import cv2

from signature\_detect.extractor import Extractor

from signature\_detect.cropper import Cropper

from signature\_detect.loader import Loader

from signature\_detect.judger import Judger

loader=Loader()

extractor=Extractor()

cropper=Cropper()

judger=Judger()

mimg=cv2.imread( "C:\\Users\\Windows 10\\Downloads\\sgn.jpg")

cv2.imshow("original",mimg)

cv2.waitKey(0)

masks=loader.get\_masks("C:\\Users\\Windows 10\\Downloads\\sgn.jpg")

signed= False

for mask in masks:

# cv2.imshow("signature",mask)

# cv2.waitKey(0)

extracted\_mask=extractor.extract(mask)

f\_img=cropper.find\_contours(extracted\_mask)

#print(f\_img)

#cv2.imshow("crop",f\_img)

#cv2.waitKey(0)

results=cropper.run(extracted\_mask)

print(results)

for result in results.values():

print(result)

#roi=

x,y,w,h=result['cropped\_region']

cv2.rectangle(mimg, (x, y), (x + w, y + h), (0, 255, 0), 2)

cv2.imshow("signs",mimg)

cv2.waitKey(0)

crop\_img = mimg[y:y + h, x:x + w]

cv2.imshow("extracted sign", crop\_img)

cv2.waitKey(0)

signed=judger.judge(result['cropped\_mask'])

if signed:

break

if signed:

break

print(signed)