TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**

A red and blue logo

Description automatically generated

**BÀI TẬP ĐÁNH GIÁ GIỮA KỲ MÔN XÁC SUẤT VÀ THỐNG KÊ ỨNG DỤNG CHO CÔNG NGHỆ THÔNG TIN**

*Người hướng dẫn*: GV NGUYỄN LÂM

*Người thực hiện*: **NGUYỄN ĐÌNH VIỆT HOÀNG – 522H0120**

Lớp **: 22H50302**

Khoá  **: 26**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2023**

TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**

A red and blue logo

Description automatically generated

**BÀI TẬP ĐÁNH GIÁ GIỮA KỲ MÔN XÁC SUẤT VÀ THỐNG KÊ ỨNG DỤNG CHO CÔNG NGHỆ THÔNG TIN**

*Người hướng dẫn*: GV NGUYỄN LÂM

*Người thực hiện*: **NGUYỄN ĐÌNH VIỆT HOÀNG – 522H0120**

Lớp **: 22H50302**

Khoá  **: 26**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2023**

LỜI CẢM ƠN

Kính gửi thầy Nguyễn Lâm cùng toàn thể quý thầy cô nhà trường, trong không gian tri thức và sự hướng dẫn của thầy và nhà trường, tôi đã có cơ hội tham gia vào một chặng đường đầy ý nghĩa và phát triển - quá trình đánh giá giữa kỳ của tôi. Nhân dịp quá trình này đi đến bước kết thúc, tôi xin gửi lời cảm ơn sâu sắc nhất đến thầy và toàn thể quý thầy cô trong nhà trường. Trước tiên, tôi muốn bày tỏ lòng biết ơn vô hạn đến thầy Nguyễn Lâm. Sự tận tâm, kiên nhẫn và sự truyền cảm hứng mà thầy đã dành cho tôi không thể đong đếm được. Thầy không chỉ là người truyền đạt kiến thức mà còn là người hướng dẫn và đồng hành trên con đường phát triển cá nhân của tôi. Thầy đã dành nhiều thời gian và công sức để giúp tôi hiểu rõ hơn về đề bài giữa kỳ và đồng thời hỗ trợ tôi trong việc nắm vững kiến thức và kỹ năng cần thiết để có thể hoàn thành bài tập đánh giá giữa kỳ một cách tốt nhất. Tôi cũng muốn gửi lời cảm ơn đến gia đình và những người thân yêu của mình. Sự hỗ trợ, khích lệ và động viên không ngừng từ phía họ đã là nguồn lực quan trọng giúp tôi vượt qua những khó khăn trong quá trình nghiên cứu và thực hiện bài tập giữa kỳ này. Đối với những người thân yêu của tôi, quá trình này không chỉ là của riêng tôi mà còn là của cả gia đình, và tôi rất biết ơn sự ủng hộ và tin tưởng của họ.

**BÀI TẬP ĐƯỢC HOÀN THÀNH**

**TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

Tôi xin cam đoan đây là sản phẩm bài tập của riêng tôi và được sự hướng dẫn của GV Nguyễn Lâm. Các nội dung nghiên cứu, kết quả trong bài tập này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây.

**Nếu phát hiện có bất kỳ sự gian lận nào tôi xin hoàn toàn chịu trách nhiệm về nội dung bài tập 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 1 tháng 8 năm 2023*

*Tác giả*

*(ký tên và ghi rõ họ tên)*

*Hoàng*

*Nguyễn Đình Việt Hoàng*

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

* Đầu tiên tôi sẽ xem tổng thể yêu cầu đề bài và đặt mục tiêu cho từng chương để hoàn thành xuất sắc chương đấy. Với chương 1,2 và 3, tôi sẽ tra google về từng hàm của mỗi chương có chức năng gì, định nghĩa hàm là gì và xem đoạn mã ví dụ trên các trang web có sẵn để từ đó có thể tự viết cho mình một đoạn code của riêng mình và chạy thử xem có lỗi gì hay không để có thể debug.
* Trong thống kê, việc tính toán các giá trị trung bình và đo lường vị trí trung tâm của tập dữ liệu là rất quan trọng để hiểu và phân tích dữ liệu. Thư viện ‘**statistics**’ trong Python cung cấp nhiều hàm hữu ích để giúp chúng ta thực hiện các phép tính này một cách dễ dàng và chính xác.
* Thư viện ‘**statistics**’ chứa nhiều hàm để tính toán các giá trị thống kê như trung bình cộng, trung vị, trung bình hình học, độ lệch chuẩn và các hàm đo lường mối quan hệ giữa hai tập dữ liệu.

