**DIGITAL IMAGE PROCESSING**

**ASSIGNMENT#2**

****

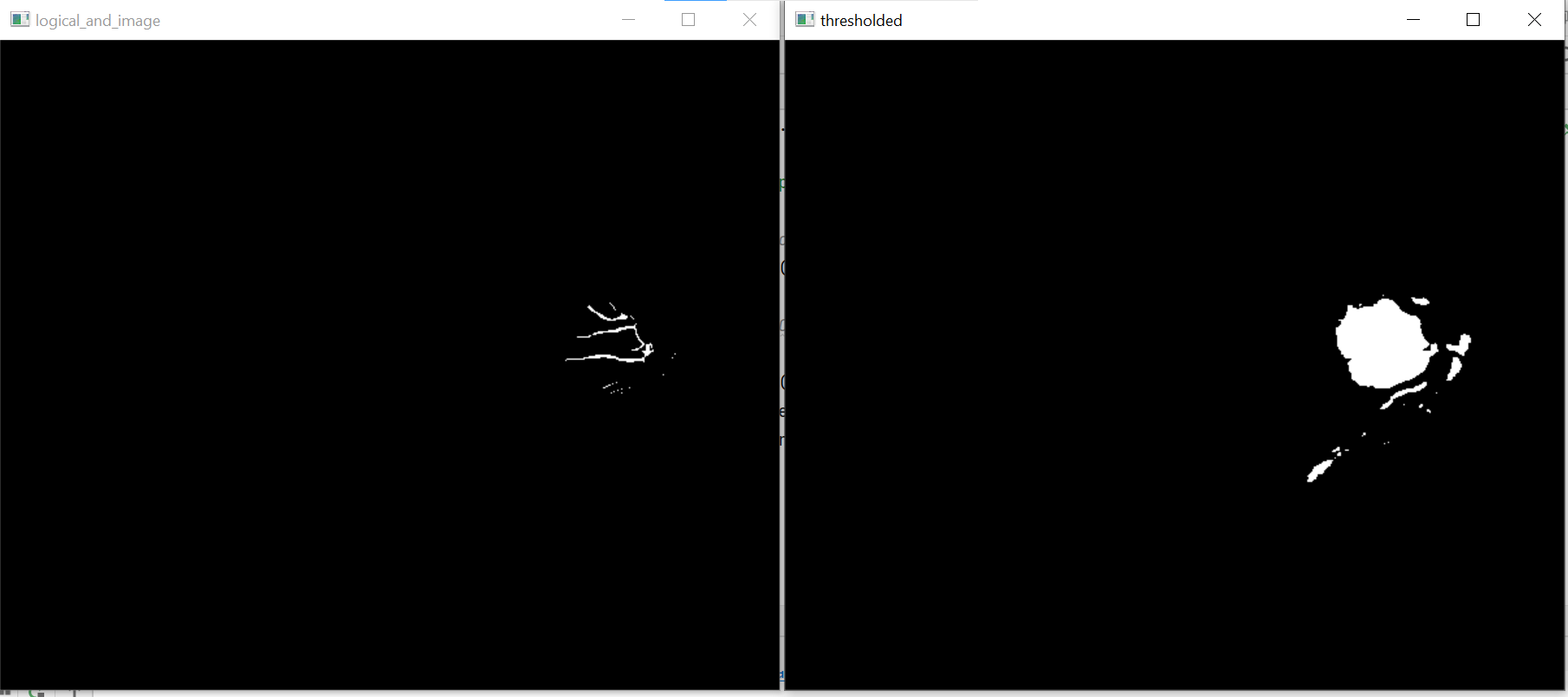
**CODE:**

import numpy as np  
import cv2 as cv  
import csv  
  
def gaussianBlur(img):  
 return cv.GaussianBlur(img, (5, 5), 0) *# smooth*def cca(img):  
 connectivity = 8  
 output = cv.connectedComponentsWithStats(img, connectivity, cv.CV\_32S)  
 return output  
  
