# hough2.py - detect lines using hough transform

import cv2 as cv

import numpy as np

import math

I1 = cv.imread('rectangle.jpg')

cv.imshow('Input image',I1)

I1 = cv.GaussianBlur(I1,(5,5),2)

cv.imshow('Filtered image',I1)

I1e = cv.Canny(I1,100,200)

cv.imshow('Image edge detection',I1e)

lines = cv.HoughLines(I1e, 1, np.pi/180, 80, None, 0, 0)

cdst = cv.cvtColor(I1e, cv.COLOR\_GRAY2BGR)

print(len(lines))

# draw the lines

for i in range(0, len(lines)):

rho = lines[i][0][0]

theta = lines[i][0][1]

a = math.cos(theta)

b = math.sin(theta)

x0 = a \* rho

y0 = b \* rho

pt1 = (int(x0 + 1000\*(-b)), int(y0 + 1000\*(a)))

pt2 = (int(x0 - 1000\*(-b)), int(y0 - 1000\*(a)))

cv.line(cdst, pt1, pt2, (0,0,255), 1, cv.LINE\_AA)

cv.imshow("Detected Lines (in red)", cdst)

cv.waitKey(0)

cv.destroyAllWindows()