



**FIB**

Facultat d'Informàtica  
de Barcelona

Departament d'Enginyeria de Sistemes,  
Automàtica i Informàtica Industrial

UNIVERSITAT POLITÈCNICA DE CATALUNYA

# VISIÓN POR COMPUTADOR

## Ejercicio 8 de Laboratorio

Facultad de Informática de Barcelona

Adrian Cristian Crisan

Filip Gedung Dorm

Pablo Vega Gallego

Barcelona, noviembre de 2021

## Índice

Introducción.....	1
Metodología .....	2

## Introducción

El objetivo de la sesión es el de conocer las posibilidades de los descriptores de forma de objetos binarizados y , sin entrar en muchos detalles, utilizar uno de los muchos clasificadores que tiene Matlab. En concreto trabajaremos con los siguientes conceptos:

- Caracterización de regiones: regionprops y otras.
- Construcción de un clasificador de objetos: TreeBagger.
- Clasificador de objetos: Predict

Los objetos a clasificar que se utilizarán en la práctica son los juegos de caracteres de las matrículas españolas que aparecen en la imagen *Joc\_de\_caràcters.jpg* disponible en la carpeta de la asignatura.

**0 1 2 3 4 5 6 7 8 9 B C D F G H J K L M N P R S T V W X Y Z**

Para comprobar la fiabilidad de la clasificación usaremos las siguientes imágenes que han estado deformadas en mida y ángulo, como se muestra en la siguiente imagen:

**0 1 2 3 4 5 6 7 8 9 B C D F G H J K L M N P R S T V W X Y Z**

Introduciremos los descriptores que nos parezcan adecuados para intentar optimizar la puntuación (score) en las predicciones de las imágenes de test.

## Metodología

Abrimos la imagen para entrenar el clasificador y la binarizamos con un umbral fijo:

```
I = rgb2gray(imread('Joc_de_caracters.jpg'));  
BI = I < 128; % lletres estan en negre
```

Obtenemos la lista de píxeles que conforman cada uno de los caracteres segmentados:

```
CC = bwconncomp(BI);
```

Construimos el output o resultado que queríamos obtener a partir de las entradas:

```
OUT = {'0' '1' '2' '3' '4' '5' '6' '7' '8' '9' 'B' 'C' 'D' 'F' 'G' 'H' 'J' 'K'  
'L' 'M' 'N' 'P' 'R' 'S' 'T' 'V' 'W' 'X' 'Y' 'Z'};
```

Seguimos los mismos pasos pero en este caso para la imagen con la que deseamos testear el clasificador:

```
I2 = rgb2gray(imread('Joc_de_caracters_deformats.jpg'));  
BI2 = I2 < 128; % separem el background  
CC2 = bwconncomp(BI2);
```

Dado que nos piden que no podemos utilizar descriptores que varíen con la medida del carácter o la posición de éste, nos quedamos con los siguientes descriptores:

