

Домашна задача 3

За сликите дадени во архивата database.zip применете некој од алгоритмите за сегментација на слика со цел да добиете црно-бели слики со јасно издвоени лист и позадина.

Коментирајте ги добиените резултати! Дали избраниот алгоритам точно ги сегментира дадените слики?

Решение:

```
import cv2
import numpy as np
import os

data_dir = "database"
image_files = os.listdir(data_dir)

for file_name in image_files:

    img = cv2.imread(os.path.join(data_dir, file_name))

    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)

    binary = cv2.adaptiveThreshold(gray, 255, cv2.ADAPTIVE_THRESH_GAUSSIAN_C,
cv2.THRESH_BINARY_INV, 51, 20)

    kernel = cv2.getStructuringElement(cv2.MORPH_ELLIPSE, (5, 5))
    opened = cv2.morphologyEx(binary, cv2.MORPH_OPEN, kernel, iterations=1)

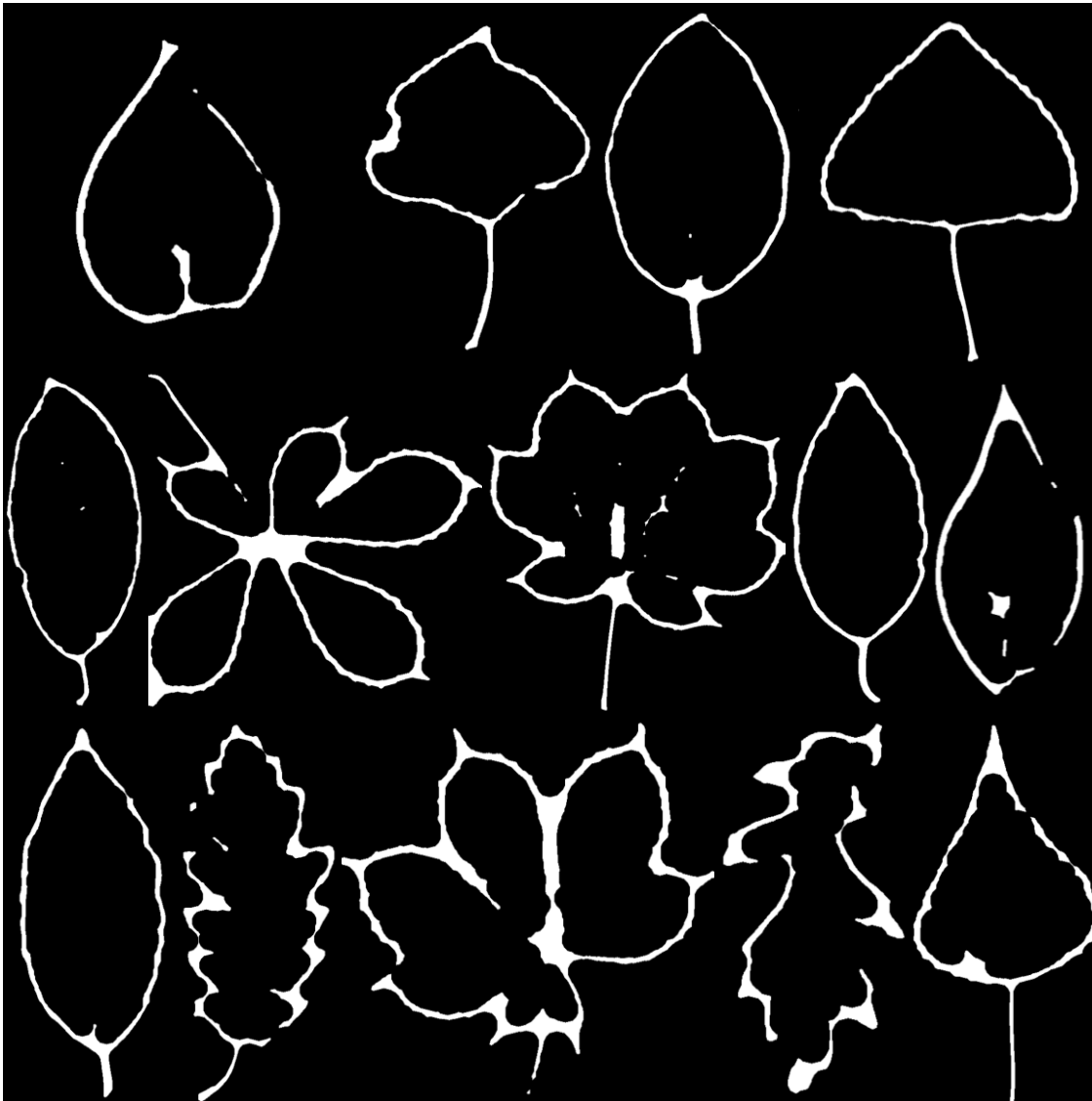
    kernel = cv2.getStructuringElement(cv2.MORPH_ELLIPSE, (15, 15))
    closed = cv2.morphologyEx(opened, cv2.MORPH_CLOSE, kernel, iterations=3)

    contours, hierarchy = cv2.findContours(closed, cv2.RETR_EXTERNAL,
cv2.CHAIN_APPROX_SIMPLE)
    contour = max(contours, key=cv2.contourArea)

    cv2.drawContours(img, [contour], -1, (0, 255, 0), 2)

    cv2.imshow("Segmented Image", closed)
    cv2.waitKey(0)
```

Во овој код, најпрво ги вчитуваме сликите од папката database и правиме for циклус со кој ги изменуваме сите слики од папката и го применуваме кодот врз секоја слика поединечно. Потоа на секоја од сликите применуваме адаптивен праг за да добиеме бинарна слика каде што листот е црн, а позадината бела. Ја користиме функцијата findContours() за да ја добиеме контурата што го дефинира листот и ја цртаме оваа контура на оригиналната слика користејќи ја функцијата drawContours(). На крајот ги прикажуваме резултантните слики. Добиените слики се црно – бели, со листот и позадината јасно одвоени, како и израмнети граници на листот.



Изработил:

Бојан Ристов (211151)