import cv2

import numpy as np

img = cv2.imread('handcut2.jpg',cv2.IMREAD\_COLOR)

img1=img

grayscaled = cv2.cvtColor(img,cv2.COLOR\_BGR2GRAY)

th = cv2.adaptiveThreshold(grayscaled, 255, cv2.ADAPTIVE\_THRESH\_GAUSSIAN\_C, cv2.THRESH\_BINARY\_INV, 115, 1)

#cv2.imshow('original',img)

#cv2.imshow('Adaptive threshold',th)

kernel = np.ones((5,5),np.float32)

opening = cv2.morphologyEx(th, cv2.MORPH\_OPEN, kernel)

#cv2.imshow('Erosion and Dilation',dilate)

#blur = cv2.GaussianBlur(opening,(7,7),0)

#blur= cv2.medianBlur(opening,5)

blur=cv2.bilateralFilter(opening,19,75,75)

dilate=cv2.dilate(blur,kernel,iterations=1)

cv2.namedWindow('Blurring',cv2.WINDOW\_NORMAL)

cv2.resizeWindow('Blurring', 800,600)

edged = cv2.Canny(dilate, 50, 150) #canny edge detection

cv2.namedWindow('Canny',cv2.WINDOW\_NORMAL)

cv2.resizeWindow('Canny', 800,600)

cv2.imshow('Canny',edged)

cv2.imshow('Blurring',dilate)

contours, hierarchy = cv2.findContours(edged,cv2.RETR\_TREE,cv2.CHAIN\_APPROX\_SIMPLE)

# For contour in contours:

# x,y,w,h = cv2.boundingRect(contour)

# if w>50 and h>50:

# cv2.rectangle(img1,(x,y),(x+w,y+h),(0,255,0),2)

# try: hierarchy = hierarchy[0]