

**TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM
TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG
KHOA CÔNG NGHỆ THÔNG TIN**



BÁO CÁO GIỮA KỲ MÔN

NHẬP MÔN XỬ LÝ ẢNH SỐ

Người hướng dẫn: **TRỊNH HÙNG CƯỜNG**

Người thực hiện: **NGUYỄN TRẦN QUANG HUY - 52000668**

Lớp : 20050201

Khoá : 24

THÀNH PHỐ HỒ CHÍ MINH, NĂM 2022

**TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM
TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG
KHOA CÔNG NGHỆ THÔNG TIN**



BÁO CÁO GIỮA KỲ MÔN

NHẬP MÔN XỬ LÝ ẢNH SỐ

Người hướng dẫn: **TRỊNH HÙNG CƯỜNG**

Người thực hiện: **NGUYỄN TRẦN QUANG HUY - 52000668**

Lớp : 20050201

Khoá : 24

THÀNH PHỐ HỒ CHÍ MINH, NĂM 2022

LỜI CẢM ƠN

.

CÔNG TRÌNH ĐƯỢC HOÀN THÀNH TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG

Tôi xin cam đoan đây là công trình nghiên cứu của riêng tôi và được sự hướng dẫn khoa học của TS Nguyễn Văn A;. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong luận văn còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

Nếu phát hiện có bất kỳ sự gian lận nào tôi xin hoàn toàn chịu trách nhiệm về nội dung luận văn của mình. Trường đại học Tôn Đức Thắng không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện (nếu có).

TP. Hồ Chí Minh, ngày 04 tháng 03 năm 2023

Tác giả

(ký tên và ghi rõ họ tên)

Nguyễn Trần Quang Huy

PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN

Phần xác nhận của GV hướng dẫn

Tp. Hồ Chí Minh, ngày tháng năm
(kí và ghi họ tên)

Phần đánh giá của GV chấm bài

Tp. Hồ Chí Minh, ngày tháng năm
(kí và ghi họ tên)

TÓM TẮT

.

MỤC LỤC

LỜI CẢM ƠN	i
CÔNG TRÌNH ĐƯỢC HOÀN THÀNH TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC	
THẮNG	ii
TÓM TẮT	iv
MỤC LỤC.....	1
DANH MỤC KÍ HIỆU VÀ CHỮ VIẾT TẮT	2
DANH MỤC CÁC BẢNG BIỂU	3
DANH MỤC HÌNH VẼ, ĐỒ THỊ	4
Chương 1 – Tối ưu	Error! Bookmark not defined.
1.1 a	Error! Bookmark not defined.
1.1.1 b.....	Error! Bookmark not defined.
1.1.1.1 c	Error! Bookmark not defined.
Chương 1 –	10
Chương 2 –	Error! Bookmark not defined.
Chương 3 –	19
3.1 D	Error! Bookmark not defined.
3.1.1 E	Error! Bookmark not defined.
3.1.1.1 F	Error! Bookmark not defined.
3.1.1.2 A.....	Error! Bookmark not defined.
Chương 4 –	Error! Bookmark not defined.
4.1 e	Error! Bookmark not defined.
4.2 f	Error! Bookmark not defined.

