Jashore University Of Science and Technology
Department of Computer Science and Engineering

Course Title: Image Processing Laboratory

Course Code: CSE 4106

Submitted To

Dr. Md. Nasim Adnan

Assistant Professor
Department of Computer Science & Engineering.
Jashore University of Science & Technology.

Submitted By

Md. Abdul Alim (150110)
Dept. Of Computer Science & Engineering
Jashore University of Science &
Technology.

Date of submission: 30-06-21

Problem:

Write a Python program that will isolate the purple zone from the whole image (see the attached file).

Used Technology:

I. **Python 3.8:** Python 3.8 is the latest major release of the Python programming language, and it contains many new features and optimizations.

II. **NumPy:** NumPy is the most important Python package for scientific computing. A powerful N-dimensional array object is included in this package. (Broadcasting) functions that are complex. NumPy is a Python library that adds support for massive, multi-dimensional arrays and matrices, as well as a vast set of high-level mathematical functions to manipulate them.

III. **OpenCV:** OpenCV (Open Source Computer Vision Library) is a machine learning and computer vision software library that is free to use. More than 2500 optimized algorithms are included in the library, which contains a comprehensive mix of both classic and cutting-edge computer vision and machine learning techniques. Most of the time, OpenCV automatically locates memory and allocates memory for output function arguments.

IV. **Notepad++:** Notepad++ is a free (as in "free speech" and also as in "free beer") source code editor and Notepad replacement that supports several languages.

V. **Operating System (Windows 10):** Windows 10 is a major release of the Windows NT operating system developed by Microsoft.

VI. **PyCharm:** PyCharm is a Python Integrated Development Environment (IDE) that includes a wide range of necessary tools for Python developers, all of which are tightly

integrated to create a pleasant environment for productive Python, web, and data science development.

VII. **GitHub:** is a service that hosts Git repositories, but it also has its own set of services. GitHub is a Web-based graphical interface for Git, which is a command line application. For each project, it also provides access control and numerous collaborative capabilities, such as wikis and rudimentary task management tools.

VIII. **Virtual environment:** A virtual environment is a tool that helps to keep dependencies required by different projects separate by creating isolated python virtual environments for them.

Methodology:

➢ Run Pycharm software.
➢ Create a project named "Isolate_the_purple_zone.py".
➢ Import numpy and opencv library.
➢ Read the image using **imread()** function.
➢ Using **fastNlMeansDenoisingColored**() function for remove noise.
➢ Calculate the height weight and channels by using "shape" method.
➢ Use the **cvtColor**() method of the cv2 module which takes the original image and the COLOR_BGR2GRAY attribute as an argument return black and white image.
➢ Find the threshold. In the **threshold()** method, the last argument defines the style of the threshold
➢ Find the image contours. Use the **findContours()** method which takes the image (we passed threshold here) and some attributes.
➢ Sort the contours by using **sorted()** method and using a for loop calculate contours only greater than 100.
➢ Generate the mask using **zeros()** method of NumPy.
➢ Draw contours by using **drawContours()** method of OpenCV.
➢ Apply the **bitwise_and()** method of OpenCV and assign into a new variable as new_img.
➢ create zeros mask 2 pixels larger in each dimension for using in **floodFill()** method of OpenCV.
➢ Do floodfill for white background by using **floodFill()** method of OpenCV.
➢ Display the Oeriginal image by using **imshow()** method of OpenCV.
➢ Display the Isolated shape with background removed image by using **imshow()** method of OpenCV.
➢ Using the **waitKey()** display untill manually stop as the parameter of the method is 0.
➢ Save the file and run the program.

