

Trường Đại Học Công Nghệ Thông Tin - ĐHQG

KHOA: KHOA HỌC MÁY TÍNH



UIT

ĐỒ ÁN MÔN HỌC

XỬ LÝ ẢNH

Đề tài:

SEAM CARVING

Giảng viên hướng dẫn : Mai Tiến Dũng

Sinh viên thực hiện : Lê Anh Minh (18521098)

Trần Lê Duy (18520674)

Nguyễn Huỳnh Nhi (18521204)

Lớp : CS406.L11.KHCL

Khóa : 2018

TP.Hồ Chí Minh, ngày 2 tháng 1 năm 2022

LỜI CẢM ƠN

Để hoàn thành được đề án môn học lần này 1 cách thành công tốt đẹp, em xin trân trọng cảm ơn tập thể lớp và giảng viên thầy Mai Tiến Dũng (phụ trách bộ môn Xử lý ảnh, mã môn học: CS406.L11.KHCL) đã hỗ trợ nhóm em trong suốt thời gian học tập.

Do kiến thức còn nhiều hạn chế và khả năng tiếp thu thực tế còn nhiều bỡ ngỡ và chưa hoàn hảo nên báo cáo sẽ còn nhiều thiếu sót, kính mong sự góp ý và giúp đỡ từ thầy Mai Tiến Dũng.

Nhóm em xin chân thành cảm ơn!

This image shows a full page of white paper with horizontal dotted lines. The lines are evenly spaced and run across the width of the page, providing a guide for handwriting or typing. There are no margins, text, or other markings on the page.

MỤC LỤC

| | |
|---|----|
| Phần 1: Tổng quan..... | 1 |
| Phần 2: Tổng quan về thuật toán Seam Carving..... | 3 |
| Phần 3: Tìm hiểu thuật toán | 5 |
| Phần 4: Năng lượng chuyển tiếp | 6 |
| Phần 5: Chèn đường nối..... | 9 |
| Phần 6: Loại bỏ đối tượng..... | 13 |
| Phần 7: Kết quả từ hình ảnh gốc | 15 |
| Phần 8: Demo chương trình | 18 |
| Phần 9: Tài liệu tham khảo..... | 24 |
| KẾT LUẬN | 25 |

Phần 1: Tổng quan:

1.1 Lý do chọn đề tài:

Ngày nay, với sự phát triển của công nghệ, cùng với đó là sự đa dạng và linh hoạt của nhiều loại thiết bị như máy tính bảng, laptop, điện thoại thông minh đã và đang đặt ra yêu cầu cho truyền thông kỹ thuật số. Ví dụ như các nhà phát triển phần mềm, trang web đã thiết kế nhiều phiên bản khác nhau cho nội dung web, hoặc những bố cục khác nhau cho các thiết bị khác nhau. Từ đó, các trang web html hay các trang tương tự đã có thể có được những thay đổi linh hoạt cho bố cục trang web hoặc văn bản trang web.

Tuy nhiên cho tới nay, tuy là một trong những yếu tố quan trọng trong truyền thông kỹ thuật số, ảnh vẫn là bị giữ nguyên kích thước, và chưa thể biến đổi để phù hợp với các layout một cách tự động. Hoặc nhiều trường hợp khác, các bức ảnh cần được để ở kích cỡ khác nhau để phù hợp cho điện thoại, máy tính, ti vi.... Từ đó sinh ra bài toán Image Retargeting.

1.2 Mô tả bài toán:

Mục tiêu của thuật toán khắc đường may này là thực hiện thay đổi kích thước hình ảnh nhận biết nội dung. Điều này cho phép thay đổi kích thước hình ảnh mà không làm mất nội dung có ý nghĩa do cắt xén hoặc chia tỷ lệ.

Ý tưởng là xác định vị trí các đường nối tối ưu của hình ảnh, các đường dẫn pixel được kết nối đi từ trên xuống dưới hoặc từ trái sang phải, để loại bỏ hoặc chèn trong khi vẫn giữ được độ chân thực của hình ảnh.

Hơn nữa, thao tác với bản đồ năng lượng gradient mô tả mức độ tối ưu của đường may cho phép thực hiện các chức năng như loại bỏ vật thể. Kỹ thuật được mô tả

dưới đây là một triển khai của thuật toán được trình bày trong (Avidan và Shamir) .