MỤC LỤC

LỜI CẢM ƠN…………………………………………………………………………..3

PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN…………………………...5

TÓM TẮT………………………………………………………………………………6

MỤC LỤC……………………………………………………………………………....7

DANH MỤC CÁC HÌNH ẢNH………………………………………………………………..8

CHƯƠNG 1 - THUẬT TOÁN CÂN BẰNG LƯỢC ĐỒ………………………………………9

1.1 Nêu vấn đề, ràng buộc/điều kiện (nếu có), phương pháp/ thuật toán…….....9

1.2 Ví dụ………………………………………………………………………..10

1.3 Nhận xét, phân tích, đánh giá của tôi............................................................11

CHƯƠNG 2 - THỰC HIỆN…………………………………………………………...13

2.1 Giải thích code của tôi từng bước chứ không chỉ đơn giản sao chép code vào chương này……………………………………………………………………..13

2.2 Hướng dẫn xây dựng và chạy mã nguồn của tôi……………………………17

2.3 Thử nghiệm: dữ liệu, kết quả và kết luận…………………………………..17

DANH MỤC CÁC HÌNH ẢNH

**DANH MỤC HÌNH**

Hình 2.3.1. Hình ảnh thể hiện kết quả của đoạn code của cách 1 trong mục 2.1 của chương 2……………………………………………………………………18

Hình 2.3.2. Hình ảnh thể hiện kết quả của đoạn code của cách 1 trong mục 2.1 của chương 2……………………………………………………………………19

CHƯƠNG 1 – THUẬT TOÁN CÂN BẰNG LƯỢC ĐỒ

* 1. Nêu vấn đề, ràng buộc/điều kiện (nếu có), phương pháp/ thuật toán…:
* Vấn đề:

+ Trong xử lý ảnh, độ tương phản của một hình ảnh có thể không phản ánh đầy đủ chi tiết. Khi các mức xám không được phân bố đều, hình ảnh có thể trở nên mờ nhạt hoặc thiếu chi tiết. Vấn đề này thường xảy ra trong các hình ảnh có độ tương phản thấp. Điều này dẫn đến việc mất mát thông tin và khả năng hiển thị kém. Vấn đề này có thể được giải quyết bằng cách áp dụng thuật toán cân bằng lược đồ để làm cho phân bố mức xám trong hình ảnh đều rộng rãi, từ đen đến trắng. Cân bằng lược đồ là một công cụ mạnh mẽ để cải thiện chất lượng hình ảnh và tạo ra các hình ảnh có độ tương phản cao và rõ ràng hơn.

* Ràng buộc/ điều kiện:

+ Hình ảnh đầu vào phải là ảnh xám (không phải hình ảnh màu).

+ Hình ảnh phải có độ tương phản thấp hoặc bị mất cân bằng lược đồ.

+ Hình ảnh phải có ít nhất hai mức xám khác nhau để có thể thực hiện cân bằng lược đồ.

+ Dữ liệu ảnh phải được biểu diễn bằng các giá trị mức xám (từ 0 đến 255).

* Phương pháp/ thuật toán:

+ Thuật toán cân bằng lược đồ bao gồm các bước chính sau:

1. Tính histogram: Đếm số lượng mỗi mức xám xuất hiện trong hình ảnh ban đầu.
2. Tính hàm phân bố tích lũy (CDF): Tích lũy các tần suất từ histogram để tạo ra hàm phân bố tích lũy.
3. Chuẩn hóa hàm CDF: Điều chỉnh hàm CDF sao cho nó nằm trong khoảng giá trị [0, 255].
4. Ánh xạ mức xám mới: Ánh xạ từng mức xám trong hình ảnh gốc sang mức xám mới bằng cách sử dụng hàm CDF đã chuẩn hóa.
5. Áp dụng ánh xám mới vào hình ảnh: Thay thế các mức xám ban đầu bằng các mức xám mới tính toán.