- Circularity
- Eccentricity
- EulerNumber
- Extent
- Solidity

Posteriormente, para obtener la mejor combinación entre estos descriptores usamos el algoritmo que dejamos a continuación:

```
%try different combinations in for loop of following parameters:  
desc = ["Circularity" "Eccentricity" "EulerNumber" "Extent" "Solidity"] ;  
%desc = ["Extent" "Solidity"] ;  
  
%should these be used?  
%MaxFeretProperties - MaxFeretAngle  
%MinFeretProperties - MinFeretAngle  
  
%making the for loop work in binary to activate and deactivate fields  
n = 2^length(desc);  
d_array = [1:n]';  
binary = dec2bin(d_array);  
  
%best combination and its values  
bestCombo = [];  
bestMean = 0;  
bestSD = 100000;  
bestTable = table();
```

```

bestScore = 0;
numCorrectLabels = 0;
bestClassifier = 0;

for i=1:n-1 %because we don't want to test for when chosen = [] (last case)
    chosen = [];
    currentBin = binary(i,:);
    %currentBin
    length(currentBin);
    for j=0:length(desc)-1
        currentBin(end-j);
        if currentBin(end-j) == '1'
            desc(:,end-j);
            chosen = [chosen, desc(:,end-j)];
        end
    end
    %train classifier
    props = regionprops(CC,chosen);
    X = cell2mat(struct2cell(props));
    Classifier = TreeBagger(100,X,OUT');

    %test with reference
    props2 = regionprops(CC2,chosen);
    X2 = cell2mat(struct2cell(props2));

    [label,score] = predict(Classifier,X2);

    % mostrem el resultat mes probable i la seva puntuació (score)
    maxScore = max(score,[],2);
    T =
table(Classifier.ClassNames,label,maxScore,'VariableNames',{'Name','Label','Score'});
    currMean = mean(maxScore);
    currSD = std(maxScore);
    currScore = currMean-currSD;
    currCorrectLabels = sum(strcmp(Classifier.ClassNames, label));

    %if values we measure better, replace and save info about combo
    if (currCorrectLabels > numCorrectLabels || currCorrectLabels ==
numCorrectLabels & currMean > bestMean)
        bestTable = T;
        bestCombo = chosen;
        bestMean = currMean;
        bestSD = currSD;
        bestScore = currScore;
        numCorrectLabels = currCorrectLabels;
        bestClassifier = Classifier;
    end
end
end

```

Obtenemos que la mejor combinación que nos da como resultado la siguiente tabla de puntuaciones:

	<i>Name</i>	<i>Label</i>	<i>Score</i>
1	'0'	'0'	0.6000
2	'1'	'1'	0.7800
3	'2'	'2'	0.3200
4	'3'	'3'	0.2500
5	'4'	'4'	0.5600
6	'5'	'5'	0.4300
7	'6'	'6'	0.6600
8	'7'	'7'	0.6100
9	'8'	'8'	0.4400
10	'9'	'9'	0.5500
11	'B'	'B'	0.5600
12	'C'	'C'	0.4500
13	'D'	'D'	0.6600
14	'F'	'F'	0.6700
15	'G'	'G'	0.6900
16	'H'	'H'	0.5700
17	'J'	'J'	0.6600
18	'K'	'K'	0.6700
19	'L'	'L'	0.5500
20	'M'	'M'	0.6500
21	'N'	'N'	0.5200
22	'P'	'P'	0.5200
23	'R'	'R'	0.5300
24	'S'	'S'	0.5300
25	'T'	'L'	0.5500
26	'V'	'V'	0.3800
27	'W'	'W'	0.5700
28	'X'	'X'	0.4900
29	'Y'	'Y'	0.4200
30	'Z'	'Z'	0.6300

Esta combinación está formada por los siguientes descriptores:

- Circularity
- Eccentricity
- Solidity

La media de puntuación es: 0.5363

La desviación estándar es: 0.1141

El número de etiquetas correctas es: 29

Si probamos con una matrícula (recortada), como la siguiente:



Y aplicamos:

```
I_license = imread('prueba.jpg');
I_license = ~imbinarize(rgb2gray(I_license));
CC3 = bwconncomp(I_license);

props3 = regionprops(CC3,bestCombo);
X3 = cell2mat(struct2cell(props3))';

Classifier = TreeBagger(100,X3, ['7'; '6'; '1'; '0'; 'J'; 'B'; 'B']);

[label,score] = predict(bestClassifier,X3);

% mostrem el resultat mes probable i la seva puntuació (score)
table(label,max(score,[],2), 'VariableNames',{'Label','Score'})
```

Obtenemos el siguiente resultado:

	<i>Label</i>	<i>Score</i>
1	'7'	0.4900
2	'6'	0.5100
3	'1'	0.7800
4	'0'	0.3900
5	'J'	0.5200
6	'B'	0.4900
7	'B'	0.5700