

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN
KHOA CÔNG NGHỆ THÔNG TIN

**MÔN HỌC : Toán ứng dụng và thống kê cho
Công nghệ thông tin**



ĐỒ ÁN THỰC HÀNH 2
BÁO CÁO

Tài liệu này mô tả nội dung đồ án môn học cho môn học Toán ứng dụng và thống kê cho Công nghệ thông tin.

Giảng viên hướng dẫn

Phan Thị Phương Uyên

Nguyễn Văn Quang Huy

Sinh viên thực hiện :

Phan Trí Nguyên - 20127578

Thành phố Hồ Chí Minh, ngày 1 tháng 7 năm 2022

MỤC LỤC

1 Tổng quan	3
1.1 Thông tin sinh viên.....	3
1.2 Thông tin đồ án	3
2 Ý tưởng thực hiện và mô tả chi tiết các hàm.....	4
2.1 Ý tưởng thực hiện.....	4
2.2 Mô tả chi tiết các hàm.....	9
3 Demo kết quả chạy chương trình	13
3.1 Thay đổi độ sáng cho ảnh	13
3.2 Thay đổi độ tương phản.....	15
3.3 Lật ảnh (ngang – dọc).....	16
3.4 Chuyển đổi ảnh RGB thành ảnh xám	18
3.5 Chồng hai ảnh cùng kích thước (chỉ làm trên ảnh xám)	19
3.6 Làm mờ ảnh.....	20
3.7 Cắt nội dung ảnh theo khung được áp lên	22
3.8 Cắt nội dung ảnh theo khung được áp lên	24
4 Nhận xét kết quả demo chạy chương trình.....	26
4.1 Đánh giá kết quả	26
5 Tài liệu tham khảo	27

1

Tổng quan

1.1 Thông tin sinh viên

MSSV	Họ và tên	Email	Lớp
20127578	Phan Trí Nguyên	20127578@student.hcmus.edu.vn phantringuyen2002@gmail.com	20CLC05

1.2 Thông tin đồ án

Tên đồ án	Môi trường lập trình
Image Processing	Phần mềm: Jupyter Notebook, Anaconda , Python 3.10, PyCharm Community Edition 2022.1.2. Ngôn ngữ: Python

Mô tả bài toán:

- Để thực hiện việc xử lý ảnh cơ bản ứng với nhu cầu người sử dụng bao gồm:
 - ✓ Thay đổi độ sáng cho ảnh (tăng hoặc giảm).
 - ✓ Thay đổi độ tương phản.
 - ✓ Lật ảnh ngang hoặc dọc.
 - ✓ Chuyển đổi ảnh RGB thành ảnh xám Greyscale.
 - ✓ Chồng hai ảnh cùng kích thước.
 - ✓ Làm mờ ảnh.
 - ✓ Cắt nội dung ảnh theo khung được áp lên, với khung là hình như hình tròn và hai hình ellip chéo nhau.

2

Ý tưởng thực hiện và mô tả chi tiết các hàm

2.1 Ý tưởng thực hiện

2.1.1 Thay đổi độ sáng cho ảnh:

- Cộng ma trận với một số ở kiểu float (để lưu lại các giá trị gần đúng ứng với từng điểm ảnh (kiểu int sẽ khiến điểm ảnh mất đi giá trị thập phân sau khi điểm ảnh cộng với một giá trị brightness)).
- Khi thực hiện việc cộng này thì có khả năng các giá trị trong ma trận được cộng với brightness mà lớn hơn 255 thì sẽ được gán thành 255 và khi một giá trị nhỏ hơn 0 thì sẽ được gán là 0.
- Sau đấy trả lại ma trận kết quả, chuyển thành hình ảnh và thực hiện việc in ra console cho người dùng đồng thời lưu lại file với tên_chức năng tương ứng.png

2.1.2 Tăng độ tương phản:

- Thực hiện giống với việc tăng độ sáng cho ảnh nhưng thay vì cộng ma trận với một số thì ta thực hiện việc nhân ma trận đó với một số contrast do người dùng nhập vào.

2.1.3 Lật ảnh (ngang - dọc):

[\[1\]](#)

- Đối với hàm lật dọc bức ảnh, ta thực hiện lật theo trục Ox (trục ứng với chiều dài của bức ảnh)
- Lưu lại các phần tử từ trục Ox lên vào một ma trận, từ ma trận đấy, gán ngược lại ma trận ảnh ban đầu, bên cạnh đó, các điểm ảnh thứ hai mà RGB và chiều Dimension của ma trận sẽ giữ nguyên.
- Với hàm lật ngang bức ảnh, ta thực hiện như với hàm lật dọc bức ảnh, với điểm khác biệt là lật theo trục Oy (trục ứng với chiều rộng của bức ảnh) và thực hiện việc lưu các phần tử

trung gian giữa ma trận ảnh gốc với ma trận ảo được tạo sẽ là ảnh kết quả sau khi lật với trục Oy.

2.1.4 Chuyển đổi ảnh RGB thành ảnh xám:

- Chuyển đổi hình ảnh màu thành hình ảnh có thang độ màu xám bằng cách lấy giá trị trung bình của các giá trị RGB: giảm các điểm ảnh có chứa điểm màu đỏ, điểm màu xanh lá và đặt điểm ảnh có màu xanh dương vào giữa 2 điểm ảnh này, và tăng các điểm ảnh có chứa, được tính theo trọng số ma trận:
- Từ đó, ta sẽ suy ra được công thức sau:

$$\text{Điểm ảnh xám} = (0.3 * R) + (0.59 * G) + (0.11 * B)$$

[2]

- Chuyển ma trận vừa được xử lý thành ảnh và thực hiện in ra console và lưu ảnh.