* Công thức:

+ Hàm phân bố tích lũy (CDF): , trong đó P(i) là xác suất xuất hiện mức xám i.

+ Ánh xám mới: *NEW*\_*GRAY*=*CDF*(*GRAY*)×(*L*−1), trong đó GRAY là mức xám ban đầu, L là số mức xám (thường là 256).

* Ưu điểm và Nhược điểm:

+ Ưu điểm: Tăng độ tương phản, làm rõ chi tiết trong hình ảnh, dễ triển khai.

+ Nhược điểm: Có thể thay đổi tỷ lệ màu sắc ban đầu, không phù hợp cho mọi loại hình ảnh.

* 1. Ví dụ:

+ Xử lý ảnh y khoa: Trong lĩnh vực y học, việc cân bằng lược đồ có thể được sử dụng để tăng cường các chi tiết quan trọng trong hình ảnh chụp cắt lớp, chẳng hạn như tia X hoặc cộng hưởng từ (MRI). Điều này giúp bác sĩ dễ dàng nhận biết các tình trạng bệnh và phân tích hình ảnh một cách chính xác hơn.

+ Cải thiện ảnh quang học: Trong quang học, thuật toán cân bằng lược đồ có thể được áp dụng để cải thiện hình ảnh được chụp từ kính hiển vi hoặc teleskop. Điều này giúp làm rõ các chi tiết và cấu trúc trong hình ảnh, làm cho việc nghiên cứu và quan sát trở nên hiệu quả hơn.

+ Xử lý ảnh vệ tinh: Trong việc xử lý hình ảnh vệ tinh, cân bằng lược đồ có thể giúp cải thiện chất lượng hình ảnh chụp từ vệ tinh, làm cho các đối tượng và đặc điểm trên bề mặt Trái Đất rõ ràng hơn. Điều này hữu ích cho việc theo dõi tình trạng môi trường, sự biến đổi địa hình và nghiên cứu địa chất.

+ Chỉnh sửa ảnh nghệ thuật: Trong ngành nghệ thuật số, thuật toán cân bằng lược đồ có thể được sử dụng để điều chỉnh mức xám trong ảnh và tạo ra hiệu ứng nghệ thuật độc đáo. Điều này giúp tạo ra các tác phẩm ảnh nghệ thuật với màu sắc và tông màu thú vị.

+ Xử lý ảnh địa lý: Trong lĩnh vực địa lý và bản đồ, thuật toán cân bằng lược đồ có thể được sử dụng để tăng cường chi tiết địa hình và làm rõ các đối tượng địa lý trên bản đồ. Điều này hỗ trợ trong việc tạo ra bản đồ chính xác và trực quan.

* 1. Nhận xét, phân tích, đánh giá của tôi:

