

**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN**

**KHOA CÔNG NGHỆ THÔNG TIN**



# **BÀI BÁO CÁO**

## **LAB 01**

**SINH VIÊN THỰC HIỆN**  
**Nguyễn Anh Thư – 18127227**

**GIÁO VIÊN HƯỚNG DẪN**  
**Thầy Lý Quốc Ngọc**  
**Thầy Phạm Minh Hoàng**  
**Thầy Phạm Thanh Tùng**

**BỘ MÔN: XỬ LÝ ẢNH SỐ VÀ VIDEO SỐ**

**Thành phố Hồ Chí Minh – 04/2021**

# MỤC LỤC

|             |  |          |
|-------------|--|----------|
| <b>I.</b>   | <b>Nội dung yêu cầu:</b> .....               | <b>3</b> |
| <b>II.</b>  | <b>Nội dung thực hiện:</b> .....             | <b>3</b> |
| 1.          | Đổi từ ảnh màu sang ảnh xám: .....           | 3        |
| 2.          | Đổi từ ảnh màu RGB sang ảnh màu HSV: .....   | 3        |
| 3.          | Đổi độ sáng ảnh màu hoặc ảnh xám:.....       | 5        |
| 4.          | Đổi độ tương phản ảnh màu hoặc ảnh xám:..... | 6        |
| 5.          | Vẽ lược đồ xám hoặc lược đồ màu: .....       | 7        |
| 6.          | Cân bằng lược đồ xám hoặc lược đồ màu: ..... | 8        |
| <b>III.</b> | <b>Tài liệu tham khảo:</b> .....             | <b>9</b> |

## I. Nội dung yêu cầu:

| STT | YÊU CẦU  | HOÀN THÀNH |
|-----|--|------------|
| 1   | Đọc ảnh đầu vào từ file  | 100%       |
| 2   | Lưu ảnh kết quả vào file   | 100%       |
| 3   | Đổi từ ảnh màu sang ảnh xám                                      | 100%       |
| 4   | Đổi ảnh từ không gian màu RGB sang không gian màu HSV            | 100%       |
| 5   | Đổi độ sáng của ảnh màu hoặc ảnh xám                             | 100%       |
| 6   | Đổi độ tương phản của ảnh màu hoặc ảnh xám                       | 100%       |
| 7   | Vẽ lược đồ đồ xám của ảnh xám hoặc lược đồ màu của ảnh màu       | 100%       |
| 8   | Cân bằng lược đồ độ xám của ảnh xám hoặc lược đồ màu của ảnh màu | 100%       |