2.1.5 Chồng 2 ảnh cùng kích thước (chỉ làm trên ảnh xám):

[3]

- Thực hiện việc đổi kích thước cho bức ảnh thứ 2 sao cho cùng kích thước với bức ảnh thứ nhất
- Chuyển đổi cả 2 bức ảnh thành ảnh trắng đen
- Từ đó thực hiện việc chồng 2 ảnh lên nhau, nhân từng ma trận với một số giao động từ 0.1 đến 0.9 và thực hiện việc cộng hai ma trận để chồng ảnh
- Chuyển đổi ma trận kết quả thành hình ảnh và in ra console.

2.1.6 Làm mờ ảnh:

- Tạo một ma trận gốc Gaussian 3x3 [4], tạo một ma trận phụ với height và width bằng với ma trận ảnh gốc với height từ giá trị thứ nhất (từ giá trị 0) đến giá trị n – 1 và step nhảy là 1.
- Thực hiện việc tính toán điểm ảnh theo dòng, cột; với các điểm ảnh theo khung 3x3 xung quanh 1 điểm ảnh nhất định trong ma trận phụ, thực hiện tính trung bình cộng và tổng hợp tất cả các số là chỉ số thứ 0 của ma trận Gauss hình dạng đề cập đến.
- Chuyển ma trận vừa được xử lý thành ảnh và thực hiện in ra console và lưu ảnh.

2.1.7 Cắt ảnh theo khung được áp vào (đối với khung là hình tròn):

- Tính width, height của ma trận ảnh
- Tính bán kính của ma trận ảnh để thực hiện việc tính khung tròn ngoài cần cắt cho ảnh. Bán kính của hình tròn tính bằng cách lấy một cạnh của ma trận vừa tính được chia đôi, ngoài ra việc cắt ảnh này đối với ảnh có kích thước hình chữ nhật thì chiều rộng của khung ảnh sẽ là cạnh tính cho bán kính cần tìm.
- Bắt đầu thực hiện việc cắt ảnh bằng phương pháp xử lý ma trận ảnh, chuyển các điểm ảnh nằm ngoài khung tròn được tính dựa trên bán kính vừa tìm được thành điểm ảnh đen (điểm ảnh đen có giá trị là 0)
- Để tính được khung tròn ở vị trí nào ảnh, dùng phương trình đường tròn để tính khung tròn và các điểm ảnh nằm ngoài khung tròn này (phương trình đường tròn sẽ lớn hơn R^2) sẽ trở thành các điểm ảnh đen. [\[5\]](#)
- Chuyển đổi ma trận kết quả thành hình ảnh và in ra console.

2.1.8 Thiết kế hàm main để xử lý theo yêu cầu:

- Bước đầu tiên, người dùng nhập vào tên file ảnh (yêu cầu file ảnh nằm cùng với file .ipynb này hoặc người dùng nhập vào đường link đến với ảnh cần xử lý).
- Hiển thị menu hướng dẫn sử dụng cho người dùng.
 - Với hàm thực hiện thay đổi độ sáng cho ảnh: nhập vào độ sáng sẽ thay đổi cho ảnh, thực hiện việc thay đổi độ sáng với hàm được miêu tả trên và in ra console tấm ảnh gốc + ảnh kết quả và sau đấy lưu tên file ảnh kết quả kèm tên chức năng tương ứng.

VD: summer.png => summer.changeBrightness.png
 ảnh gốc ảnh kết quả

- Tương tự với hàm thay đổi độ sáng cho ảnh, hàm thay đổi độ tương phản, nhập vào độ tương phản cần thay đổi cho ảnh và thực hiện lưu ảnh như yêu cầu.

VD: summer.png => summer_changeContrast.png
 ảnh gốc ảnh kết quả

- Với hàm lật ảnh (ngang - dọc), cho người dùng chọn một trong hai phương án, lật ngang ảnh hoặc lật dọc ảnh, sau khi hàm xoay thực hiện thành công, in ảnh gốc và ảnh kết quả ra màn hình console và thực hiện lưu ảnh như yêu cầu.

VD: summer.png => summer_horizonImage.png
ảnh gốc *ảnh kết quả*

summer.png => summer_verticImage.png
ảnh gốc *ảnh kết quả*

- Tương tự hàm lật ảnh ngang – dọc, ảnh sau khi được xử lý chuyển đổi từ ảnh RGB thành ảnh xám Grayscale thực hiện thành công, in ảnh gốc và ảnh kết quả ra màn hình console và thực hiện lưu ảnh như yêu cầu.

VD: summer.png => summer_grayImage.png
ảnh gốc *ảnh kết quả*

- Với hàm chồng 2 ảnh cùng kích thước lên nhau, cho người dùng nhập vào tên file ảnh thứ 2, sau khi hàm xử lý chồng 2 ảnh thực hiện thành công, in ảnh kết quả ra màn hình console và thực hiện lưu ảnh như yêu cầu.

summer.png => summer_stack2Image.png
ảnh gốc *ảnh kết quả*

- Với hàm thực hiện việc cắt nội dung ảnh theo khung được áp lên, với khung là hình như hình tròn, sau khi hàm xử lý cắt nội dung ảnh theo khung thực hiện thành công, in ảnh kết quả ra màn hình console và thực hiện lưu ảnh như yêu cầu.

