import numpy as np  
import cv2  
 # we will read the image directly as a grayscale image  
img = cv2.imread('sample 4.png', cv2.IMREAD\_GRAYSCALE) # gaussian filter is not used explicitly as it is already included in the canny function itself  
img = cv2.Canny(img,50,200)  
h,w = img.shape #image dimensions in (no. of rows, no. of columns)  
lines = cv2.HoughLinesP(img,1,np.pi/180,150,maxLineGap = 40) # HoughLines function gives an output of an array with end points.   
no\_of\_lines = len(lines) # this gives no. of edges lines identified and crossed vote limit by hough transformation.  
line\_lengths = np.zeros(no\_of\_lines,dtype = int) # to record the line lengths.  
line\_spacing = []  
line\_lengths   
import math  
for i in range(0,no\_of\_lines):  
 y = lines[i][0][3] - lines[i][0][1]  
 x = lines[i][0][2] - lines[i][0][0]  
 line\_lengths[i] = math.sqrt(y\*\*2 + x\*\*2) #for line lengths  
for i in range(0,no\_of\_lines-1):  
 for j in range(i+1,no\_of\_lines):  
 y = lines[i][0][3] - lines[i][0][1]  
 x = lines[i][0][2] - lines[i][0][0]  
 if (x != 0):  
 a = round(np.degrees(np.arctan(y/x)))  
 else:  
 a = 90  
 y1 = lines[j][0][3] - lines[j][0][1]  
 x1 = lines[j][0][2] - lines[j][0][0]  
 if (x != 0):  
 b = round(np.degrees(np.arctan(y/x)))  
 else:  
 b = 90  
 b = round(np.degrees(np.arctan(y1/x1)))  
 if(a == b):  
 midy = (lines[i][0][3] + lines[i][0][1])/2  
 midx = (lines[i][0][2] + lines[i][0][0])/2  
 midy1 = (lines[j][0][3] + lines[j][0][1])/2  
 midx1 = (lines[j][0][2] + lines[j][0][0])/2  
 string = "the lines "+str(i)+" and "+str(j)+" are parallel"  
 dist = round(math.sqrt((midy1-midy)\*\*2 + (midx1-midx)\*\*2))  
 line\_spacing.append([string,dist])  
 else:  
 string = "the lines "+str(i)+" and "+str(j)+" are not parallel"  
 dist = 0  
 line\_spacing.append([string,dist]) # for line spacing  
   
cv2.namedWindow('img',cv2.WINDOW\_NORMAL)  
cv2.imshow('img', img)  
  
cv2.waitKey(0)  
  
cv2.imwrite('canny image3.jpg', img)

[ Top 5 ]  
C:\Users\Sudhir\anaconda3\lib\selectors.py:314: size=144 KiB, count=3, average=48.0 KiB  
C:\Users\Sudhir\anaconda3\lib\site-packages\IPython\core\compilerop.py:101: size=81.2 KiB, count=1564, average=53 B  
<ipython-input-37-4973ef602813>:8: size=49.2 KiB, count=2, average=24.6 KiB  
C:\Users\Sudhir\anaconda3\lib\site-packages\IPython\core\compilerop.py:133: size=6878 B, count=67, average=103 B  
<ipython-input-36-c3de276a43c2>:22: size=4632 B, count=1, average=4632 B  
C:\Users\Sudhir\anaconda3\lib\json\decoder.py:353: size=4270 B, count=18, average=237 B  
C:\Users\Sudhir\anaconda3\lib\site-packages\jupyter\_client\session.py:911: size=3543 B, count=5, average=709 B  
C:\Users\Sudhir\anaconda3\lib\site-packages\IPython\core\interactiveshell.py:3107: size=3002 B, count=1, average=3002 B  
C:\Users\Sudhir\anaconda3\lib\site-packages\IPython\core\history.py:709: size=3001 B, count=1, average=3001 B  
<string>:1: size=2904 B, count=45, average=65 B

import lizard  
i = lizard.analyze\_file("reference.py")

i.\_\_dict\_\_

{'filename': 'reference.py',  
 'nloc': 42,  
 'function\_list': [],  
 'token\_count': 674}

i.function\_list.\_\_dict\_\_

---------------------------------------------------------------------------  
  
AttributeError Traceback (most recent call last)  
  
<ipython-input-11-4011c5fd433b> in <module>  
----> 1 i.function\_list.\_\_dict\_\_  
  
  
AttributeError: 'list' object has no attribute '\_\_dict\_\_'

print (i.function\_list)

[]