DANH MỤC KÍ HIỆU VÀ CHỮ VIẾT TẮT

CÁC KÝ HIỆU

CÁC CHỮ VIẾT TẮT

DANH MỤC CÁC BẢNG BIỂU

DANH MỤC BẢNG BIỂU

Bảng 3-1 Bảng nề.....**Error! Bookmark not defined.**

DANH MỤC HÌNH VẼ, ĐỒ THỊ

DANH MỤC HÌNH, ĐỒ THỊ

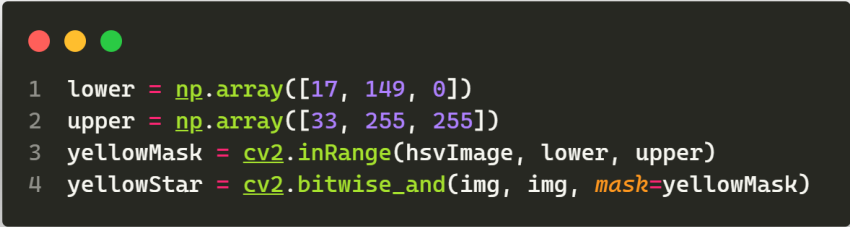
Hình 2-1 Kết quả ngôi sao màu vàng.....	10
Hình 2-2 Kết quả ngôi sao màu tím	10
Hình 2-3 Kết quả ngôi sao màu hồng	11
Hình 2-4 Kết quả ngôi sao màu cam.....	11
Hình 2-5 Kết quả ngôi sao màu xanh lá.....	12
Hình 2-6 Kết quả ngôi sao màu xanh nước.....	12
Hình 2-7 Kết quả đổi màu ngôi sao màu xanh nước sang xanh lá.....	13
Hình 2-8 Kết quả đổi màu ngôi sao màu xanh lá sang xanh lá.....	13
Hình 2-9 Kết quả đổi màu ngôi sao màu cam sang xanh lá.....	14
Hình 2-10 Kết quả đổi màu ngôi sao hồng nước sang xanh lá	14
Hình 2-11 Kết quả đổi màu ngôi sao màu tím sang xanh lá	15
Hình 2-12 Kết quả đổi màu ngôi sao màu vàng sang xanh lá.....	15
Hình 2-13 Kết quả chuyển tất cả ngôi sao sang màu xanh lá	16
Hình 2-14 Kết quả viền của tất cả ngôi sao chuyển sang màu đen.....	16
Hình 2-15 Kết quả tất cả ngôi sao màu đen trên nền trắng.....	17
Hình 2-16 Kết quả ngôi sao được bao quanh bởi viền màu đỏ.....	17
Hình 2-17 Kết quả các ngôi sao được bao quanh bởi hình vuông.....	18
Hình 2-18 Kết quả của gaussian noise và chèn MSSV vào ảnh	18

Chương 1 – Methodology of Solving Tasks

1.1 Viết một đoạn mô tả

Giải thích ngắn gọn ngắn về các phương pháp giải các bài tập trong phần “Programming part”.

1.1.1 Giải thích câu a

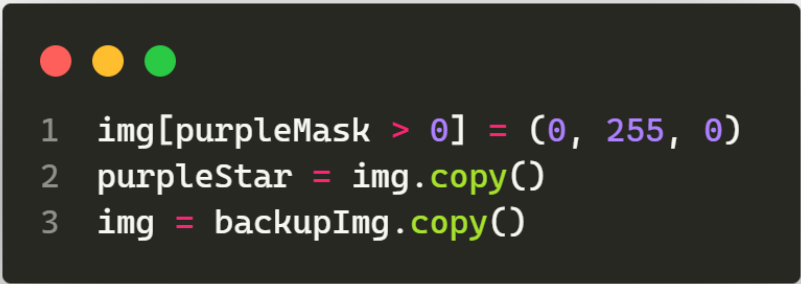


```
1 lower = np.array([17, 149, 0])
2 upper = np.array([33, 255, 255])
3 yellowMask = cv2.inRange(hsvImage, lower, upper)
4 yellowStar = cv2.bitwise_and(img, img, mask=yellowMask)
```

Hình 1-1

Ta sẽ sử dụng hàm `inRange` có trong thư viện `opencv` của python để tạo ra lớp mask, tiếp đến ta sử dụng hàm `bitwise_and` để lấy phần giao của ảnh đầu vào dựa vào lớp mask.

1.1.2 Giải thích câu b

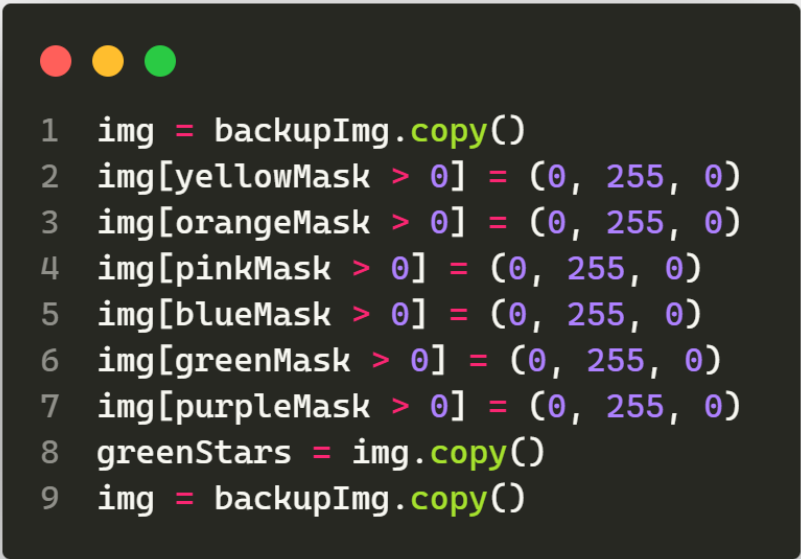


```
1 img[purpleMask > 0] = (0, 255, 0)
2 purpleStar = img.copy()
3 img = backupImg.copy()
```

Hình 1-2

Ta sử dụng lại lớp mask đã tạo từ câu a sau đó với mỗi pixel có giá trị là 255 (nghĩa là màu trắng, hoặc là forceground) thì ta gán lại cho chúng màu xanh lá cây với giá trị màu là (0, 255, 0).

