**TOÁN ỨNG DỤNG VÀ THỐNG KÊ CHO CÔNG NGHỆ THÔNG TIN**

ĐỒ ÁN MÔN HỌC #2

REPORT

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

Họ tên sinh viên: Nguyễn Văn Đạt

MSSV: 20127132

C:\Users\tdqua_000\Dropbox\SS-Slides\DeCuong-CDIO\Template CDIO v4.2\Templates\Hinh anh\LogoTruong.png

Khoa Công nghệ Thông tin

Đại học Khoa học Tự nhiên TP HCM

Tháng Thg7-22

**MỤC LỤC**

[1 Tổng quan 3](#_Toc107339764)

[Thông tin nhóm 3](#_Toc107339765)

[Thông tin đồ án 3](#_Toc107339766)

[2 Nội dung đồ án 4](#_Toc107339767)

[2.1 Giới thiệu 4](#_Toc107339768)

[2.2 Ý tưởng thực hiện 5](#_Toc107339769)

[2.3 Mô tả hàm 6](#_Toc107339770)

[2.4 Kết quả 7](#_Toc107339771)

[3 Chức năng 9](#_Toc107339772)

[4 Mức độ hoàn thành 10](#_Toc107339773)

[5 Tài liệu tham khảo 11](#_Toc107339774)

# Tổng quan

### Thông tin nhóm

|  |  |  |
| --- | --- | --- |
| MSSV | Họ tên | Email |
| 20127132 | Nguyễn Văn Đạt | 20127132@student.hcmus.edu.vn |

### Thông tin đồ án

|  |  |
| --- | --- |
| Tên đồ án: Image Processing | |
| Công cụ hướng dẫn | Visual Studio Code  Jupyter Notebook |

# Nội dung đồ án

## Giới thiệu

Trong đồ án này, bạn được yêu cầu thực hiện các chức năng xử lý ảnh cơ bản sau:

1. Thay đổi độ sáng cho ảnh (1 điểm)

2. Thay đổi độ tương phản (1 điểm)

3. Lật ảnh (ngang - dọc) (2 điểm)

4. Chuyển đổi ảnh RGB thành ảnh xám (2 điểm)

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

6. Làm mờ ảnh (2 điểm)

7. Viết hàm main xử lý (1 điểm) với các yêu cầu sau:

- Cho phép người dùng nhập vào tên tập tin ảnh mỗi khi hàm main được thực thi.

- Cho phép người dùng lựa chọn chức năng xử lý ảnh (từ 1 đến 6, đối với chức năng 4 cho phép lựa chọn giữa lật ngang hoặc lật dọc). Lựa chọn 0 cho phép thực hiện tất cả chức năng với tên file đầu ra tương ứng với từng chức năng. Ví dụ:

- Đầu vào: `cat.png`

- Chức năng: Làm mờ

- Đầu ra: `cat\_blur.png`

Nâng cao:

Sinh viên thực cắt nội dung ảnh theo khung được áp lên, với khung là hình như hình tròn và 2 hình ellip chéo nhau như các ảnh sau:

- Khung là hình tròn

- Khung là 2 hình ellip chéo nhau

## Ý tưởng thực hiện

Tạo tất cả các hàm thỏa yêu cầu đồ án:

1. Thay đổi độ sáng cho ảnh: Cộng hình ảnh ban đầu với một số tự cho

2. Thay đổi độ tương phản: Dùng công thức tìm hệ số rồi tính theo công thức được gợi ý trên các công cụ tìm kiếm

3. Lật ảnh (ngang - dọc): Thay đổi vị trí các cột, hàng

4. Chuyển đổi ảnh RGB thành ảnh xám: Nhân R,G,B với hệ số cho trước

5. Chồng 2 ảnh cùng kích thướ): Cộng 2 numpy array lại với nhau

6. Làm mờ ảnh: Dùng thuật toán Box Blur

7. Hàm main: Thực hiện tất cả chức năng mà đồ án đã thực hiện thỏa yêu cầu

8. Khung là hình tròn: Dùng phương trình đường tròn và giới hạn lại các điểm thỏa bất phương trình.

9. Khung là 2 hình ellip chéo nhau: : Dùng phương trình elip và giới hạn lại các điểm thỏa bất phương trình.

## Mô tả hàm

1. Thay đổi độ sáng cho ảnh:

def changeBrightnessImage(img, scale\_brightness)

Cộng với số scale\_brightness (có thể âm hoặc dương), rồi nếu vượt quá 255 thì trả về 255, ngược lại nếu nhỏ hơn 0 thì trả về 0.

2. Thay đổi độ tương phản:

def changeContrastImage(img, contrast)

Tính hệ số factor bằng công thức được tham khảo trên các trang tìm kiếm. Sau đó dùng công thức factor \* (image - 128) + 128 để thay rồi RGB của bức ảnh. Nếu vượt quá 255 thì trả về 255, ngược lại nếu nhỏ hơn 0 thì trả về 0.

3. Lật ảnh (ngang - dọc):

def reverseImageHorizontal(img)

def reverseImageVertical(img)

Đảo ngược hàng (hoặc cột) bằng ::-1 được tích hợp trong numpy array của python.