VD: summer.png => summer_circleFrameCut.png
ảnh gốc *ảnh kết quả*

Reference

- [1] tài liệu này tham khảo tại <https://www.analyticsvidhya.com/blog/2021/05/image-processing-using-numpy-with-practical-implementation-and-code/> dùng để lật ảnh ngang – dọc.
- [2] tài liệu này tham khảo tại https://www.tutorialspoint.com/dip/grayscale_to_rgb_conversion.htm dùng để chuyển đổi ảnh màu RGB sang ảnh trắng đen Grayscale.
- [3] tài liệu này tham khảo tại <https://www.analyticsvidhya.com/blog/2021/05/image-processing-using-numpy-with-practical-implementation-and-code/> dùng để chồng hai tấm ảnh lên nhau.
- [4] tài liệu này tham khảo tại [https://en.wikipedia.org/wiki/Kernel_\(image_processing\)](https://en.wikipedia.org/wiki/Kernel_(image_processing)) dùng để làm mờ ảnh bằng phương pháp Kernel.
- [5] tài liệu này tham khảo tại <https://loigiaihay.com/ly-thuyet-phuong-trinh-duong-tron-c45a5233.html> dùng để tính phương trình đường tròn có tâm $I(a,b)$ và bán kính R .

2.2 Mô tả các hàm

Tên hàm

```
import matplotlib.pyplot as plt
import numpy as np
from PIL import Image
```

Mô tả

Import các thư viện cần thiết trong đó:

- matplotlib.pyplot: hiển thị ảnh.
- numpy: dùng cho tính toán trên ma trận.
- PIL: đọc ảnh.

```
def changeBrightness(image_path, brightness):
```

- Thực hiện việc cộng ma trận với một số nguyên brightness.
- Vì mỗi điểm ảnh có giá trị từ 0 đến 255 nên em sử dụng hàm np.clip(A_list, 0, 255) với A_list là ma trận ảnh gốc, các giá trị trong ma trận được cộng với brightness mà lớn hơn 255 thì sẽ được gán thành 255; các giá trị nhỏ hơn 0 thì sẽ được gán là 0.
- **Input:** tên file ảnh, chỉ số brightness cho ảnh.
- **Output:** ảnh sau khi được tăng (giảm) điểm ảnh với một giá trị brightness người dùng nhập vào.

```
def changeContrast(image_path, contrast):
```

- Thực hiện việc nhân ma trận với một số nguyên contrast.
- Cách giới hạn các điểm ảnh tương tự hàm changeBrightness.
- **Input:** tên file ảnh, chỉ số contrast cho ảnh.
- **Output:** ảnh sau khi được tăng (giảm) điểm ảnh với một giá trị contrast người dùng nhập vào.

```
def rotateImageHorizontal(image):
```

- Thực hiện việc tạo một ma trận tạm và xoay theo trục Ox cho lật dọc.
- Tạo ra một vùng miền cho ứng với chiều dài của ma trận để thực hiện lật dọc.
- Gán các giá trị điểm ảnh theo chiều ngược lại (nghịch đảo ma trận)
- **Input:** mảng 3 chiều ma trận ảnh gốc

	<ul style="list-style-type: none">- Output: ảnh sau khi được lật dọc thành công.
<pre>def rotateImageVertical(image):</pre>	<ul style="list-style-type: none">- Thực hiện việc tạo một ma trận tạm và xoay theo trục Oy cho lật ngang.- Tạo ra một vùng miền cho ứng với chiều dài của ma trận để thực hiện lật ngang.- Gán các giá trị điểm ảnh theo chiều ngược lại (nghịch đảo ma trận)- Input: mảng 3 chiều ma trận ảnh gốc.- Output: ảnh sau khi được lật ngang thành công.
<pre>def to_grayscale(image):</pre>	<ul style="list-style-type: none">- Thực hiện việc chuyển đổi từ ảnh RGB thành ảnh xám Greyscale.- Với công thức các điểm ảnh xám $0.3 \times \text{red}$, $0.59 \times \text{green}$ và $0.11 \times \text{blue}$. Lặp lại số lượng phần tử có trong một mảng với hàm np.tile(). Khi in ảnh ra, sử dụng cmap = 'Greys' để thực hiện in ảnh xám.- Input: mảng 3 chiều ma trận ảnh gốc (RGB image).- Output: mảng 3 chiều ma trận với các điểm ảnh xám tương ứng.
<pre>def RGB2Greyscale(image):</pre>	<ul style="list-style-type: none">- Thực hiện việc chạy trung gian giữa hàm main và hàm to_grayscale(image).- Chuyển đổi ảnh gốc thành ma trận.- Đưa ma trận 3 chiều vào hàm to_grayscale(image) xử lý và chuyển ma trận 3 chiều Greyscale thành ảnh, in ra console và lưu ảnh.- Input: ảnh gốc- Output: ảnh sau khi được Greyscale.

```
def stack2Image(image_path_1, image_path_2):
```

- Thực hiện việc chồng 2 ảnh lên nhau.
- Resize lại ảnh để cả 2 tấm ảnh đều cũng kích thước, để resize lại ảnh cần 2 yếu tố: chiều dài và chiều rộng.
- Chuyển đổi 2 ảnh thành ảnh trắng đen nhờ vào hàm `to_grayscale(image)` trên.
- Nhân từng ma trận với một số giao động từ 0.1 đến 0.9 và cộng hai ma trận với nhau, đặc biệt khi nhân với một số, ứng với một ma trận, nên chọn một số khác với số còn lại để khi các điểm ảnh của ma trận ảnh này sẽ có sự khác biệt so với ma trận ảnh kia.
- Chuyển ma trận về dạng ảnh, in ảnh ra console và lưu ảnh.
- **Input:** địa chỉ ảnh thứ 1, địa chỉ ảnh thứ 2.
- **Output:** ảnh sau khi chồng 2 ảnh lên nhau.

```
def blurImage(image_path):
```