Source Code:

```python
# importing cv2  and numpy
import cv2
import numpy as np

# Reading the image
imge = cv2.imread('./A1.jpg', cv2.IMREAD_COLOR)

# Using fastNlMeansDenoisingColored() function for denoising
img = cv2.fastNlMeansDenoisingColored(imge, None, 20, 10, 7, 21)

# Calculating the height weight and channels
h, w, c = img.shape

# Use the cvtColor() method of the cv2 module which takes the original image and
the COLOR_BGR2GRAY attribute as an argument return black and white image.
gray_img = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)

# Find the threshold. In the threshold() method, the last argument defines the
style of the threshold
_, thresh = cv2.threshold(gray_img, 127, 255, cv2.THRESH_BINARY_INV +
cv2.THRESH_OTSU)

#Find the image contours. Use the findContours() which takes the image (we passed
threshold here) and some attributes.
img_contours = cv2.findContours(thresh, cv2.RETR_TREE, cv2.CHAIN_APPROX_SIMPLE)[-2]

# Sort the contours.
img_contours = sorted(img_contours, key=cv2.contourArea)
for i in img_contours:
    if cv2.contourArea(i) > 100:
        break


# Generate the mask using np.zeros:
mask = np.zeros(img.shape[:2], np.uint8)

# Draw contours:
cv2.drawContours(mask, [i], -1, 255, -1)

# Apply the bitwise_and operator:
new_img = cv2.bitwise_and(img, img, mask=mask)

# create zeros mask 2 pixels larger in each dimension
mask = np.zeros([h + 2, w + 2], np.uint8)

# do floodfill for white background
cv2.floodFill(new_img, mask, (619, 342), (255, 255, 255), (3, 151, 65), (3, 151,
65), flags=8)


# display the Oeriginal image
cv2.imshow("Original Image", img)

# display the Isolated purple shape
cv2.imshow("Isolated shape with background removed", new_img)

#display untill manually stop
cv2.waitKey(0)
```
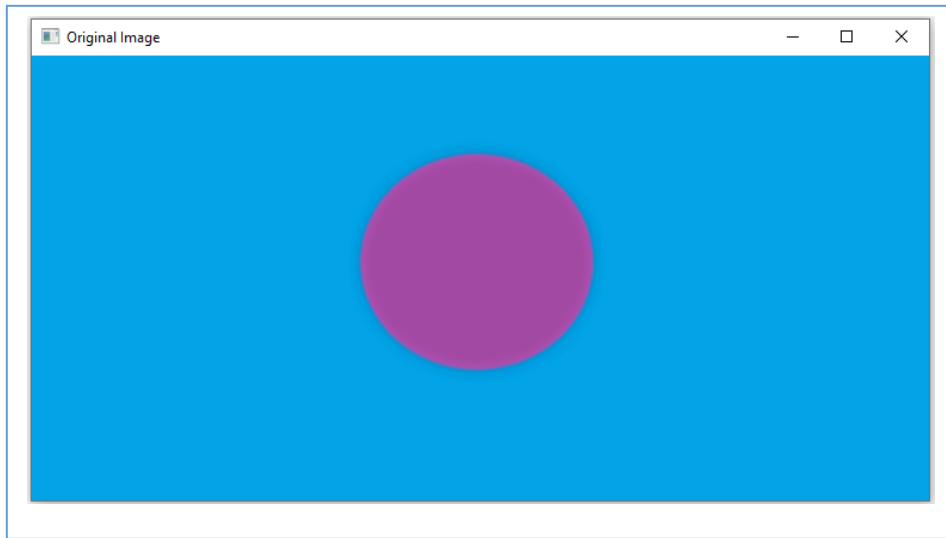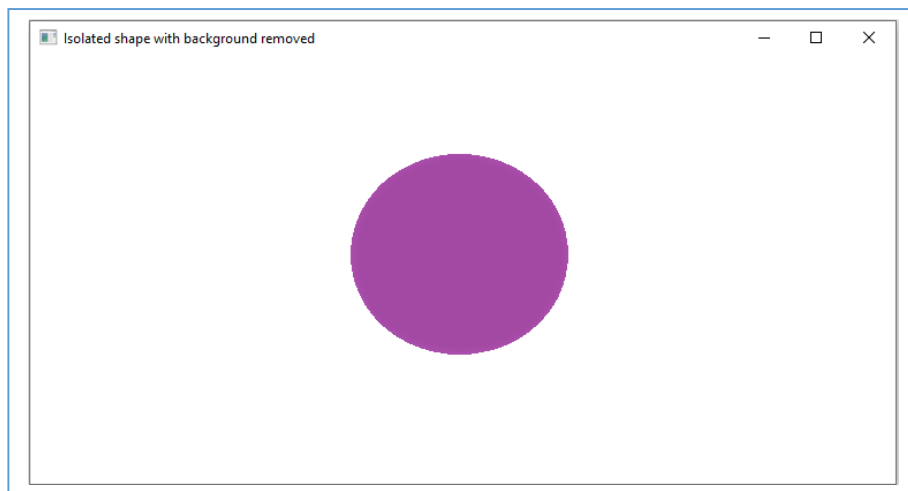
Input and Output:

Fig: Input Image

Fig: Output Image

References:

https://github.com/alimabdul420/image_processing

https://www.python.org/downloads/release/python-380/

https://pypi.org/project/numpy/

https://likegeeks.com/python-image-processing/

https://www.geeksforgeeks.org/opencv-python-tutorial/