La Transformada de Hough es una técnica popular para detectar cualquier forma, si puedes representar esa forma en forma matemática. Puede detectar la forma incluso si está rota o distorsionada un poco. Veremos cómo funciona para una línea.

Transformación dura en OpenCV Se encapsula en la función OpenCV, cv.HoughLines() . Simplemente devuelve una matriz de valores :math:(rho, theta)`.ρse mide en píxeles yθse mide en radianes. El primer parámetro, la imagen de entrada debe ser una imagen binaria, por lo tanto, aplique un umbral o utilice una detección de bordes astuta antes de aplicar una transformación completa. El segundo y tercer parámetro sonρyθprecisiones respectivamente. El cuarto argumento es el umbral, que significa el voto mínimo que debe obtener para ser considerado una línea. Recuerde, la cantidad de votos depende de la cantidad de puntos en la línea. Por lo tanto, representa la longitud mínima de línea que debe detectarse.

import cv2 as cv  
import numpy as np  
img = cv.imread(cv.samples.findFile('resources/sudoku.png'))  
gray = cv.cvtColor(img,cv.COLOR\_BGR2GRAY)  
edges = cv.Canny(gray,50,150,apertureSize = 3)  
lines = cv.HoughLines(edges,1,np.pi/180,200)  
for line in lines:  
 rho,theta = line[0]  
 a = np.cos(theta)  
 b = np.sin(theta)  
 x0 = a\*rho  
 y0 = b\*rho  
 x1 = int(x0 + 1000\*(-b))  
 y1 = int(y0 + 1000\*(a))  
 x2 = int(x0 - 1000\*(-b))  
 y2 = int(y0 - 1000\*(a))  
 cv.line(img,(x1,y1),(x2,y2),(0,0,255),2)  
cv.imwrite('resources/houghlines3.jpg',img)

True

Transformada probabilística de Hough.

En la transformada gruesa, puedes ver que incluso para una línea con dos argumentos, se necesitan muchos cálculos. La transformada probabilística de Hough es una optimización de la transformada de Hough que vimos. No tiene en cuenta todos los puntos. En cambio, sólo se necesita un subconjunto aleatorio de puntos que sea suficiente para la detección de líneas. Sólo tenemos que disminuir el umbral. Vea la imagen a continuación que compara la transformada de Hough y la transformada probabilística de Hough en el espacio de Hough. (Imagen cortesía: página de inicio de Franck Bettinger )

import cv2 as cv  
import numpy as np  
img = cv.imread(cv.samples.findFile('resources/sudoku.png'))  
gray = cv.cvtColor(img,cv.COLOR\_BGR2GRAY)  
edges = cv.Canny(gray,50,150,apertureSize = 3)  
lines = cv.HoughLinesP(edges,1,np.pi/180,100,minLineLength=100,maxLineGap=10)  
for line in lines:  
 x1,y1,x2,y2 = line[0]  
 cv.line(img,(x1,y1),(x2,y2),(0,255,0),2)  
cv.imwrite('resources/houghlines5.jpg',img)

True