22437 - Industrial Vision
Final Project, 2022-2023

Universitat de les Illes Balears

Introduction

Computer vision techniques can be used for recognizing shapes or objects. A very useful task is the automatic recognition of car license plates. In this final project, you have to develop a simple car licence plate recognizer employing the different techniques learnt during this course. The car licence plates that will be considered by our recognizer are composed of the following elements:

- A set of white letters over a blue background to indicate the country of the license. These letters are always located to the left of the plate and can take the following values:
  - GB for Great Britain
  - D for Germany
  - PL for Poland
- Three capital letters, which can take the following values: A, B or C.
- Four numbers, which can take the following values: 0, 1, 7 or 8.

Some examples of these car license plates are shown in Fig. 1. The recognition module should be able to determine the registration number of a car license plate under these conditions.

Description of Work

Six sets of car license plates are distributed in folders (from *G1* to *G6*) along with this file. Each group of students will work with its own folder, according to the identifier of the group. The implementation of the recognition module must be written mainly in two files, *main.m* and *process_license_plate.m*, following the guidelines explained below. If it is required to write additional functions, they should be also included in the final documentation and zip file.

- The entry point of the recognition module is the *main.m* file. Here, you have to implement a loop for loading, in the correct order, the images stored in the directory of your corresponding group. Then, each image have to be passed as input argument to the *process_license_plate* function, where license plate recognition will be performed. The output of the *process_license_plate* function is a string indicating the detected registration number, and it should be displayed on the screen from the main file using the *disp* function.
- The code needed for processing the images must be included in the *process_license_plate* function, inside the *process_license_plate.m* file. This function must perform the required digital image processing techniques to determine the correct license number. As a hint, you will find very useful the Matlab function **regionprops** in order to obtain several properties from image regions.

  This function must also show different figures to illustrate the intermediate results of the process. More precisely:
  - A figure showing the input image in gray scale and its corresponding histogram (see Fig. 2).

Figure 1: Examples of car license plates.

- A figure showing the edges of the image and the original image with the outer lines and the centroids of the letters (see Fig. 3).
- A figure showing the image after a thresholding, the skeletons of the letters and the endpoints/branch-points of the letters (see Fig. 4).

Validation

Any technique learnt during the course can be used to write your solution. Your code must work for, at least, all the images of your correspondent folder. Solutions including code for specific images **will not be accepted**, as well as, **solutions suspected of being copied**. The output of the program must include the correct identification of the license plates, as well as the total number of license plates of each country. For instance, using the three images shown in Fig. 1 as input, the output of the program should be as follows:

```
GB-ABC0178
PL-AAC1817
D-BBB0101
------
Totals:
GB: 1
PL: 1
D: 1
```

Delivering Instructions

- The solution to this final project must be submitted before **May 28, 2023** at **23:59**.
- The final lab will be explained in oral assessment on **May 29, 2023**.
- Each working group must submit a single compressed file (.zip). The file name must be *Gii_FinalProject.zip*, where *ii* is the group number. The compressed file must include:
  - *main.m*, *process_license_plate.m* with the solution, and any other files needed for its execution, properly commented.
  - The corresponding image directory.
  - For each input image, the resulting figures.

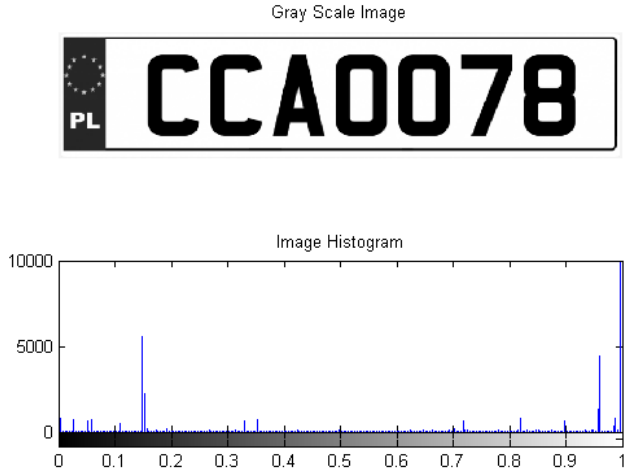

Figure 2: Gray scale image and its corresponding histogram.

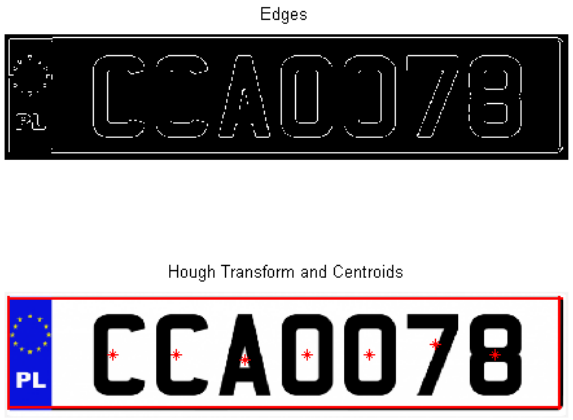

Figure 3: Edges, outer lines and centroids.

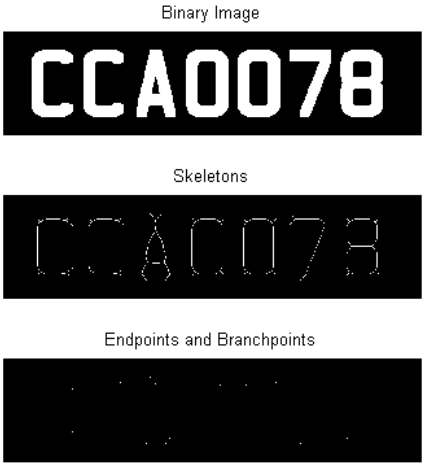

Figure 4: Binary image, skeletons and endpoints/branchpoints of the image.