1. Ưu điểm:
   * Cải thiện tương phản: Thuật toán cân bằng lược đồ giúp tăng cường tương phản giữa các mức xám trong ảnh. Điều này làm cho các chi tiết và cạnh trong ảnh dễ dàng nhận biết hơn.
   * Làm đẹp hình ảnh: Thuật toán cân bằng lược đồ có thể làm cho hình ảnh trở nên đẹp hơn và dễ nhìn hơn bằng cách làm cho các mức xám phân bố đều.
   * Khắc phục quá trình chiếu sáng không đồng đều: Khi ảnh bị chiếu sáng không đồng đều, các vùng quá sáng hoặc quá tối có thể được điều chỉnh để có độ tương phản tốt hơn.
2. Hạn chế:
   * Không phù hợp cho mọi trường hợp: Trong một số tình huống, việc cân bằng lược đồ có thể làm mất mát thông tin và gây mất mát các chi tiết quan trọng trong ảnh.
   * Không xử lý được các vấn đề phức tạp: Thuật toán cân bằng lược đồ không thể giải quyết tốt các vấn đề phức tạp như nhiễu, nhiễu màu, hay biến dạng hình ảnh.
3. Cách hoạt động:
   * Tính toán histogram: Thuật toán bắt đầu bằng việc tính toán histogram của ảnh gốc, tức là số lượng pixel có cùng mức xám trong ảnh.
   * Tích luỹ histogram: Từ histogram, tích luỹ histogram được tính toán. Tích luỹ histogram cho biết tổng số pixel có mức xám nhỏ hơn hoặc bằng mức xám đó.
   * Tính giá trị mới cho mỗi mức xám: Sử dụng tích luỹ histogram, giá trị mới cho mỗi mức xám được tính toán. Công thức thường sử dụng là chia tích luỹ histogram cho tổng số pixel và nhân với 255.
   * Thay thế giá trị mức xám: Mỗi giá trị mức xám trong ảnh gốc được thay thế bằng giá trị mới tương ứng đã tính toán.
4. Áp dụng trong thực tế:
   * Cải thiện tương phản: Thuật toán cân bằng lược đồ thường được sử dụng để cải thiện tương phản trong ảnh chụp dưới điều kiện ánh sáng không tốt, hoặc trong các tình huống khi độ tương phản yếu.
   * Xử lý ảnh y tế: Trong lĩnh vực y tế, thuật toán cân bằng lược đồ có thể được sử dụng để tăng cường chi tiết trong hình ảnh chụp từ các thiết bị y tế như X-quang, MRI.
   * Xử lý ảnh nghệ thuật: Trong nghệ thuật số và xử lý ảnh nghệ thuật, thuật toán này có thể được sử dụng để tạo ra các hiệu ứng độ tương phản đặc biệt.
   * Phương pháp này cho chất lượng dữ liệu rất cao, cải thiện chất lượng model deep learning rất nhiều.

CHƯƠNG 2 – THỰC HIỆN

2.1 Giải thích code của tôi từng bước chứ không chỉ đơn giản sao chép code vào chương này:

# Cách 1:

# Import thư viện và modules

# Đây là việc import thư viện numpy để làm việc với mảng và tính toán số học, và pyplot từ thư viện matplotlib để hiển thị ảnh và biểu đồ.

import numpy as np

from matplotlib import pyplot as plt

# Hàm thuattoancanbangluocdo(image): Đây là hàm thực hiện thuật toán cân bằng lược đồ cho ảnh đầu vào.

def thuattoancanbangluocdo(image):

# Tính toán lược đồ của ảnh gốc

# Dùng mảng luocdo để tính số lần xuất hiện của mỗi mức xám trong ảnh. Hàm ravel() được sử dụng để duỗi mảng ảnh thành mảng 1D, từ đó dễ dàng duyệt qua từng pixel.

luocdo = np.zeros(256)

for pixel in image.ravel():

luocdo[pixel] += 1

# Tính toán cdf (cumulative distribution function) của lược đồ

# Tính toán hàm phân phối tích lũy (CDF) từ lược đồ. CDF là tổng lũy thừa của lược đồ và đại diện cho phân phối tổng lũy thừa của mức xám trong ảnh.

cdf = np.zeros\_like(luocdo)

cdf[0] = luocdo[0]

for x in range(1, len(luocdo)):

cdf[x] = cdf[x-1] + luocdo[x]

# Chuẩn hóa cdf để có giá trị trong khoảng [0, 255]

# Chuẩn hóa CDF sao cho giá trị của CDF nằm trong khoảng [0, 255], để đảm bảo rằng giá trị của ảnh sau khi cân bằng sẽ nằm trong khoảng này.

cdf\_normalized = (cdf - cdf.min()) \* 255 / (cdf.max() - cdf.min())

# Áp dụng thuật toán cân bằng lược đồ cho từng pixel của ảnh

# Sử dụng hàm np.interp() để ánh xạ giá trị mức xám của ảnh gốc sang giá trị mới theo CDF đã chuẩn hóa. Điều này giúp cân bằng lược đồ cho ảnh.

image\_equalized = np.interp(image, np.arange(256), cdf\_normalized)