1.1.3 Giải thích câu c



```
1 img = backupImg.copy()
2 img[yellowMask > 0] = (0, 255, 0)
3 img[orangeMask > 0] = (0, 255, 0)
4 img[pinkMask > 0] = (0, 255, 0)
5 img[blueMask > 0] = (0, 255, 0)
6 img[greenMask > 0] = (0, 255, 0)
7 img[purpleMask > 0] = (0, 255, 0)
8 greenStars = img.copy()
9 img = backupImg.copy()
```

Hình 1-3

Ta áp dụng tư tưởng giống câu b và với tất cả lớp mask của 6 ngôi sao gán mỗi pixel trong lớp mask bằng màu xanh lá cây.

1.1.4 Giải thích câu d

```

1 grayImg = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
2
3 lower = np.array([0, 70, 0])
4 upper = np.array([179, 255, 255])
5 borderMask = cv2.inRange(hsvImage, lower, upper)
6 borderMask = cv2.bitwise_not(borderMask)
7 grayImgWithBlackBorder = grayImg.copy()
8 grayImgWithBlackBorder[borderMask > 0] = 0

```

Hình 1-4

Ta áp dụng ý tưởng `inRange` để lấy được lớp mask chứa tất cả các màu trong ảnh ngoại trừ màu trắng sau đó ta dùng `bitwise_not` để phủ định lại mục đích là lấy lại lớp màu trắng. Sau cùng với mỗi pixel trong mask là màu trắng ta cập nhật lại thành màu đen giá trị là 0.

1.1.5 Giải thích câu e

```

1 lower = np.array([0, 160, 0])
2 upper = np.array([179, 255, 255])
3 starMask = cv2.inRange(hsvImage, lower, upper)
4 starMask = cv2.dilate(starMask, np.ones((5, 5), np.uint8), iterations=1)
5
6 starMask = cv2.bitwise_or(starMask, borderMask)
7 starMask = cv2.bitwise_not(starMask)
8 starMask[img.shape[0]-80:img.shape[0], 0:img.shape[1]] = 0
9
10 grayImgWithBlackBorder = grayImg.copy()
11 grayImgWithBlackBorder[starMask > 0] = 0

```

Hình 1-5

Ta dùng `inRange` để lấy ra lớp mask của các ngôi sao sau đó dùng `dilate` để mở rộng vùng mask của các ngôi sao. Tiếp đến ta dùng `bitwise_or` với `borderMask` ở câu d để mở rộng vùng mask ra các viền màu trắng của các ngôi sao. Sau đó ta dùng

bitwise_not để nghịch đảo vùng chọn lại. Sau đó gán vùng dữ liệu ở phía dưới tấm ảnh sang màu đen.

1.1.6 Giải thích câu f

```
1 edged = cv2.Canny(grayImg, 30, 200)
2 contours, hierarchy = cv2.findContours(edged, cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_SIMPLE)
3 startWithRedContours = img.copy()
4 cv2.drawContours(startWithRedContours, contours, -1, (0, 0, 255), 2)
5
```

Hình 1-6

Ta dùng Canny thuật toán lấy cạnh để lấy cạnh của tấm ảnh xám, tiếp theo dùng findContours để lấy ra đường viền trong tấm ảnh. Sử dụng drawContours để vẽ đường viền các ngôi sao trong ảnh.