def euclidean\_distance(x1, y1, x2, y2):  
 return np.sqrt((x2 - x1) \*\* 2 + (y2 - y1) \*\* 2)  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 image1\_file = '02\_test.tif'  
 actual\_centre = (0, 0)  
 with open('coordinates.csv', 'r') as file:  
 reader = csv.reader(file)  
 for row in reader:  
 if (image1\_file) in row:  
 index = row.index(image1\_file)  
 actual\_centre = (int(row[index+1]), int(row[index+2]))  
  
 img = cv.imread(image1\_file, 0)  
 cv.imshow('original', cv.resize(img, (600, 500), interpolation=cv.INTER\_NEAREST))  
  
 img = gaussianBlur(img)  
 ret, thresh = cv.threshold(img, np.max(img) - 70, 255, cv.THRESH\_BINARY)  
  
 img\_vessel = cv.imread('02 vessel.png', 0) *# load vessel image  
  
 # Apply logical AND operation to focus on the region of interest* logical\_and\_image = cv.bitwise\_and(thresh, img\_vessel) *# AND operation to focus on disc* output = cca(logical\_and\_image) *# CCA on it* cv.imshow('thresholded', cv.resize(thresh, (600, 500), interpolation=cv.INTER\_NEAREST))  
 cv.imshow('vessel\_image', cv.resize(img\_vessel, (600, 500), interpolation=cv.INTER\_NEAREST))  
 cv.imshow('logical\_and\_image', cv.resize(logical\_and\_image, (600, 500), interpolation=cv.INTER\_NEAREST))  
  
 num\_labels = output[0]  
 labels = output[1]  
 stats = output[2]  
 centroids = output[3]  
  
 *# Find the label with the largest area* max\_label = np.argmax(stats[1:, cv.CC\_STAT\_AREA]) + 1 *# Avoid background label at index 0  
  
 # Calculate centroid coordinates* x\_centroid = int(centroids[max\_label][0])  
 y\_centroid = int(centroids[max\_label][1])  
  
 print("Centroid Coordinates (X, Y):", x\_centroid, y\_centroid)  
  
 *# Calculate Euclidean distance* distance = euclidean\_distance(x\_centroid, y\_centroid, actual\_centre[0], actual\_centre[1])  
  
 print("Euclidean Distance:", distance)  
  
 *# Encircle the optical disc* img\_with\_circle = cv.cvtColor(img, cv.COLOR\_GRAY2BGR)  
 center = (x\_centroid, y\_centroid)  
 *# Reduce the radius by half* radius = int(max(stats[max\_label, cv.CC\_STAT\_WIDTH], stats[max\_label, cv.CC\_STAT\_HEIGHT]) / 2)  
 cv.circle(img\_with\_circle, center, radius, (0, 255, 0), 2)  
  
 cv.imshow('optical\_disc\_encircled', cv.resize(img\_with\_circle, (600, 500), interpolation=cv.INTER\_NEAREST))  
 cv.imwrite('G:/6th SEMESTER/DIP/New folder/output/'+image1\_file, img\_with\_circle)  
 cv.imwrite('G:/6th SEMESTER/DIP/New folder/intermediate output/'+image1\_file, logical\_and\_image)  
  
 cv.waitKey(0)  
 cv.destroyAllWindows()

OUTPUT:

NOTE: All images (logical AND image and disc encircled) are added to report in separate folder form in a zip file

For image (02 test):

