Exercise 6: Edges

In this exercise, you will find the image gradients, apply canny and find circles using Hough circle detection.

- Load an image and compute the gradient magnitude and gradient direction for each pixel in the image. Use your own convolution implementation from report exercise 2 with a Sobel kernel instead of an averaging kernel. Hint 1: Your will need to modify your convolution method to handle Mat objects of type float32 instead of uchar, otherwise you will run into problems with negative output when using the sobel mask. Hint 2: You might want to use OpenCV method cartToPolar to convert between cartesian and polar coordinates (x,y -> magnitude, angle). To visualize gradients which are best expressed as CV_32F images you will need to convert your images to 255 uchar. Use fx. OpenCVs convertScaleAbs.).
- Use OpenCV's canny method to detect lines in an image. Choose appropriate threshold for the hysteresis-thresholding. Try to change the thresholds and observe the change in the computed canny edges.
- Optional: Implement Canny yourself: Smoothing, gradient computation, non-maxima supression, hysteresis-thresholding
- Use OpenCVs Hough line or Hough circle method to detect lines or circles in an image.