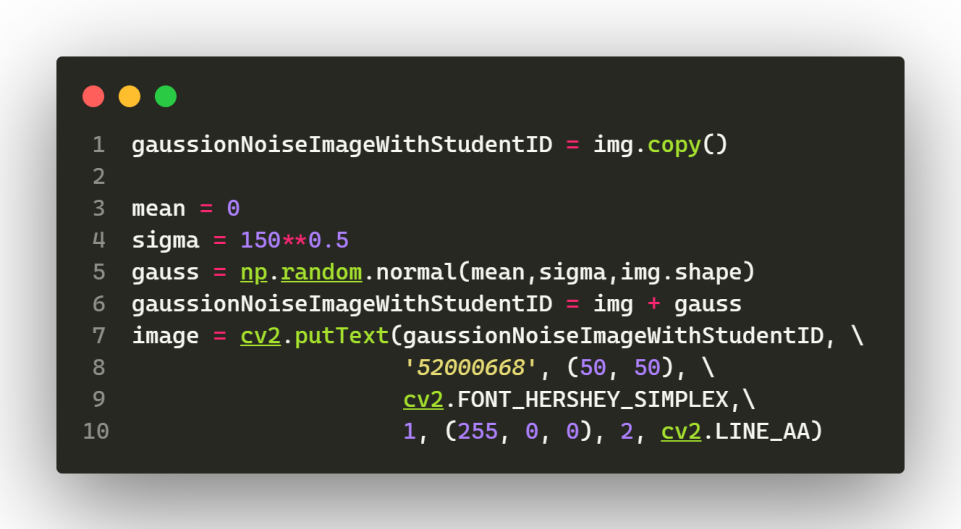
1.1.7 Giải thích câu g

```
1 edged = cv2.Canny(grayImg, 30, 200)
2 ret,thresh = cv2.threshold(grayImg,200,255,cv2.THRESH_BINARY)
3 thresh = cv2.erode(thresh, np.ones((5, 5), np.uint8), iterations=2)
4
5 contours, hierarchy = cv2.findContours(
6     thresh, mode=cv2.RETR_TREE, method=cv2.CHAIN_APPROX_SIMPLE
7 )
8 boxImg = img.copy()
9 print(len(contours))
10 for index in range(len(contours)):
11     cnt = contours[index]
12     x, y, w, h = cv2.boundingRect(cnt)
13     if(w < 100 or h < 100):
14         continue
15     cv2.rectangle(boxImg, (x, y), (x + w, y + h), (0, 255, 0), 2)
16
```

Hình 1-7

Áp dụng ý tưởng giống như câu f tuy nhiên không dùng hàm drawContours mà ta duyệt qua các phần tử trong contours sau đó dùng hàm rectangle để vẽ hình chữ nhật bao quanh ngôi sao tương ứng.

1.1.8 Giải thích câu h



```

1 gaussianNoiseImageWithStudentID = img.copy()
2
3 mean = 0
4 sigma = 150**0.5
5 gauss = np.random.normal(mean, sigma, img.shape)
6 gaussianNoiseImageWithStudentID = img + gauss
7 image = cv2.putText(gaussianNoiseImageWithStudentID, \
8                     '52000668', (50, 50), \
9                          cv2.FONT_HERSHEY_SIMPLEX, \
10                        1, (255, 0, 0), 2, cv2.LINE_AA)

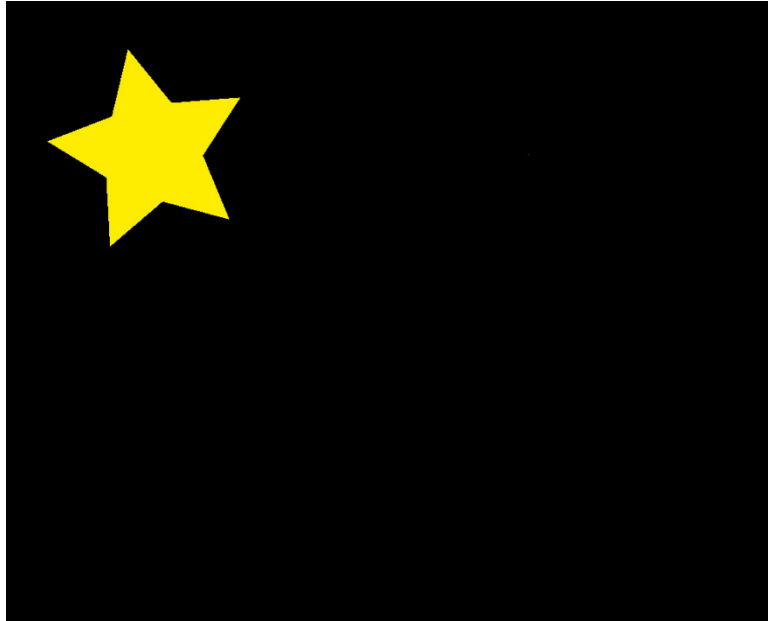
```

Hình 1-8

Dùng hàm np.random.normal sau đó dùng phép toán “+” noise vào ảnh và cuối cùng là gắn MSSV vào ảnh.