4. Chuyển đổi ảnh RGB thành ảnh xám:

def changeGrayscaleImage(img):

Dùng công thức R \* 0.3 + G \* 0.59 + B \* 0.11 để gán vào R,G,B hiện tại.

5. Chồng 2 ảnh cùng kích thước):

def mergeImages(img1, img2)

Chuyển 2 hình ảnh hiện tại về hình ảnh xám (để tránh vượt quá 255) rồi cộng với nhau.

6. Làm mờ ảnh:

def blurImage(img, img\_height, img\_width)

Dùng công thức Box Blur sau đó gán kết quả cho bức ảnh

7. Hàm main:

def main()

Thực hiện các chức năng đã hoàn thành

8. Khung là hình tròn:

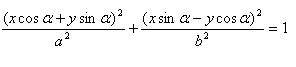
def cutCircleImage(img, img\_height, img\_width)

Tìm các điểm ảnh nằm ngoài đường tròn bằng phương trình đường tròn gán cho màu trắng.

9. Khung là 2 hình ellip chéo nhau:

def cutElipImage(img, img\_height, img\_width)

Tìm các điểm ảnh nằm ngoài 2 đường elip bằng phương trình đường elip gán cho màu trắng.



## Kết quả

Ảnh có chứa văn bản, cây, động vật có vú, chồn đất

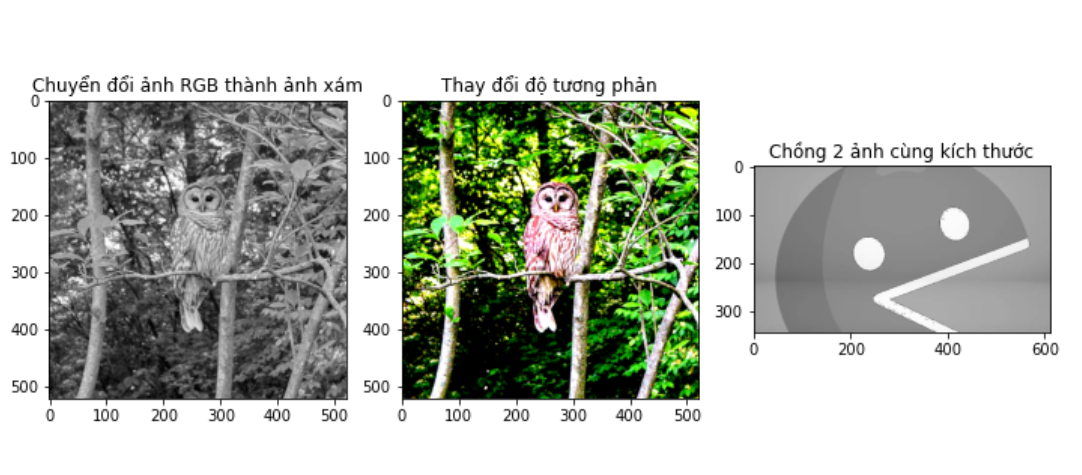
Mô tả được tạo tự động

Ảnh có chứa văn bản, cây

Mô tả được tạo tự động

Ảnh có chứa văn bản, động vật có vú

Mô tả được tạo tự động



# Chức năng

|  |  |  |
| --- | --- | --- |
| **STT** | **Chức năng** | **Đã/chưa hoàn thành** |
| 1 | Thay đổi độ sáng cho ảnh | Đã hoàn thành |
| 2 | Thay đổi độ tương phản | Đã hoàn thành |
| 3 | Lật ảnh (ngang - dọc) | Đã hoàn thành |
| 4 | Chuyển đổi ảnh RGB thành ảnh xám | Đã hoàn thành |
| 5 | Chồng 2 ảnh cùng kích thước | Đã hoàn thành |
| 6 | Làm mờ ảnh | Đã hoàn thành |
| 7 | Cắt khung là hình tròn | Đã hoàn thành |
| 8 | Cắt khung là 2 hình ellip chéo nhau | Đã hoàn thành |

# Mức độ hoàn thành

|  |  |  |
| --- | --- | --- |
| **STT** | **Chức năng** | **Mức độ hoàn thành** |
| 1 | Thay đổi độ sáng cho ảnh | 100% |
| 2 | Thay đổi độ tương phản | 100% |
| 3 | Lật ảnh (ngang - dọc) | 100% |
| 4 | Chuyển đổi ảnh RGB thành ảnh xám | 100% |
| 5 | Chồng 2 ảnh cùng kích thước | 100% |
| 6 | Làm mờ ảnh | 100% |
| 7 | Cắt khung là hình tròn | 100% |
| 8 | Cắt khung là 2 hình ellip chéo nhau | 100% |

# Tài liệu tham khảo

- Tài liệu từ web:

<https://numpy.org/>

<https://www.dfstudios.co.uk/articles/programming/image-programming-algorithms/image-processing-algorithms-part-5-contrast-adjustment/>

<https://stackoverflow.com/questions/12201577/how-can-i-convert-an-rgb-image-into-grayscale-in-python>

<https://www.maa.org/external_archive/joma/Volume8/Kalman/General.html#:~:text=%EF%BF%BDThe%20result%20will%20be,translation%20followed%20by%20a%20rotation.>