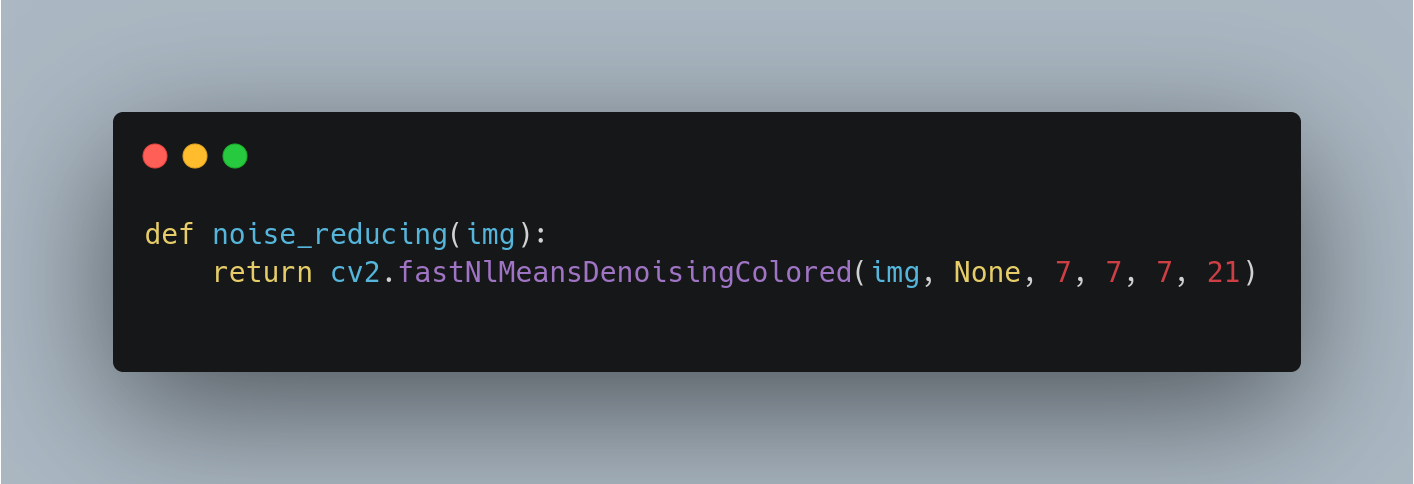
1. Thư viện:

-OpenCV: cung cấp công cụ đọc ảnh, chuyển đổi hệ màu và các công cụ xử lý ảnh.

-Face\_recognition: cung cấp công cụ nhận diện khuôn mặt.

2. Các bước thực hiện:

1. Noise reducing



Sử dụng hàm fastNlMeansDenoisingColored(src, dst, h\_luminance, photo\_render, block\_size, search\_window) được cung cấp bởi thư viện opencv, ở đây hàm sử dụng thuật toán Non-Local Means Denoising để khử nhiểu.

Argument:

-src: ảnh input

-dst: ảnh output cùng kích thước với input

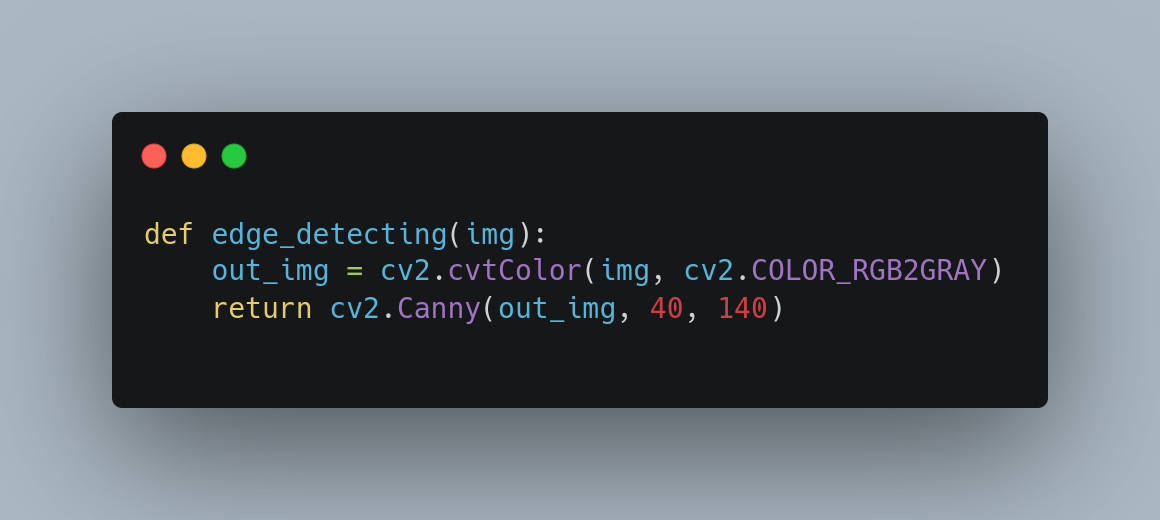
-h\_luminance: mức độ lọc, giá trị càng cao thì khả năng khử nhiễu càng mạnh nhưng cùng với đó là gây mất chi tiết ảnh.