Chương 2 – Task results

2.1 Kết quả của câu a



Hình 2-1 Kết quả ngôi sao màu vàng



Hình 2-2 Kết quả ngôi sao màu tím



Hình 2-3 Kết quả ngôi sao màu hồng



Hình 2-4 Kết quả ngôi sao màu cam



Hình 2-5 Kết quả ngôi sao màu xanh lá



Hình 2-6 Kết quả ngôi sao màu xanh nước

2.2 Kết quả của câu b



Hình 2-7 Kết quả đổi màu ngôi sao màu xanh nước sang xanh lá



Hình 2-8 Kết quả đổi màu ngôi sao màu xanh lá sang xanh lá



Hình 2-9 Kết quả đổi màu ngôi sao màu cam sang xanh lá



Hình 2-10 Kết quả đổi màu ngôi sao hồng nước sang xanh lá



Hình 2-11 Kết quả đổi màu ngôi sao màu tím sang xanh lá



Hình 2-12 Kết quả đổi màu ngôi sao màu vàng sang xanh lá

2.3 Kết quả của câu c



Hình 2-13 Kết quả chuyển tất cả ngôi sao sang màu xanh lá

2.4 Kết quả của câu d



Hình 2-14 Kết quả viền của tất cả ngôi sao chuyển sang màu đen

2.5 Kết quả của câu e



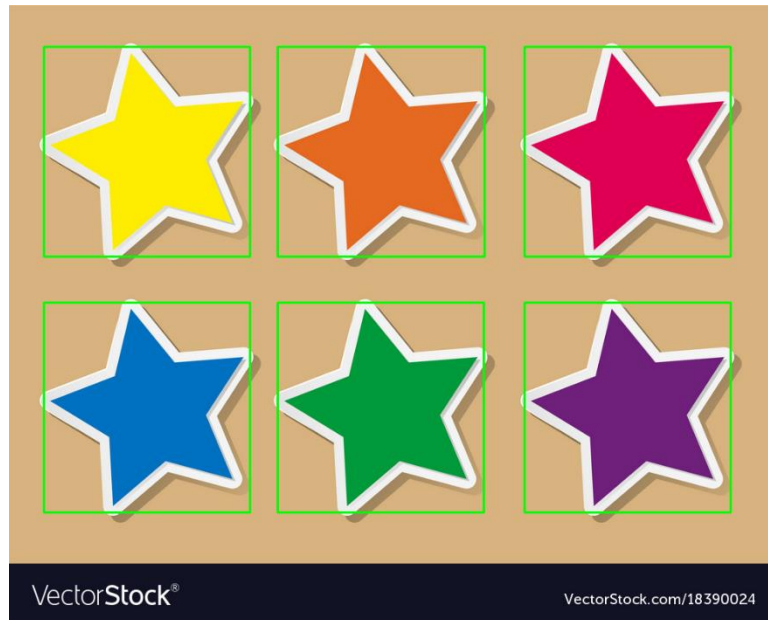
Hình 2-15 Kết quả tất cả ngôi sao màu đen trên nền trắng

2.6 Kết quả của câu f



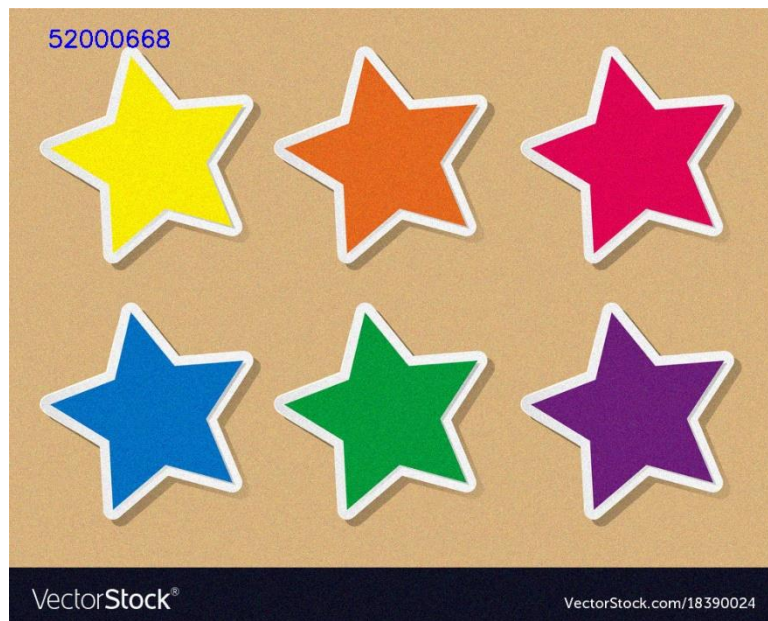
Hình 2-16 Kết quả ngôi sao được bao quanh bởi viền màu đỏ

2.7 Kết quả của câu g



Hình 2-17 Kết quả các ngôi sao được bao quanh bởi hình vuông

2.8 Kết quả của câu h



Hình 2-18 Kết quả của gaussian noise và chèn MSSV vào ảnh

TÀI LIỆU THAM KHẢO

Tiếng Việt

[1].

Tiếng Anh

PHỤ LỤC