- Thực hiện việc làm mờ bức ảnh.
- Lấy ra chiều dài, chiều rộng của ảnh.
- Tạo một ma trận Gaussian
- Thực hiện việc tính toán điểm ảnh theo dòng, cột; với các điểm ảnh theo khung 3x3 xung quanh 1 điểm ảnh nhất định trong ma trận phụ, thực hiện tính trung bình cộng và tổng hợp tất cả các số là chỉ số thứ 0 của ma trận Gauss hình dạng đề cập đến.
- Chuyển ma trận về dạng ảnh, in ảnh ra console và lưu ảnh.
- **Input:** địa chỉ ảnh nhập vào
- **Output:** ảnh sau khi được làm mờ.

```
def crop_img_by_circle_frame(image_path):
```

- Thực hiện việc cắt nội dung ảnh theo khung được áp lên, với khung là hình như hình tròn.
- Thực hiện việc lấy ra các giá trị chiều dài, chiều rộng và chiều của ma trận ảnh.
- Chọn chiều rộng (cạnh ngắn hơn) của khung hình và chia đôi để tính ra bán kính hình tròn.
- Kiểm tra từng điểm ảnh trong ma trận ảnh với điều kiện điểm ảnh nào nằm ngoài

	<p>phương trình đường tròn với bán kính và tâm đường tròn có hoành độ là chiều dài chia 2 và tung độ là chiều rộng chia 2 bằng cách xét các điểm với phương trình đường tròn mà lớn hơn bán kính hình tròn thì là điểm ảnh nằm ngoài hình tròn – lưu thành điểm ảnh đen (có giá trị là 0).</p> <ul style="list-style-type: none"> - Chuyển ma trận ảnh vừa được xử lý thành ảnh, song thực hiện in ảnh ra console và lưu ảnh về máy. - Input: địa chỉ ảnh nhập vào. - Output: ảnh khi được cắt khung là hình tròn thành công.
<code>class Operate:</code>	- Tạo class Operate để tạo hàm main xử lý.
<code>def out_sentence(self, img_res):</code>	<ul style="list-style-type: none"> - Thực hiện in ra console kết quả file sau khi save thành công. - Input: tên file khi lưu kết quả các hàm xử lý ảnh thành công. - Output: in kết quả tên file để lưu.
<code>def save_changeBrightness(self, image_path, result):</code>	<ul style="list-style-type: none"> - Thực hiện việc lưu ảnh sau khi được hàm xử lý chạy thành công. - Tên file sẽ theo cấu trúc: tên file gốc(không kèm đuôi format) + _tên chức năng + đuôi format (trong bài này em chọn lưu ở định dạng .png) - Input: tên file ảnh gốc, ảnh sau khi được hàm xử lý thành công - Output: Ảnh được lưu vào máy tính.

3

Demo kết quả chạy chương trình

3.1 Thay đổi độ sáng cho ảnh

- ✚ Đầu tiên người dùng nhập vào tên file kèm với đuôi format.
- ✚ Chọn ý 1.
- ✚ Chọn một giá trị brightness mà người dùng muốn tăng (số dương) hoặc giảm (số âm) cho ảnh
- ✚ Hàm xử lý ảnh, in ảnh ra console và lưu ảnh về máy tính.

Enter image file name (include image format): summer.png

Input:

-Image file name: summer.png

- 1.Change image brightness
- 2.Change image contrast
- 3.Flip photos horizontally or vertically
- 4.Convert RGB image to grayscale image
- 5.Stack 2 images of the same size
- 6.Blur image
- 7.Crop image according to the circle frame
- 0.Option to perform all functions from 1 to 7

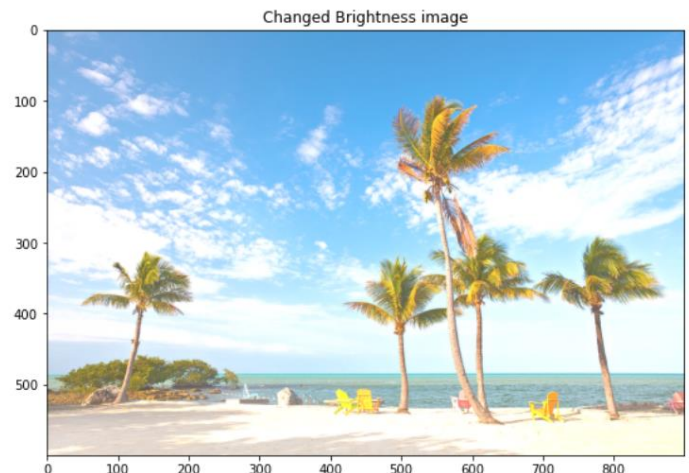
Choose an image processing function (from 1 to 7) or 0 for excuting all six functions: 1

Enter image brighthness: 58

Output:

Image saved successfully!

-Image's name saved as summer_changeBrightness.png



🌈 Với giá trị brightness là số âm (giảm điểm ảnh)

Enter image file name (include image format): summer.png

Input:

-Image file name: summer.png

- 1.Change image brightness
- 2.Change image contrast
- 3.Flip photos horizontally or vertically
- 4.Convert RGB image to grayscale image
- 5.Stack 2 images of the same size
- 6.Blur image
- 7.Crop image according to the circle frame
- 0.Option to perform all functions from 1 to 7

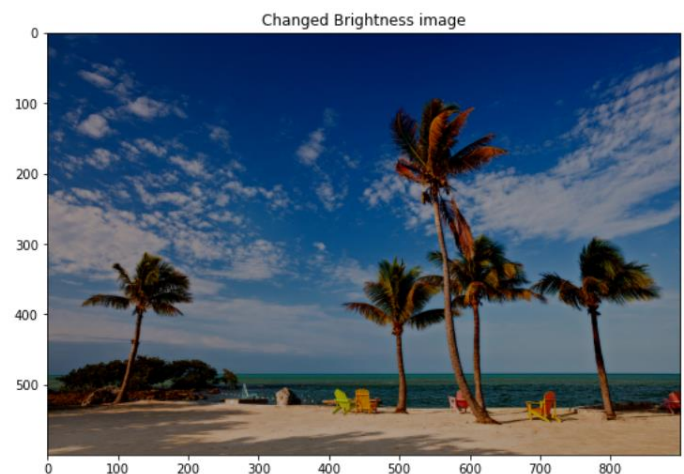
Choose an image processing function (from 1 to 7) or 0 for excuting all six functions: 1

Enter image brighthness: -79

Output:

Image saved successfully!

