



Report about Lab4

Raffaele DI NARDO DI MAIO 1204879

May 11, 2020

1 Goal of the experience

The goal of this lab experience was to detect a road lane and a street signal in an image, highlighting them respectively with red and green colors. Their extraction was done by using the Canny edge detector and Hough transforms and by tuning their parameters.

2 Code Organization

The code of this lab is organized in ten main files (for which I provide also the documentation through doxygen):

- **Lab4.cpp** and its header file **Lab4.h**.
They are associated to *main()* where an instance of *StreetDetector()* is created and its function **detect()** is called.
- **StreetDetector.cpp** and its header file **StreetDetector.h**.
These are used to create the class *StreetDetector*. Its main function is called **detect()** and it creates all the windows for the Canny detector and Hough transforms, creating objects of the classes that implement them. In this function, the main flow of the program is managed. This is organized in two main phases:
 1. **Detection of the street.**
On screen there are two windows at the same time (Canny edge detection and Hough lines transform). By updating the first one, the second one is also affected by these changes. When one key, between 'q', 'ESC' and '\r', is pressed by user, the last image in the Hough lines window is used for the next phase.
 2. **Detection of the signal.**
Using the previously computed image, a new window is created for Hough circles transform.
- **CannyDetector.cpp** and its header file **CannyDetector.h**.
These are used to create the class *CannyDetector*, that performs the Canny edge detection.
- **HoughLinesDetector.cpp** and its header file **HoughLinesDetector.h**.
These are used to create the class *HoughLinesDetector*, that performs Hough transform detection for lines and highlights in red a triangle in the image. This is obtained by the two most significant detected lines.
- **HoughCirclesDetector.cpp** and its header file **HoughCirclesDetector.h**.
These are used to create the class *HoughCirclesDetector*, that performs Hough transform detection for circles and highlights in green the most significant circle in the image, among all detected circles.

3 Command line parameters

The program needs to have, as command line argument, the path of the image (including its name) on which you want to detect the street and the signal.

4 Experimental results

In the following part there are the resulting images used in my workflow:

1. **Parameters evaluation for Canny detection and Hough lines transform.**

Before applying the Canny Detection, to reduce hysteresis thresholds values, I apply a Blur filter of size 9x9. After this smoothing step, I apply the Canny Detector and, after that, the Hough lines transform. The most significant parameters of these detectors are respectively the first threshold and the angle resolution. Through Hough Transform, I found the two most significant lines (e.g. Figure 1). Then evaluating their equations, I compute the three corners, that defines the triangle of the street, and I draw it through `cv::fillPoly()` function.

2. **Parameters evaluation for Hough circles transform.**

The most significant parameters of Hough circles transform are max radius, if we consider a fixed minimum radius of 1, and the second method parameter. By changing second method parameter, in fact, the range of maximum radii, that correctly highlights the signal, changes.

In Table 1, there are the parameters I used to obtain a good detection of the road lane and the signal. In Figure 2 there is the image, obtained by my program, with detected street and signal. The specified parameters are set at object construction.

Canny Detector	<i>threshold₁</i>	[0.0, 157.0]
	<i>threshold₂</i>	200
Hough Lines Detector	<i>distance resolution</i>	1
	<i>angle resolution</i>	$\frac{37 \cdot \pi}{150}$
	accumulator threshold	[0,14]
Hough Circles Detector	<i>minimum distance</i>	69.0
	<i>parameter 1</i>	30.0
	<i>parameter 2</i>	23.0
	<i>minimum radius</i>	1
	<i>maximum radius</i>	[10, 21]

Table 1: Estimated detected parameters that highlight the signal and the street.



Figure 1: Image with highlighted most significant lines.



Figure 2: Image with highlighted street in red and signal in green.