## II. Nội dung thực hiện:

### 1. Đổi từ ảnh màu sang ảnh xám:

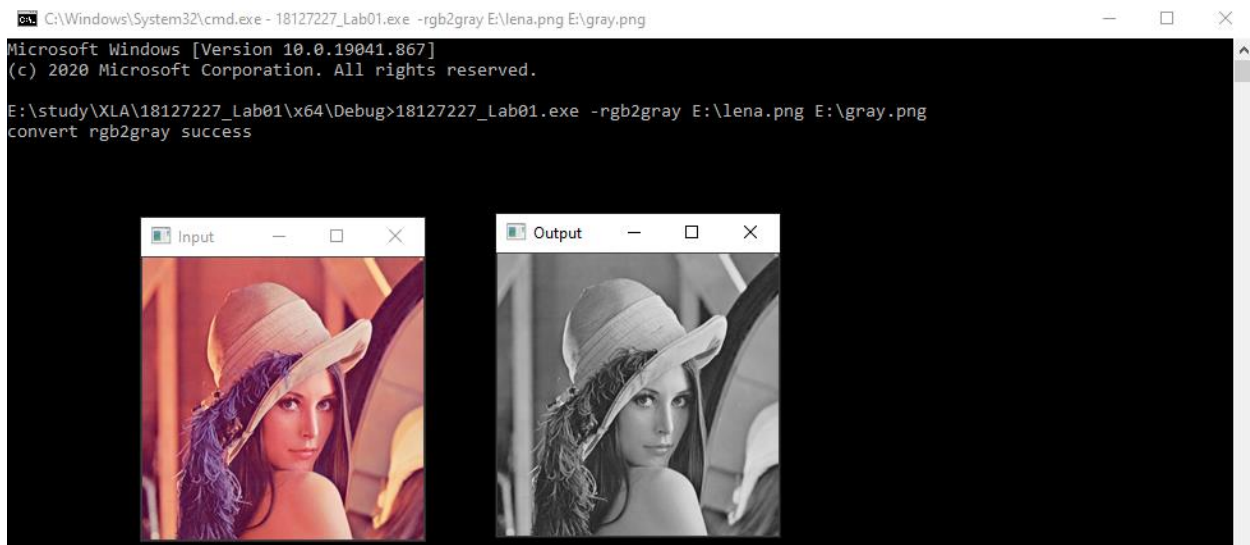
- Input: ảnh màu RGB  $f(x, y)$

- Output: ảnh xám  $g(x, y)$

- Thuật toán sử dụng:

$$g(x, y) = 0.3 * f(x, y).R + 0.59 * f(x, y).G + 0.11 * f(x, y).B$$

- Chạy chương trình và kết quả:



### 2. Đổi từ ảnh màu RGB sang ảnh màu HSV:

- Input: ảnh màu RGB  $f(x, y)$

- Output: ảnh màu HSV  $g(x, y)$

- Thuật toán sử dụng:

## RGB $\leftrightarrow$ HSV

In case of 8-bit and 16-bit images, R, G, and B are converted to the floating-point format and scaled to fit the 0 to 1 range.

$$\begin{aligned} V &\leftarrow \max(R, G, B) \\ S &\leftarrow \begin{cases} \frac{V - \min(R, G, B)}{V} & \text{if } V \neq 0 \\ 0 & \text{otherwise} \end{cases} \\ H &\leftarrow \begin{cases} 60(G - B)/(V - \min(R, G, B)) & \text{if } V = R \\ 120 + 60(B - R)/(V - \min(R, G, B)) & \text{if } V = G \\ 240 + 60(R - G)/(V - \min(R, G, B)) & \text{if } V = B \\ 0 & \text{if } R = G = B \end{cases} \end{aligned}$$

If  $H < 0$  then  $H \leftarrow H + 360$ . On output  $0 \leq V \leq 1, 0 \leq S \leq 1, 0 \leq H \leq 360$ .

The values are then converted to the destination data type:

- 8-bit images:  $V \leftarrow 255V, S \leftarrow 255S, H \leftarrow H/2$  (to fit to 0 to 255)
- 16-bit images: (currently not supported)  $V < -65535V, S < -65535S, H < -H$
- 32-bit images: H, S, and V are left as is

See also

`cv::COLOR_BGR2HSV`, `cv::COLOR_RGB2HSV`, `cv::COLOR_HSV2BGR`, `cv::COLOR_HSV2RGB`

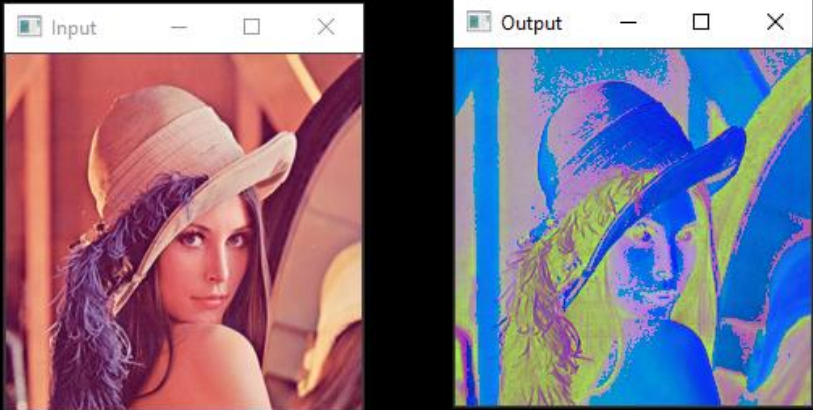
## [OpenCV: Color conversions](#)