-Image's name saved as summer_changeBrightness.png



3.2 Thay đổi độ tương phản cho ảnh

- ✚ Đầu tiên người dùng nhập vào tên file kèm với đuôi format.
- ✚ Chọn ý thứ 2.
- ✚ Nhập vào một giá trị contrast để thay đổi độ tương phản ảnh: tăng (contrast > 0) và giảm (contrast < 0).
- ✚ Hàm xử lý, in ảnh ra console và lưu ảnh về máy tính.

Enter image file name (include image format): summer.png

Input:

-Image file name: summer.png

- 1.Change image brightness
- 2.Change image contrast
- 3.Flip photos horizontally or vertically
- 4.Convert RGB image to grayscale image
- 5.Stack 2 images of the same size
- 6.Blur image
- 7.Crop image according to the circle frame
- 0.Option to perform all functions from 1 to 7

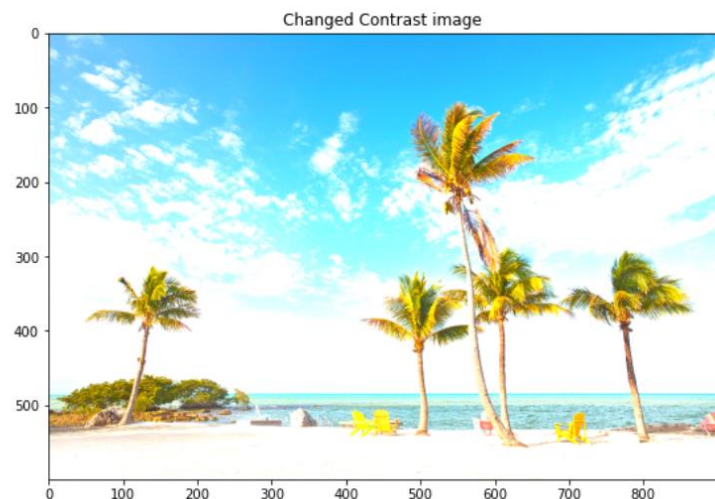
Choose an image processing function (from 1 to 7) or 0 for excuting all six functions: 2

Enter image contrast: 1.7284

Output:

Image saved successfully!

-Image's name saved as summer_changeContrast.png



3.3 Lật ảnh ngang – dọc

- ✚ Đầu tiên người dùng nhập vào tên file kèm với đuôi format.
- ✚ Chọn ý số 3.
- ✚ Chọn ý số 1 nếu người dùng muốn lật ảnh theo chiều dọc.
- ✚ Hàm xử lí ảnh, in ảnh ra console và lưu ảnh về máy tính.

Enter image file name (include image format): summer.png

Input:

-Image file name: summer.png

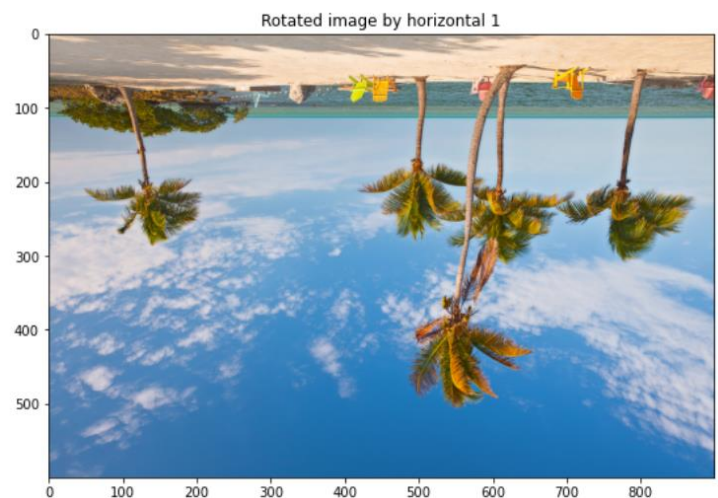
- 1.Change image brightness
- 2.Change image contrast
- 3.Flip photos horizontally or vertically
- 4.Convert RGB image to grayscale image
- 5.Stack 2 images of the same size
- 6.Blur image
- 7.Crop image according to the circle frame
- 0.Option to perform all functions from 1 to 7

Choose an image processing function (from 1 to 7) or 0 for excuting all six functions: 3
Flip the image horizontally (1) or vertically (2): 1

Output:

Image saved successfully!

-Image's name saved as summer_horizonImage.png



- ✚ Đầu tiên người dùng nhập vào tên file kèm với đuôi format.
- ✚ Chọn ý số 3.
- ✚ Chọn ý số 2 nếu người dùng muốn lật ảnh theo chiều ngang.
- ✚ Hàm xử lý ảnh, in ảnh ra console và lưu ảnh về máy tính.

Enter image file name (include image format): summer.png

Input:

-Image file name: summer.png

- 1.Change image brightness
- 2.Change image contrast
- 3.Flip photos horizontally or vertically
- 4.Convert RGB image to grayscale image
- 5.Stack 2 images of the same size
- 6.Blur image
- 7.Crop image according to the circle frame
- 0.Option to perform all functions from 1 to 7

Choose an image processing function (from 1 to 7) or 0 for excuting all six functions: 3
Flip the image horizontally (1) or vertically (2): 2

Output:

Image saved successfully!