-photo\_render: tương tự với h\_luminance nhưng dành cho thành phần màu. Với nhiều ảnh thì giá trị 10 sẽ loại bỏ nhiễu màu và không gây sai lệch màu sắc.

-block\_size: kích thước pixels dùng để tính trọng số. Giá trị đề xuất là 7.

-search\_window: kích thước của cửa sổ quét qua từng điểm ảnh. Giá trị đề xuất là 21.

1. Edge decting



-Đầu tiên ta chuyển đổi hệ màu của ảnh sang hệ màu xám.

-Tiếp đến ta dùng hàm Canny(src, threshold1, threshold2) được cung cấp bởi opencv để dò cạnh. Ở đây hàm sử dụng phương pháp dò cạnh canny, tập hợp những thuật toán để dò độ rộng của cạnh.

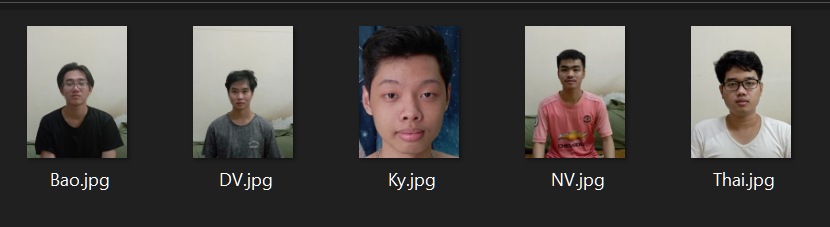
Argument:

-src: ảnh input

-threshold1: ngưỡng hysteresis thứ nhất.

-threshold2: ngưỡng hysteresis thứ hai.

1. Face detecting and recognize



-Đầu tiên thu thập ảnh chứa mặt các thành viên trong nhóm để làm ảnh mẫu mà chúng ta đã định nghĩa trong thư mục samples



-Tiếp đến ta tiến hành mã hóa các ảnh chứa mặt của cách thành viên bằng hàm face\_encodings của thư viện face\_recognition, hàm sẽ trả về 1 mảng 128 chiều công động những chi tiết của khuôn mặt có trong ảnh. Sau đó tạo 2 danh sách, 1 danh sách chứa những khuôn mặt đã mã hóa và danh sách còn lại chứa tên của những khuôn mặt tương ứng.



-Tiếp đến ta tiến hành xác định vị trí các khuôn mặt của ảnh input bằng cách sử dụng hàm face\_location của thư viện face\_recognition, mặc định hàm sẽ sử dụng phương pháp HOG (histogram of oriented gradients) để xác định khuôn mặt. Sau đó ta dùng face\_encodings để mã hóa các khuôn mặt có trong ảnh input. Với từng khuôn mặt có trong input ta sẽ so sánh với những khuôn mặt mẫu bằng các sử dụng hàm compare\_faces với những tham số truyền vào là danh sách các khuôn mặt mẫu, khuôn mặt đang xét, ngưỡng khoảng cách cho phép (khoảng cách 2 khuôn mặt), ở đây hàm sẽ trả về 1 danh sách gồm các giá trị True và False cho biết có khuôn mặt mẫu nào khớp với khuôn mặt đang xét. Nếu khuôn mặt đang xét khớp với một khuôn mặt mẫu nào đó thì ta sẽ đặt tên cho khuôn mặt này ứng với khuôn mặt mẫu đó nếu không thì tên của khuôn mặt này sẽ là rỗng. Cuối cùng ta sẽ vẽ khung và điền tên cho khuôn mặt vào ảnh.