- Chạy chương trình và kết quả:

```
C:\Windows\System32\cmd.exe - 18127227_Lab01.exe -rgb2hsv E:\lena.png E:\hsv.png
Microsoft Windows [Version 10.0.19041.867]
(c) 2020 Microsoft Corporation. All rights reserved.

E:\study\XLA\18127227_Lab01\x64\Debug>18127227_Lab01.exe -rgb2gray E:\lena.png E:\gray.png
convert rgb2gray success

E:\study\XLA\18127227_Lab01\x64\Debug>18127227_Lab01.exe -rgb2hsv E:\lena.png E:\hsv.png
convert rgb2hsv success
```

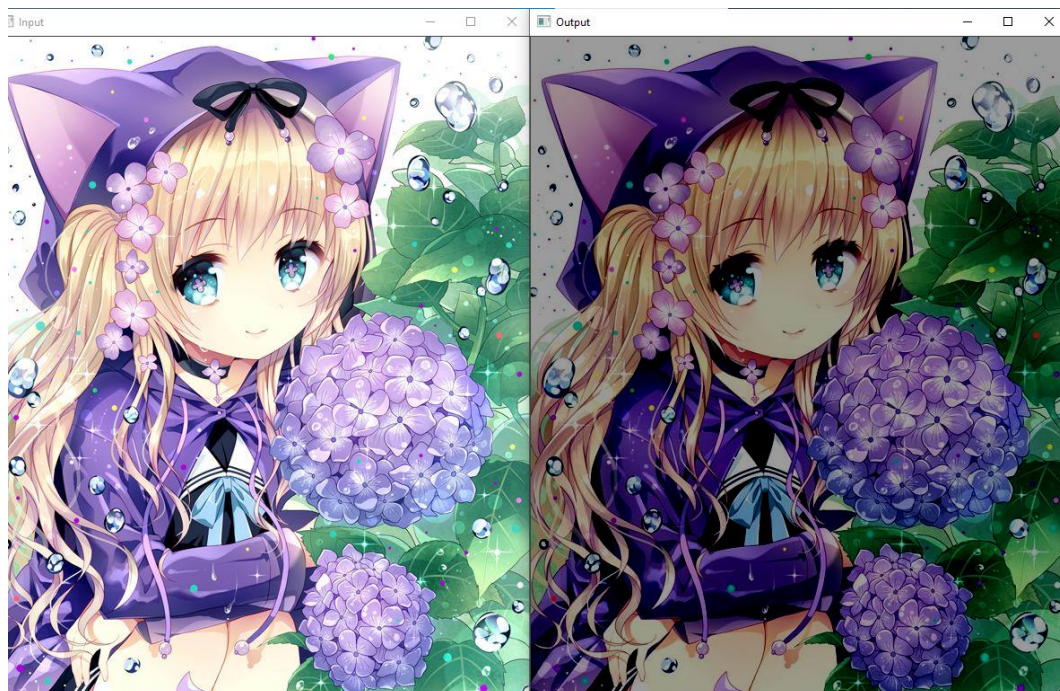


### 3. Đổi độ sáng ảnh màu hoặc ảnh xám:

- Input: ảnh màu hoặc ảnh xám  $f(x, y)$
- Output: ảnh màu hoặc ảnh xám  $g(x, y)$
- Thuật toán sử dụng:

$$g(x, y) = f(x, y) + b$$

- Chạy chương trình và kết quả:





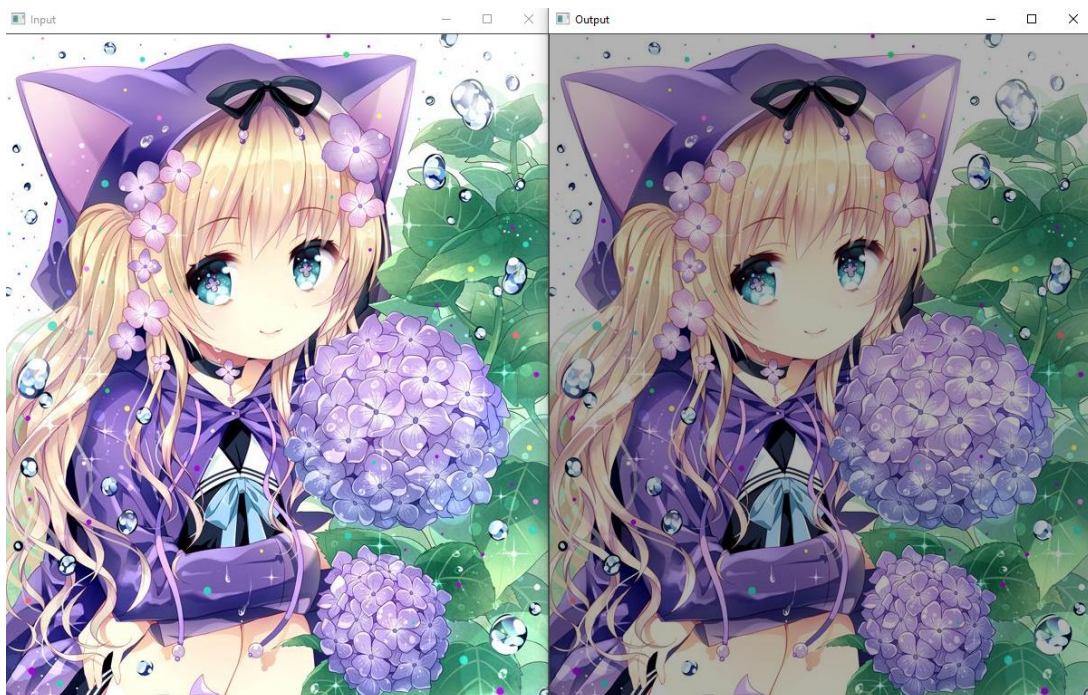
#### 4. Đổi độ tương phản ảnh màu hoặc ảnh xám:

- Input: ảnh màu hoặc ảnh xám  $f(x, y)$
- Output: ảnh màu hoặc ảnh xám  $g(x, y)$
- Thuật toán sử dụng:

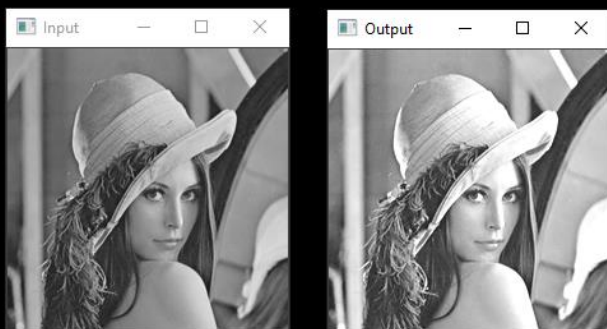
$$g(x, y) = f(x, y) * c$$

- Chạy chương trình và kết quả:

```
C:\Windows\System32\cmd.exe
E:\study\XLA\18127227_Lab01\x64\Debug>18127227_Lab01.exe -contrast E:\neko.jpg E:\cgiam.png 0.7
change contrast success
```



```
C:\Windows\System32\cmd.exe - 18127227_Lab01.exe -contrast E:\gray.png E:\ctang.png 1.3
E:\study\XLA\18127227_Lab01\x64\Debug>18127227_Lab01.exe -contrast E:\gray.png E:\ctang.png 1.3
change contrast success
```