-Image's name saved as summer_verticImage.png



3.4 Chuyển từ ảnh RGB thành ảnh xám Greyscale

- Đầu tiên người dùng nhập vào tên file kèm với đuôi format.
- Chọn ý số 4.
- Hàm xử lý ảnh, in ảnh ra console và lưu ảnh về máy tính.

Enter image file name (include image format): summer.png

Input:

-Image file name: summer.png

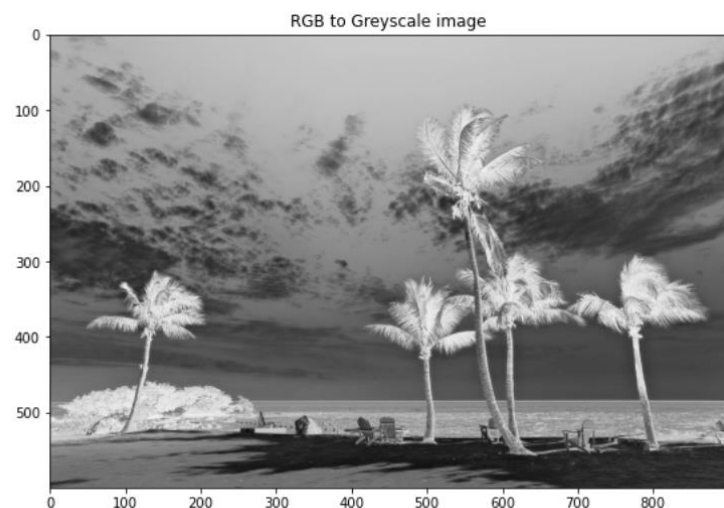
- 1.Change image brightness
- 2.Change image contrast
- 3.Flip photos horizontally or vertically
- 4.Convert RGB image to grayscale image
- 5.Stack 2 images of the same size
- 6.Blur image
- 7.Crop image according to the circle frame
- 0.Option to perform all functions from 1 to 7

Choose an image processing function (from 1 to 7) or 0 for excuting all six functions: 4

Output:

Image saved successfully!

-Image's name saved as summer_grayImage.png



3.5 Chồng hai ảnh cùng kích thước:

- Đầu tiên người dùng nhập vào tên file kèm với đuôi format.
- Chọn ý số 5.
- Hàm xử lý ảnh, in ảnh ra console và lưu ảnh về máy tính.

Enter image file name (include image format): summer.png

Input:

-Image file name: summer.png

- 1.Change image brightness
- 2.Change image contrast
- 3.Flip photos horizontally or vertically
- 4.Convert RGB image to grayscale image
- 5.Stack 2 images of the same size
- 6.Blur image
- 7.Crop image according to the circle frame
- 0.Option to perform all functions from 1 to 7

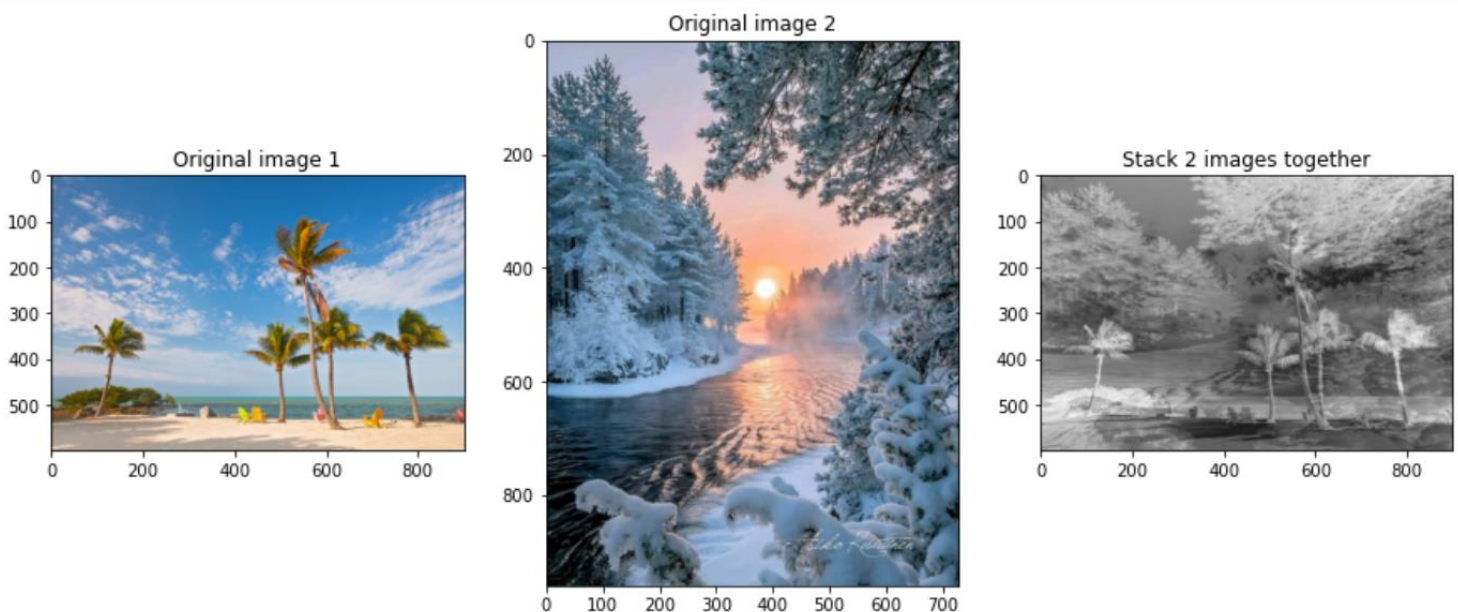
Choose an image processing function (from 1 to 7) or 0 for excuting all six functions: 5

Enter the second image file name (include image format): winter.jpg

Output:

Image saved successfully!

