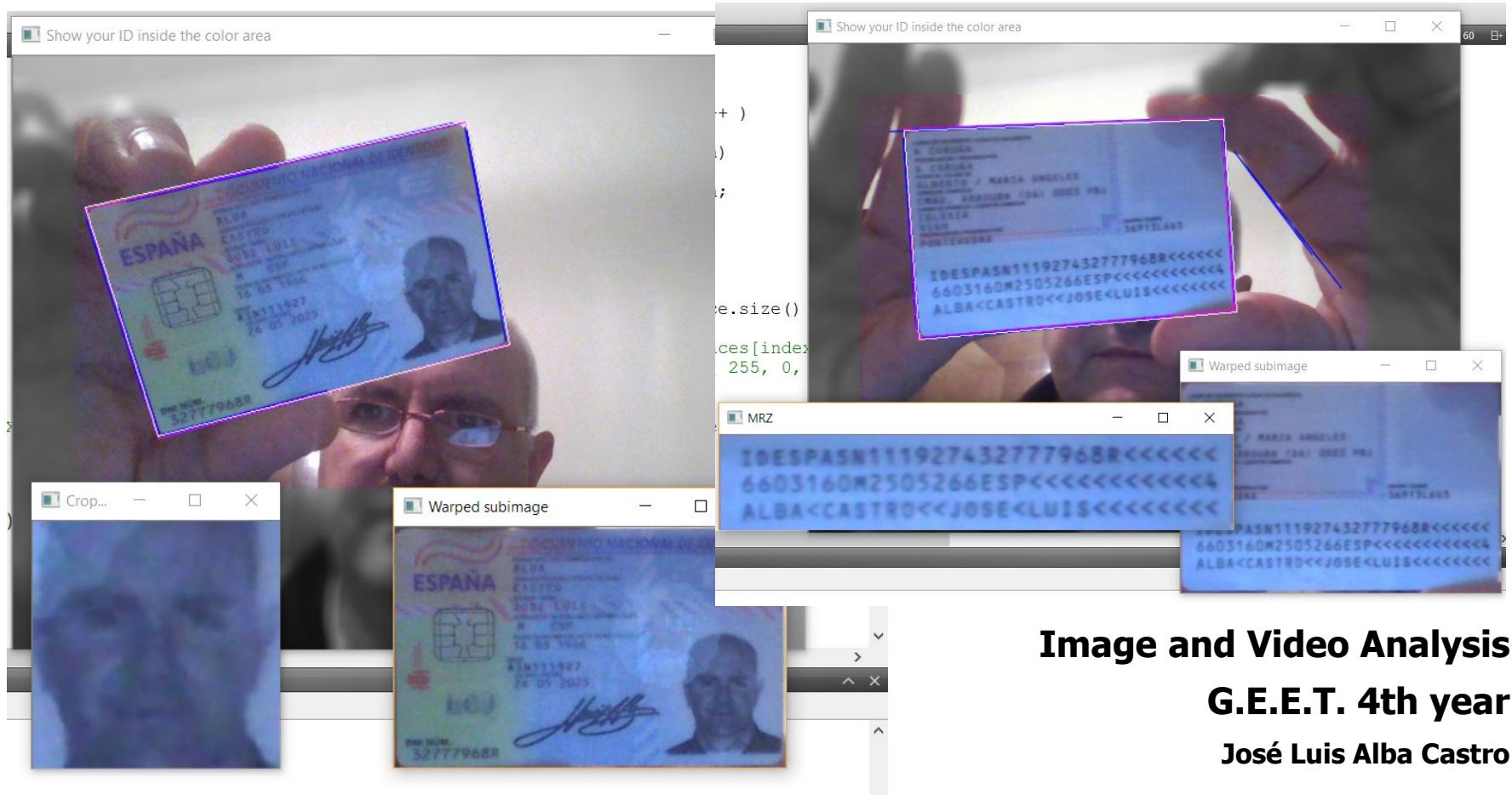


# Subproject 2:

## Detecting, segmenting and recognizing descriptive regions in ID cards

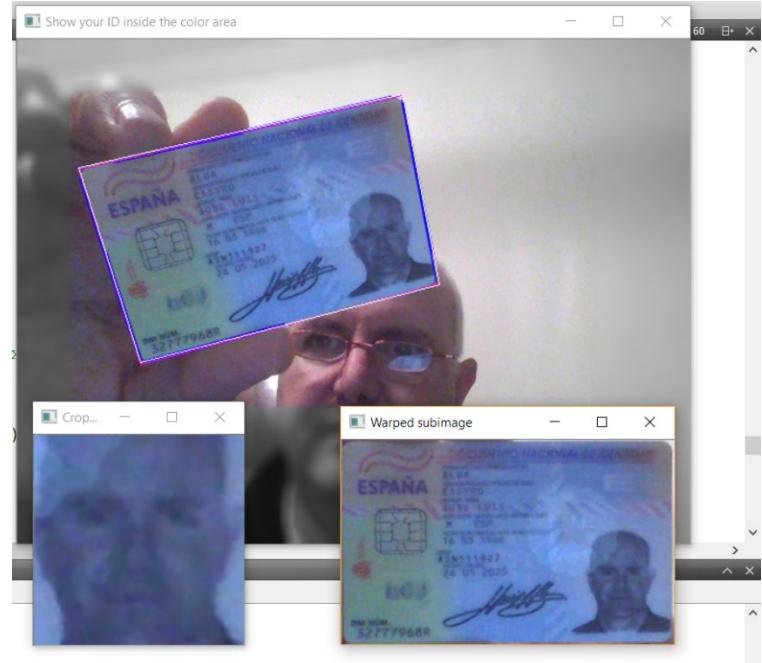


**Image and Video Analysis**  
**G.E.E.T. 4th year**  
**José Luis Alba Castro**

- **General goal:** to analyze a video flow, detect an ID card shown to it, segment the card and analyze the different fields: the MRZ (machine readable zone), Face Photo, signature and personal data

- **Application scenarios:**

- **Static:** the ID card is left on a surface below the camera and the picture is analyzed when no movement is detected
- **Dynamic:** the ID card is held with your hand in front of the camera and the picture is analyzed as soon as a valid picture is taken (large and not blurred)



- Tasks:
  - Detection, segmentation and normalization of the ID card
    - Detecting a good image to analyze (low blurring, low reflection)
    - Perspective correction (OpenCV)
    - Masking the reflection parts to avoid analyzing them
  - Segmentation of the Face Photo and MRZ (front and back of the ID card)
    - Country-dependent → start with ES ID card template analysis → find ID cards from other countries or versions of the same country in internet, class mates or friends.
    - Country-independent → use **morphology filters** to find the MRZ and **face detectors** to find the face photo. Test with several countries or versions of Spanish DNIs and NIEs.
  - Segmentation of the information, name and DNI number fields and signature subimage
    - Template matching, texture descriptors (keypoint detectors)
    - Morphology filtering
    - Pattern recognition, Deep Neural Networks

- **Image processing concepts that can be used:**
  - Color spaces: RGB, HSL, YCbCr, Lab, etc
  - Resizing
  - Block processing: scanning the image, filtering, etc.
  - Hough transform
  - Contour finding
  - Morphological operations
  - Histogram tracking (meanshift)
  - Shape description
  - Perspective correction
  - Haar-based object detection code (OpenCV)
  - KeyPoint detectors (Harris, SURF, SHIFT, ..)
  - Correlation-based matching
  - Foreground-Background segmentation
  - Clustering → k-means
  - Principal Component Analysis
  - Pattern matching
  - DNN
  - Neural Networks, DNN

## • Useful references:

- Book: OpenCV 2 Computer Vision Application Programming Cookbook
  - Chapter 8: Detecting and Matching Interest Points
  - Chapter 10: Processing Video Sequences
- Book: Mastering OpenCV with Practical Computer Vision Projects
  - Chapter 8: Face Recognition using Eigenfaces or Fisherfaces
- Book: Learning OpenCV 4 Computer Vision with Python 3
  - Chapter 5: Detecting and Recognizing Faces
  - Chapter 8: Tracking objects
  - Chapter 10: Introduction to Neural Networks with OpenCV