A close-up of a white circle

Description automatically generated

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Image Name** | | **Given Original Values** | | | **Obtained Values** | | | **ERROR** | |  |
| **x** | | | **y** | **x** | | **y** | | **Euclidean distance** | |  |
| 02\_test.tif | 458 | | 275 | | 446 | | 283 | | 14.42 |  |
| 04\_test.tif | 361 | | 275 | | 365 | | 2763 | | 4.47 |  |
| 06\_test.tif | 461 | | 268 | | 448 | | 272 | | 13.6 |  |
| 08\_test.tif | 485 | | 277 | | 464 | | 282 | | 21.58 |  |
| 10\_left.jpeg | 2439 | | 1697 | | 2383 | | 1702 | | 56.22 |  |
| 10\_right.jpeg | 2985 | | 1562 | | 2931 | | 2093 | | 533.73 |  |
| 10\_test.tif | 468 | | 278 | | 454 | | 281 | | 14.31 |  |
| 12\_test.tif | 82 | | 257 | | 101 | | 260 | | 19.23 |  |
| 13\_left.jpeg | 504 | | 832 | | 567 | | 838 | | 63.2 |  |
| 13\_right.jpeg | 2119 | | 852 | | 2082 | | 860 | | 37.8 |  |
| 13\_test.tif | 486 | | 268 | | 474 | | 275 | | 13.9 |  |
| 14\_test.tif | 479 | | 275 | | 467 | | 287 | | 17 |  |
| 15\_test.tif | 193 | | 282 | | 180 | | 285 | | 13.34 |  |
| 16\_test.tif | 479 | | 258 | | 465 | | 271 | | 19.1 |  |
| 17\_left.jpeg | 1396 | | 1322 | | 1439 | | 1327 | | 43.2 |  |
| 17\_test.tif | 467 | | 267 | | 450 | | 267 | | 17 |  |
| 18\_test.tif | 471 | | 262 | | 473 | | 263 | | 2.23 |  |
| 19\_left.jpeg | 1445 | | 1176 | | 1508 | | 1181 | | 63.2 |  |
| 19\_right.jpeg | 2456 | | 1269 | | 2389 | | 1277 | | 67.4 |  |
| 19\_test.tif | 486 | | 275 | | 469 | | 280 | | 17.7 |  |
| 1ffa92e4-8d87-11e8-9daf-6045cb817f5b..JPG | 541 | | 1530 | | 565 | | 1546 | | 28.8 |  |
| 1ffa92fb-8d87-11e8-9daf-6045cb817f5b..JPG | 2620 | | 1343 | | 1129 | | 1052 | | 1529 |  |
| 1ffa9309-8d87-11e8-9daf-6045cb817f5b..JPG | 609 | | 1167 | | 847 | | 391 | | 811 |  |
| 1ffa9523-8d87-11e8-9daf-6045cb817f5b..JPG | 652 | | 1070 | | 672 | | 1065 | | 20.6 |  |
| 1ffa952f-8d87-11e8-9daf-6045cb817f5b..JPG | 2523 | | 1246 | | 2398 | | 1280 | | 129 |  |
| 1ffa9555-8d87-11e8-9daf-6045cb817f5b..JPG | 512 | | 1501 | | 1957 | | 2161 | | 1588 |  |
| 20\_left.jpeg | 2539 | | 1513 | | 2559 | | 1514 | | 20.02 |  |
| 20\_right.jpeg | 1293 | | 1296 | | 1254 | | 1254 | | 57 |  |
| 20\_test.tif | 482 | | 284 | | 468 | | 276 | | 16.1 |  |
| 21\_left.jpeg | 1724 | | 1589 | | 997 | | 1239 | | 806 |  |
| 21\_right.jpeg | 3036 | | 1556 | | 3011 | | 1505 | | 56.7 |  |
| 21\_training.tif | 77 | | 257 | | 84 | | 254 | | 7.6 |  |
| 22\_left.jpeg | 747 | | 848 | | 847 | | 811 | | 106 |  |
| 22\_right.jpeg | 1821 | | 910 | | 1795 | | 878 | | 41 |  |
| 22\_training.tif | 471 | | 272 | | 392 | | 378 | | 132 |  |
| 23\_left.jpeg | 1417 | | 1127 | | 1438 | | 1181 | | 58 |  |
| 23\_right.jpeg | 2676 | | 1242 | | 2186 | | 1868 | | 794 |  |
| 23\_training.tif | 428 | | 227 | | 440 | | 223 | | 12.6 |  |
| 24\_training.tif | 472 | | 289 | | 459 | | 290 | | 13 |  |
| 25\_left.jpeg | 2635 | | 1455 | | 2597 | | 1403 | | 64 |  |
| 25\_right.jpeg | 791 | | 1846 | | 881 | | 1852 | | 90 |  |
| 25\_training.tif | 464 | | 270 | | 446 | | 274 | | 18 |  |
| 26\_training.tif | 80 | | 245 | | 104 | | 227 | | 30 |  |
| 27\_training.tif | 492 | | 283 | | 473 | | 281 | | 19 |  |