

Computer Graphics and Image Processing

Part 3: Image Processing Assignment Specification

Martin Urschler, PhD



Assignment: License Plate Detection

- Aim: Detect Bounding Box around the license plate in an image of a car



Assignment: License Plate Detection

■ How?

- We will use the image processing techniques that we study in the lecture (and implement in Coderunner)

- Conversion to Greyscale
- Contrast Stretching
- Filtering to detect high contrast regions
- Thresholding for Segmentation
- Morphological operations
- Connected component analysis

- This is a (tested) suggestion, you can also develop your own algorithm.

- But, **you are not allowed to use high level libraries like OpenCV!**



Assignment: License Plate Detection

■ Algorithm

- Read the input image, convert RGB data to greyscale and stretch the values to lie between 0 and 255
- See Coderunner Programming Examples in Week 10



Assignment: License Plate Detection

■ Algorithm

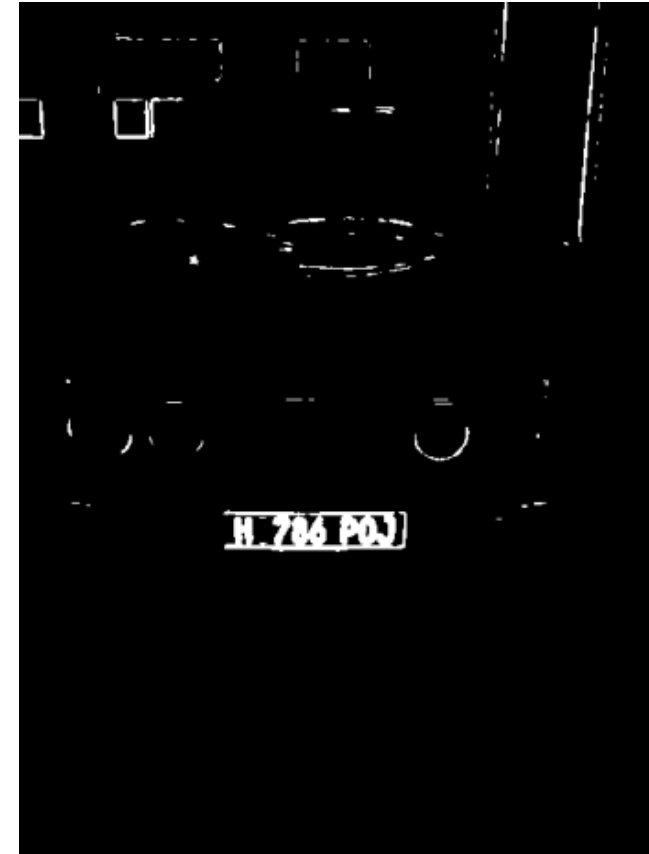
- Find structures with high contrast in the image by computing the standard deviation in the pixel neighbourhood
- See Coderunner Programming Examples in Week 11
- Hint: Use a 5x5 neighbourhood and stretch the result between 0 and 255



Assignment: License Plate Detection

■ Algorithm

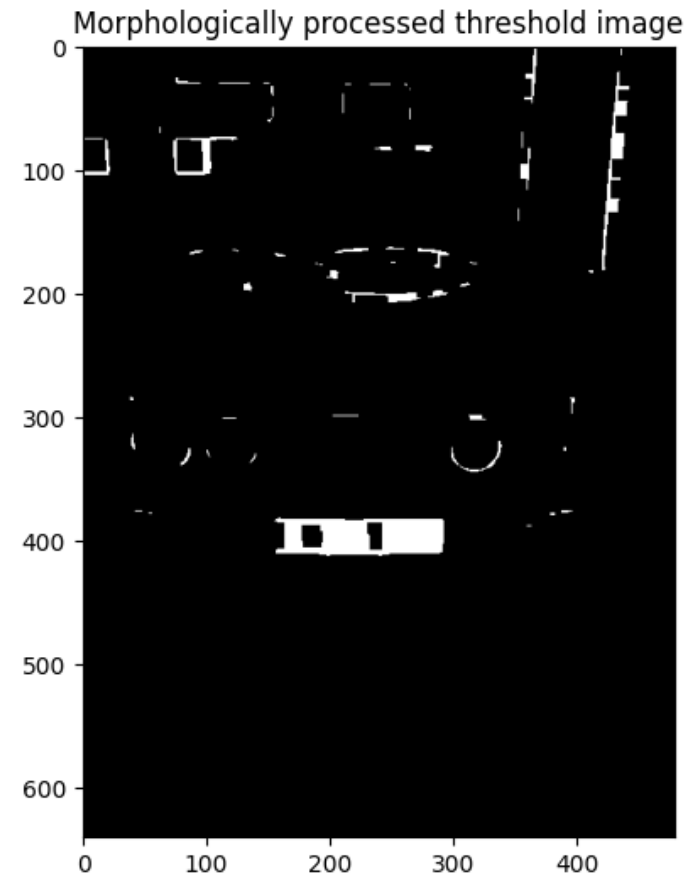
- perform a thresholding operation to get the high contrast regions as a **binary image**
- Hint: a good threshold value is around 150, if you did the contrast stretching in the previous step! (But it is up to you to explore different values, or determine threshold automatically.)



Assignment: License Plate Detection

■ Algorithm

- Perform several 3x3 **dilation** steps followed by several 3x3 **erosion** steps to get a “blob” region for the license plate (morphological closing)
- See Coderunner Programming Examples in Week 12



Assignment: License Plate Detection

■ Algorithm

- perform a **connected component analysis** to find the largest connected object
- See Coderunner Programming Examples in Week 12
- For some of our provided example images, the largest connected component won't be the license plate.
 - So additionally you could also analyze the **aspect ratio** of the generated bounding box and look for the largest connected component within an aspect ratio (i.e. width / height) range between 1.5 and 5

Assignment: License Plate Detection

■ Algorithm

- Extract the final **bounding box** around this region, by looping over the image and looking for the minimum and maximum x and y coordinates of the pixels of the previously determined connected component.



Assignment: License Plate Detection

■ Organization

- Download Python code skeleton from my github (see Assignment description on Canvas for the link)
- Zip your solution and submit it via Canvas to the **Assignment Dropbox**
- Note: Work on assignment at the same time as Coderunner quizzes!
- 10 marks for solving task for this image
- Deadline **Monday June 6, 23:59**
- 5 marks for your **extension**, together with a short reflective report (no report needed for main task!)



Assignment: License Plate Detection

- Important: Use a lab computer to test if your code works on Windows on a different machine (>300 students!)
- Examples for extensions (describe in **reflective report**, submit it as a **pdf** in main folder of your code submission)
 - Find challenging images, show where the algorithm fails and discuss why, attempt to overcome failures
 - Attempt to read the license plate letters and numbers
 - Largest contribution for full marks:
Think about at least own idea for your own individual extension...