# Trả về ảnh kết quả

return image\_equalized

# Đọc ảnh gốc và chuyển đổi sang ảnh xám

# Sử dụng plt.imread() để đọc ảnh gốc, sau đó tính trung bình của các kênh màu (RGB) để chuyển đổi ảnh màu sang ảnh xám.

image = plt.imread('input.jpg')

gray = np.mean(image, axis=2).astype(np.uint8)

# Áp dụng thuật toán cân bằng lược đồ cho ảnh xám

gray\_equalized = thuattoancanbangluocdo(gray)

# Hiển thị kết quả

# Dùng plt.subplot() để tạo cửa sổ hiển thị gồm hai phần: ảnh gốc và ảnh sau khi cân bằng lược đồ.

# plt.imshow() dùng để hiển thị ảnh và plt.title() để đặt tiêu đề.

# Cuối cùng, plt.show() để hiển thị cửa sổ đồ họa.

plt.subplot(121), plt.imshow(gray, cmap='gray')

plt.title('Original Image')

plt.subplot(122), plt.imshow(gray\_equalized, cmap='gray')

plt.title('Histogram Equalized Image')

plt.show()

# Cách 2:

# Import thư viện cv2 (OpenCV) và numpy để làm việc với mảng và tính toán số học.

import cv2

import numpy as np

# Hàm histogram\_equalization(image): Đây là hàm thực hiện thuật toán cân bằng lược đồ cho ảnh đầu vào.

def histogram\_equalization(image):

# Dùng mảng histogram để tính số lần xuất hiện của mỗi mức xám trong ảnh.

histogram = np.zeros(256, dtype=int)

# Hàm image.shape trả về kích thước ảnh (chiều cao và chiều rộng).

for i in range(image.shape[0]):

for j in range(image.shape[1]):

histogram[image[i, j]] += 1

# Tính toán hàm phân phối tích lũy (CDF) từ lược đồ. CDF là tổng lũy thừa của lược đồ và đại diện cho phân phối tổng lũy thừa của mức xám trong ảnh.

cdf = np.zeros(256, dtype=int)

cdf[0] = histogram[0]

for i in range(1, 256):

cdf[i] = cdf[i-1] + histogram[i]

# Chuẩn hóa CDF sao cho giá trị của CDF nằm trong khoảng [0, 255], để đảm bảo rằng giá trị của ảnh sau khi cân bằng sẽ nằm trong khoảng này.

cdf\_normalized = (cdf - cdf.min()) \* 255 / (cdf.max() - cdf.min())

# Tạo một mảng trống có cùng kích thước với ảnh đầu vào để chứa ảnh sau khi cân bằng lược đồ.

equalized\_image = np.zeros\_like(image)

# Sử dụng mảng cdf\_normalized để ánh xạ giá trị mức xám của ảnh gốc sang giá trị mới dựa trên CDF đã chuẩn hóa.

for i in range(image.shape[0]):

for j in range(image.shape[1]):

equalized\_image[i, j] = cdf\_normalized[image[i, j]]

return equalized\_image

# Đọc ảnh đầu vào và chuyển đổi sang ảnh xám bằng cách sử dụng cv2.imread với cờ cv2.IMREAD\_GRAYSCALE.

image = cv2.imread('input.jpg', cv2.IMREAD\_GRAYSCALE)

# Thực hiện cân bằng lược đồ bằng cách áp dụng hàm cân bằng lược đồ lên ảnh xám.

equalized\_image = histogram\_equalization(image)

# Hiển thị hai cửa sổ hiển thị gồm ảnh gốc và ảnh sau khi cân bằng lược đồ sử dụng cv2.imshow().

# Hàm cv2.waitKey() dùng để chờ một sự kiện từ bàn phím trước khi đóng cửa sổ.

cv2.imshow('Original Image', image)

cv2.imshow('Equalized Image', equalized\_image)

cv2.waitKey(0)

cv2.destroyAllWindows()

2.2 Hướng dẫn xây dựng và chạy mã nguồn của tôi:

* *Cách 1:*

1. Tạo một tập tin ảnh có tên là ‘input.jpg’ trong cùng thư mục với đoạn mã.
2. Cài đặt thư viện ‘numpy’ và ‘matplotlib’ nếu chưa có bằng cách sử dụng lệnh ‘pip install numpy’ và ‘pip install matplotlib’ trong Command Prompt hoặc Terminal.
3. Chạy đoạn mã trong môi trường Python.