- Blogs: [learnopencv.com](http://learnopencv.com) (C++, python), [pyimagesearch.com](http://pyimagesearch.com) (python)

# **What should you deliver for this part?**

## **A code project with the next files:**

- python file that runs in Windows platforms
- all the needed libraries to run the code
- The program should receive no arguments because it opens the webcam, but you should allow an optional argument to open a prerecorded videofile for testing
- The executable should analyze the frames from the videoflow and search for an ID document. As soon as a document is found, it should analyze the content and output the data found in it.

## **A PowerPoint presentation explaining the method:**

- you should specify how the program(s) can be used by a third party
- you should highlight the part of the code that has been copied and indicate where you copied it from, even if it comes from a forum discussion
- you should indicate under what conditions the method works and in what cases it doesn't work.
- you should include images to explain how the methods work and some example where it doesn't work and why.

## **What scores can you expect from Subproject 2?**

Independently on the amount of work you do, your Project should fulfill the next requirements in order to pass this part:

- 1) The program should run in a system with Windows11, 64 bit, OpenCV4.X or upper, so make sure to give the correct instructions for me to run it from the source files as if I were a client. I advice you to simulate a second installation in a friend's computer. If you are using a library that is not provided by the distribution we use in class, provide it.
- 2) The executable should run as explained in the previous slide. Any modification of that should be explained and justified.
- 3) The PowerPoint should include the 4 points explained in the previous slide
- 4) The program should work (not halting!) for a video of any length or webcam flow.
- 5) The program should stop only after an ID is found in the video flow and the content is analyzed

The minimum functionality for every score range is the following (the individual score will depend on the contributions in the forum):

1. Up to 5 points: If the program works as explained with any rotation of the ID in a flat surface and **extracts** the image of the face and the signature in upright position
2. Up to 7 points: if the program works as in 1) and also **segments** the sub-images of the main personal data: DNI number, expiration date, name, surname, birthday and MRZ
3. Up to 9 points if the program works as in 2) and also works when you **hold the ID with your hand** in front of the webcam
4. Up to 10 points if the program works as in 2) but the MRZ is read from the image and shown in the console (OCR).