## 5. Vẽ lược đồ xám hoặc lược đồ màu:

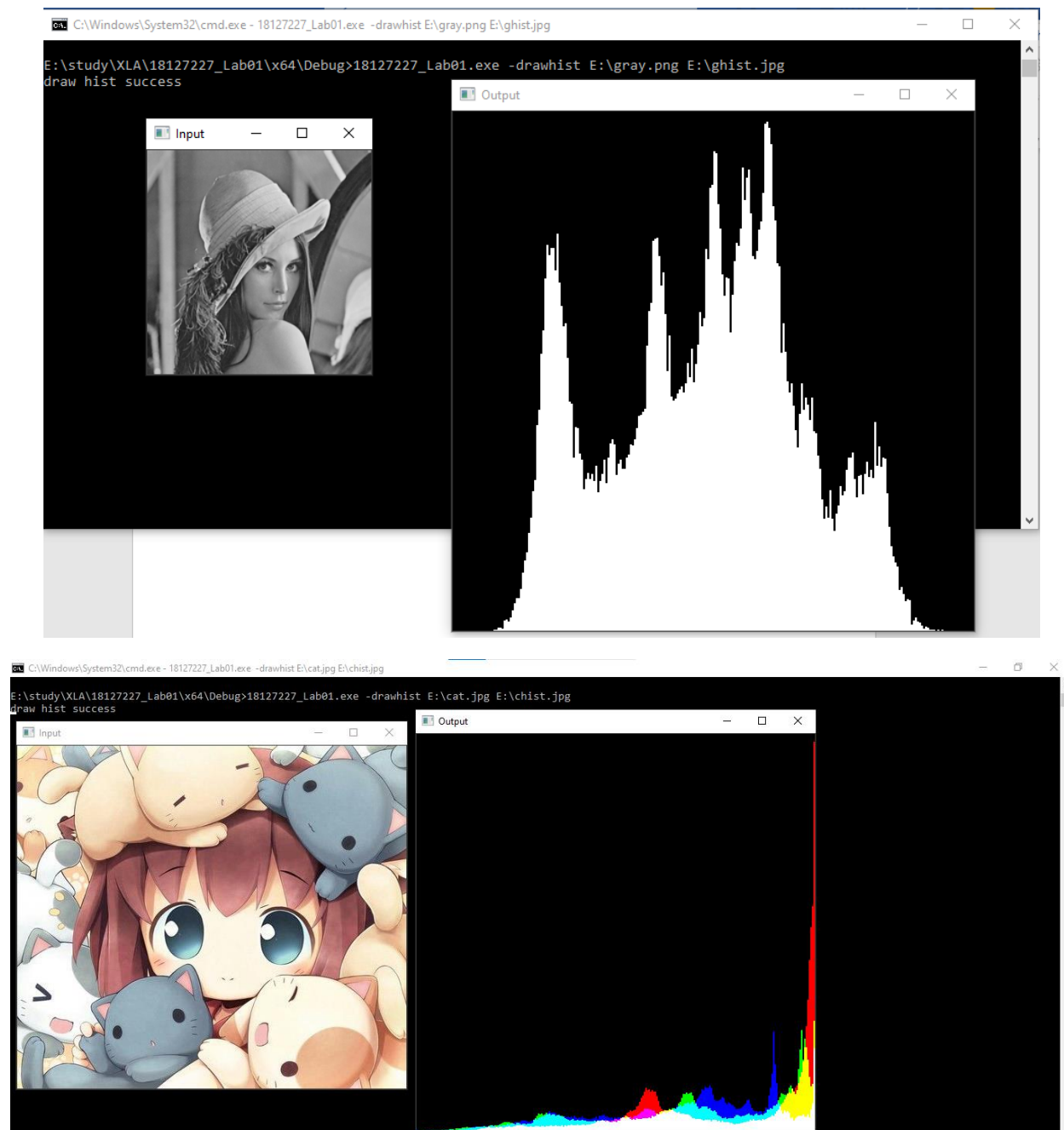
- Input: ảnh màu hoặc ảnh xám  $f(x, y)$

