4]);  
JG = medfilt2(G, [1 4]);  
JB = medfilt2(B, [1 4]);
```

Per poder utilitzar el filtre medfilt2 hem agut de separar la matriu en tres dimensions, ja que el filtre només pot operar en 2 dimensions.

2. Convertim a un espai de color HSV, Lab o similar.

```
RGB = cat(3, JR, JG, JB);  
HSV = rgb2hsv(RGB);
```

3. Detecció del color groc per  $HUE = 50^\circ \pm \text{tolerància}$

Donem una tolerància d'un  $\pm 10^\circ$  i donat que el HUE té  $360^\circ$  el dividim per aquest nombre per obtenir un rang entre 0 i 1 per poder operar amb la matriu.

```
Yellow = (HSV(:, :, 1) < 60/360) & (HSV(:, :, 1) > 40/360);  
imshow(Yellow)
```



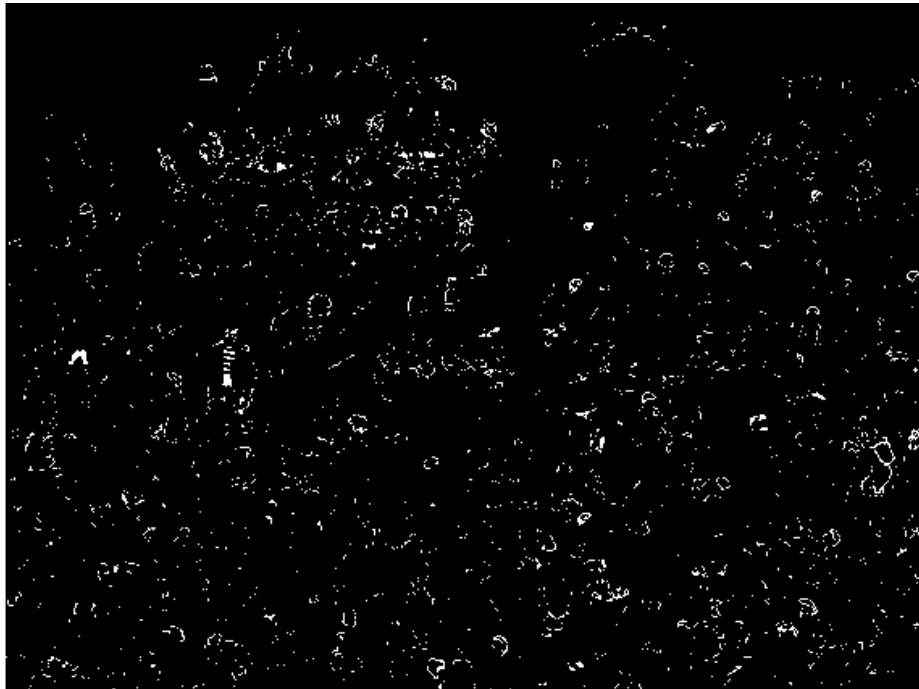
Fig. HUE del color groc

Obtenim la imatge únicament amb el color groc.

4. Detectem el color negre (tres components RGB de valor baix)

Establim un límit de 70 pel color negre. Aquest valor l'hem establert a partir de provar amb diferents límits.

```
black_lim = 70;
Black = (R < black_lim) & (B < black_lim) & (B < black_lim);
imshow(Black)
```



Obtenim una imatge únicament amb el color negre.

5. Filtrat per detectar patrons de línies **grogues** horitzontals SE7 = strel('line',7,90);

```
SED = strel('line', 7, 90);
```

6. Ajuntem línies properes

```
IY = imdilate(Yellow, SED);
```

7. Eliminem les línies aïllades

El radi de 17 l'hem establert a partir de provar diferents valors per tal de que erosioni lo suficient per eliminar les solucions invàlides que detecta.

```
SER = strel('line', 17, 90);
IY = imerode(IY, SER);
```

8. Filtrat per detectar patrons de línies **negres** horitzontals

Utilitzem els mateixos filrats que pel color groc.

9. Ajuntem línies properes

```
IB = imdilate(Black, SED);
```

10. Eliminem les línies aïllades

```
IB = imerode(IB, SER);
```

11. Unim els dos resultats fent una **and** lògica. RES = GROC&NEGRE;

```
res = IY & IB;
```

12. Dilatem els píxels “supervivents” de l'anterior operació.



```
SE = strel('disk', 7);  
res = imdilate(res, SE);
```

13. Realitzem una composició de la imatge original en gris i el resultat en vermell superposat.

```
res = cat(3, res*255, res*0, res*0);  
IFinal = imfuse(res, rgb2gray(I), 'blend');  
imshow(IFinal);
```