-Image's name saved as summer_stack2Image.png



3.6 Làm mờ ảnh

- Đầu tiên người dùng nhập vào tên file kèm với đuôi format.
- Chọn ý số 6.
- Hàm xử lý ảnh, in ảnh ra console và lưu ảnh về máy tính.

Enter image file name (include image format): summer.png

Input:

-Image file name: summer.png

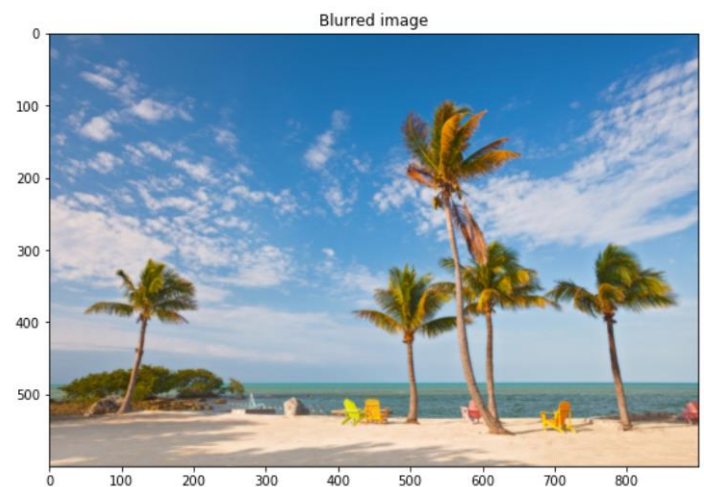
- 1.Change image brightness
- 2.Change image contrast
- 3.Flip photos horizontally or vertically
- 4.Convert RGB image to grayscale image
- 5.Stack 2 images of the same size
- 6.Blur image
- 7.Crop image according to the circle frame
- 0.Option to perform all functions from 1 to 7

Choose an image processing function (from 1 to 7) or 0 for excuting all six functions: 6

Output:

Image saved successfully!

-Image's name saved as summer_blurImage.png





3.7 Cắt nội dung ảnh theo khung được áp lên, với khung là hình tròn

- 🚦 Đầu tiên người dùng nhập vào tên file kèm với đuôi format.
- 🚦 Chọn ý số 7.
- 🚦 Hàm xử lý ảnh, in ảnh ra console và lưu ảnh về máy tính.

Enter image file name (include image format): summer.png

Input:

-Image file name: summer.png

- 1.Change image brightness
- 2.Change image contrast
- 3.Flip photos horizontally or vertically
- 4.Convert RGB image to grayscale image
- 5.Stack 2 images of the same size
- 6.Blur image
- 7.Crop image according to the circle frame
- 0.Option to perform all functions from 1 to 7

Choose an image processing function (from 1 to 7) or 0 for excuting all six functions: 7

Output:

Image saved successfully!

-Image's name saved as summer_circleFrameCut.png





3.8 Hàm thực hiện tất cả các hàm xử lý ảnh trên

- ✚ Đầu tiên người dùng nhập vào tên file kèm với đuôi format.
- ✚ Chọn ý số 0.
- ✚ Nhập vào giá trị brightness để thay đổi độ sáng của ảnh.
- ✚ Nhập vào giá trị contrast để thay đổi độ tương phản của ảnh.
- ✚ Nhập vào tên ảnh thứ hai để chồng ảnh thứ nhất và ảnh thứ hai.
- ✚ Hàm xử lý ảnh, in tất cả ảnh ra console và lưu ảnh về máy tính.

Enter image file name (include image format): summer.png

Input:

-Image file name: summer.png

- 1.Change image brightness
- 2.Change image contrast
- 3.Flip photos horizontally or vertically
- 4.Convert RGB image to grayscale image
- 5.Stack 2 images of the same size
- 6.Blur image
- 7.Crop image according to the circle frame
- 0.Option to perform all functions from 1 to 7

Choose an image processing function (from 1 to 7) or 0 for excuting all six functions: 0

1. In ra ảnh được thay đổi độ sáng

Enter image brigthness: 71

Output:

Image saved successfully!

-Image's name saved as summer_changeBrightness.png

2. In ra ảnh được thay đổi độ tương phản

Enter image contrast: 1.7329

Output:

Image saved successfully!

-Image's name saved as summer_changeContrast.png

- 3: Lật ảnh (ngang - dọc)

Output:

Image saved successfully!

-Image's name saved as summer_horizonImage.png

Output:

Image saved successfully!

-Image's name saved as summer_verticImage.png

- 4: Chuyển đổi ảnh RGB thành ảnh xám

Output:

Image saved successfully!

-Image's name saved as summer_grayImage.png

5: Chồng 2 ảnh cùng kích thước (1 điểm): chỉ làm trên ảnh xám

Enter the second image file name (include image format): autumn.jpg

Output:

Image saved successfully!

-Image's name saved as summer_stack2Image.png

6: Làm mờ ảnh

Output:

Image saved successfully!