* *Cách 2:*

1. Tạo một tập tin ảnh có tên là ‘input.jpg’ trong cùng thư mục với đoạn mã.
2. Cài đặt thư viện ‘numpy’ và ‘opencv-python’ nếu chưa có bằng cách sử dụng lệnh ‘pip install numpy’ và ‘pip install opencv-python’ trong Command Prompt hoặc Terminal.
3. Chạy đoạn mã trong môi trường Python.

2.3 Thử nghiệm: dữ liệu, kết quả và kết luận:

* *Cách 1:*

+ Dữ liệu: Ảnh ‘input.jpg’.

+ Kết quả: Hai ảnh được hiển thị, một là ảnh gốc và một là ảnh sau khi cân bằng lược đồ.

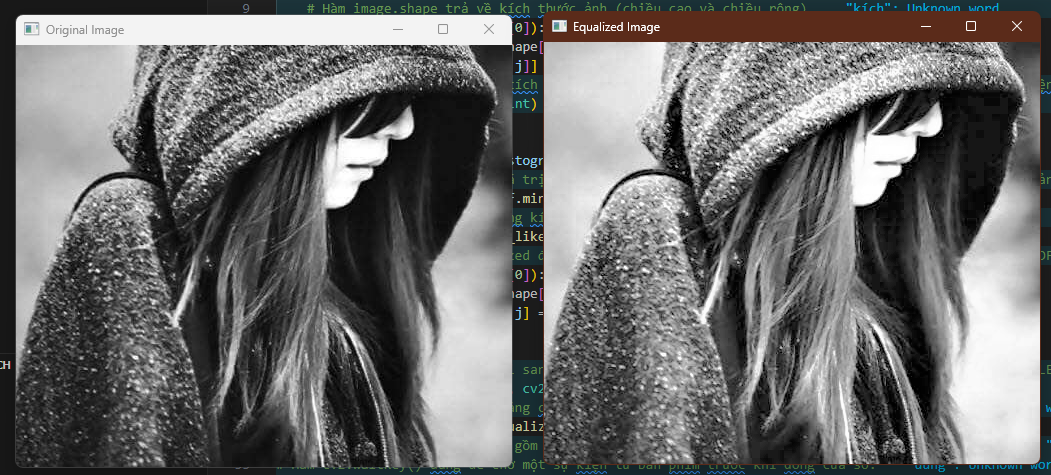
Hình 2.3.1. Hình ảnh thể hiện kết quả của đoạn code của cách 1 trong mục 2.1 của chương 2.

+ Kết luận: Thấy được sự cải thiện trong phân bố mức xám của ảnh sau khi áp dụng thuật toán cân bằng lược đồ.

* *Cách 2:*

+ Dữ liệu: Ảnh ‘input.jpg’.

+ Kết quả: Hai cửa sổ hiển thị, một là ảnh gốc và một là ảnh sau khi cân bằng lược đồ.



Hình 2.3.2. Hình ảnh thể hiện kết quả của đoạn code của cách 2 trong mục 2.1 của chương 2.

+ Kết luận: Thấy được sự cải thiện trong phân bố mức xám của ảnh sau khi áp dụng thuật toán cân bằng lược đồ.

**TÀI LIỆU THAM KHẢO**

* **Việt Nam:**

1. Chúng ta có thể tìm hiểu thêm tại: <https://docs.python.org/3/library/statistics.html>

[Page Not Found - GeeksforGeeks](https://www.geeksforgeeks.org/python-statistics/?fbclid=IwAR1nKTEIsTGO78koW3mYWjPWzqI68rhBfMTfS1y5U68oobGZdVNiLyGFBLQ)  
[Python statistics Module (w3schools.com)](https://www.w3schools.com/python/module_statistics.asp)

[Module statistics trong Python - Học lập trình Python - Viettuts](https://viettuts.vn/python/mo-dun-statistics-trong-python)