- Output: ảnh histogram  $H$

- Thuật toán sử dụng:

$$H[f(x, y)] += 1$$

- Chạy chương trình và kết quả:



## 6. Cân bằng lược đồ xám hoặc lược đồ màu:

- Input: ảnh màu hoặc ảnh xám  $f(x, y)$
- Output: ảnh màu hoặc ảnh xám  $g(x, y)$
- Thuật toán sử dụng:

### 3.1.3.1. Histogram Equalization (Algorithm)

**Step1.** Create an array  $H$  of length  $nG$  initialized with 0 values (for an  $N \times M$  image  $f$  of  $nG$  grey-levels).

**Step2.** Form the **image histogram** of  $f$ , save to  $H$

$$H[f(x, y)] += 1$$

**Step3.** Form the **cumulative image histogram** of  $f$ , save to  $T$

$$T[0] = H[0]; \quad T[p] = T[p-1] + H[p], \quad p = 1, 2, \dots, nG-1$$

**Step4.** Constructing a **lookup table  $T$**  in range  $[0; nG-1]$

$$T[p] = \text{round}((nG-1)/NM)T[p]$$

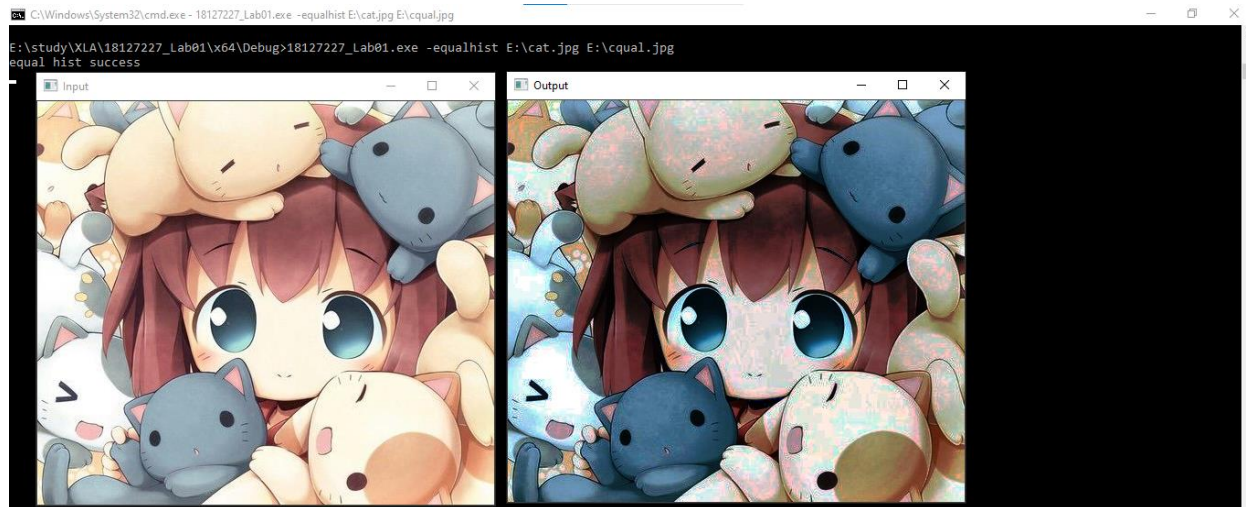
**Step5.** Form the **output image  $g$** :  $g(x, y) = T[f(x, y)]$

Tham khảo từ slide lý thuyết

- Chạy chương trình và kết quả:







### III. Tài liệu tham khảo:

- (1) Slide lý thuyết Image Pre-processing
- (2) [OpenCV: Color conversions](#)