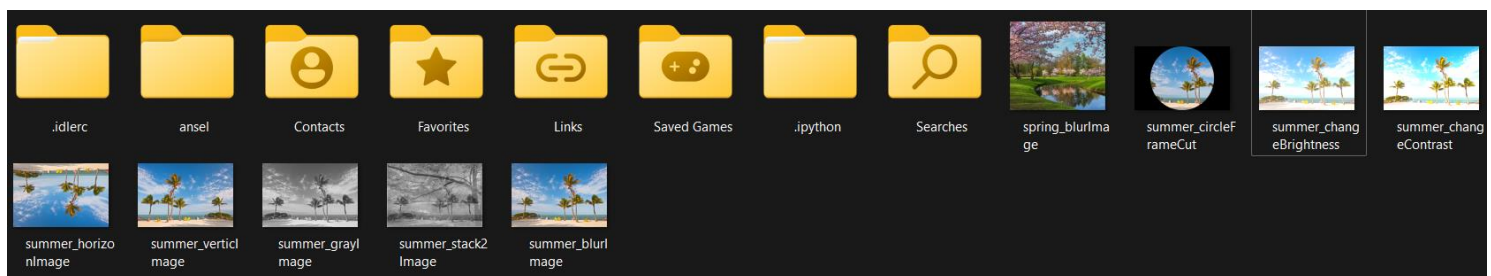
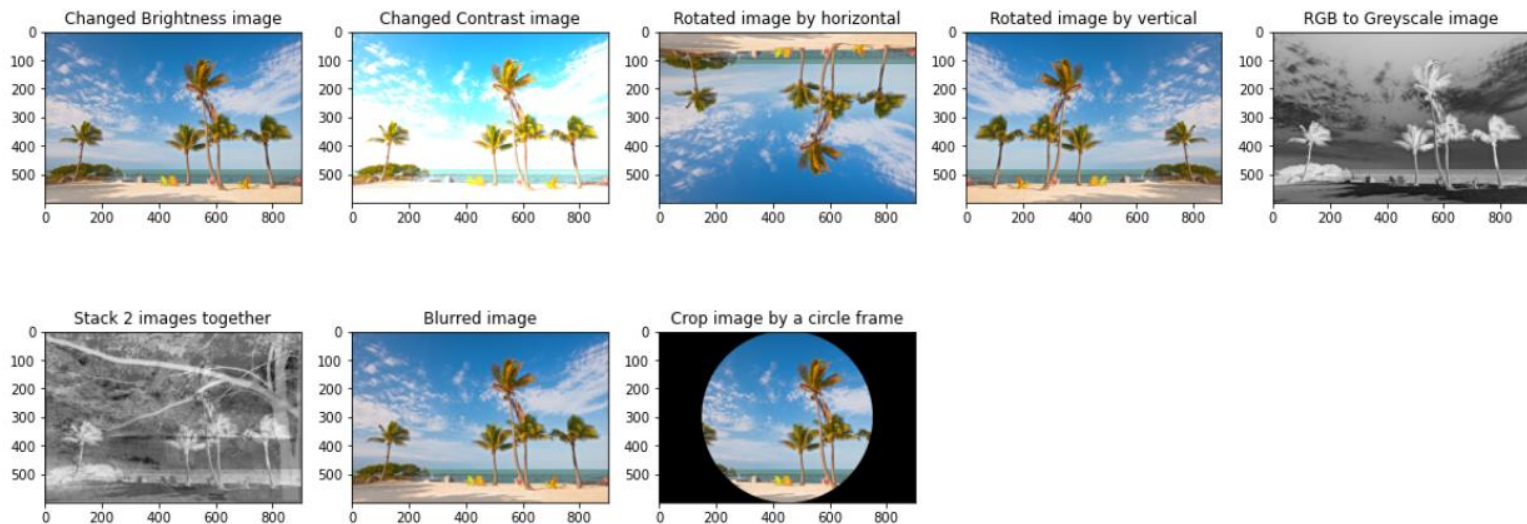
-Image's name saved as summer_blurImage.png

7: Cắt nội dung ảnh theo khung được áp lên, với khung là hình như hình tròn

Output:

Image saved successfully!

-Image's name saved as summer_circleFrameCut.png



4

Nhận xét kết quả demo chạy chương trình

4.1 Đánh giá kết quả

STT	Chức năng	Mức độ hoàn thành
1	Thay đổi độ sáng cho ảnh (1đ)	100%
2	Thay đổi độ tương phản (1đ)	100%
3	Lật ảnh (ngang - dọc) (2đ)	100%
4	Chuyển đổi ảnh RGB thành ảnh xám (2đ)	100%
5	Chồng 2 ảnh cùng kích thước (1đ)	100%
6	Làm mờ ảnh (2đ)	100%
7	Viết hàm main xử lý (1đ)	100%
8	Cắt nội dung ảnh theo khung được áp lên: khung tròn (0.5đ)	100%
9	Cắt nội dung ảnh theo khung được áp lên: khung 2 hình elip (0.5đ)	0%

5

Tài liệu tham khảo

https://github.com/maponti/imageprocessing_course_icmc/blob/master/06a_colour_processing.ipynb

<https://github.com/ijmbarr/image-processing-with-numpy/blob/master/image-processing-with-numpy.ipynb>

https://github.com/marwagaser/ImageProcessing/blob/master/Assign1_Problem1.ipynb

https://github.com/maponti/imageprocessing_course_icmc

<https://github.com/kieuconghau/image-processing/blob/master/Source/18127259.ipynb>

<https://stackoverflow.com/questions/3584805/in-matplotlib-what-does-the-argument-mean-in-fig-add-subplot111>

<https://www.analyticsvidhya.com/blog/2021/05/image-processing-using-numpy-with-practical-implementation-and-code/>

[https://note.nkmk.me/en/python-numpy-flip-flipud-fliplr/#:~:text=Flip%20image%20vertically%20and%20horizontally,-Image%20files%20can&text=You%20can%20flip%20the%20image,flipr\(\)%20](https://note.nkmk.me/en/python-numpy-flip-flipud-fliplr/#:~:text=Flip%20image%20vertically%20and%20horizontally,-Image%20files%20can&text=You%20can%20flip%20the%20image,flipr()%20)

https://www.tutorialspoint.com/dip/grayscale_to_rgb_conversion.htm