Ngoài ra, một cải tiến đối với việc loại bỏ đường nối được thực hiện bằng cách sử dụng năng lượng chuyển tiếp và thuật toán được mở rộng để chạy trên video dựa trên (Avidan, Rubinstein và Shamir) .

Hình ảnh bên dưới là phần thay đổi kích thước nhận biết nội dung của hình ảnh bên trên.



Phần 2: Tổng quan về thuật toán Seam Carving:

1. Tính toán bản đồ năng lượng

Đối với mỗi kênh màu, năng lượng được tính bằng cách cộng giá trị tuyệt đối của gradient theo hướng x với giá trị tuyệt đối của gradient theo hướng y.

Năng lượng cho tất cả các kênh màu được tổng hợp thành một hình ảnh 2D để tạo bản đồ năng lượng.

2. Tìm đường nối tối thiểu từ cạnh trên xuống dưới

Đầu tiên, một ma trận chi phí tích lũy phải được xây dựng bằng cách bắt đầu từ cạnh trên cùng và lặp lại qua các hàng của bản đồ năng lượng. Giá trị của một pixel trong ma trận chi phí tích lũy bằng với giá trị pixel tương ứng của nó trong bản đồ năng lượng được thêm vào mức tối thiểu của ba cận kề trên cùng của nó (trên cùng bên trái, trên cùng giữa và trên cùng bên phải) từ ma trận chi phí tích lũy.

Giá trị của hàng trên cùng (rõ ràng là không có hàng nào phía trên) của ma trận chi phí tích lũy bằng với bản đồ năng lượng. Các điều kiện biên trong bước này cũng được xem xét.

Nếu một pixel lân cận không khả dụng do cạnh trái hoặc cạnh phải, nó chỉ đơn giản là không được sử dụng trong phép tính láng giềng tối thiểu.

Sau đó, đường may tối thiểu được tính bằng cách kéo ngược từ mép dưới lên mép trên. Đầu tiên, pixel giá trị nhỏ nhất ở hàng dưới cùng của ma trận chi phí tích lũy được đặt. Đây là pixel dưới cùng của đường may tối thiểu. Sau đó, đường nối được truy ngược lên hàng trên cùng của ma trận

chi phí tích lũy bằng cách sau. Các tọa độ đường may tối thiểu được ghi lại.

Bước này được thực hiện với lập trình động. Một cải tiến bổ sung cho bước này mang lại giá trị năng lượng chính xác hơn được mô tả trong phần năng lượng chuyển tiếp.

3. Loại bỏ đường may tối thiểu từ cạnh trên xuống dưới

Tọa độ đường may tối thiểu từ Bước 2 sau đó được sử dụng để loại bỏ đường may tối thiểu.

Tất cả các pixel trong mỗi hàng sau khi xóa pixel sẽ được chuyển qua một cột.

Cuối cùng, chiều rộng của hình ảnh đã được giảm chính xác một pixel.

4. Lặp lại các bước 1 - 3 cho đến khi số lượng đường nối mong muốn được loại bỏ

Quá trình này được lặp lại cho đến khi số lượng đường nối tối thiểu mong muốn được loại bỏ.

Bản đồ năng lượng cần được tính toán lại mỗi khi loại bỏ một đường nối.

5. Lặp lại các bước 1 - 4 cho cạnh trái sang phải

Bước 1-4 ban đầu mô tả việc giảm chiều rộng của hình ảnh. Có thể dễ dàng sử dụng cùng một đoạn mã để giảm chiều cao của hình ảnh bằng cách đơn giản lấy sự chuyển vị của hình ảnh đầu vào.

Phần 3: Tìm hiểu thuật toán:

Sử dụng thuật toán Dynamic Programing để có thể tìm những đường seam có giá trị nhỏ nhất để xóa hoặc thêm chúng, mục đích là tạo ra ảnh mới. Nhóm sẽ tính toán các giá trị năng lượng bằng cách xóa đi những pixel có giá trị thấp. Những pixel này sẽ liên kết với nhau sao cho giá trị của chúng sẽ là nhỏ nhất.

Có 3 trường hợp để có thể chọn được giá trị pixel nào tiếp theo sẽ được tính toán tiếp theo để thêm vào đường seam.

Trường hợp 1: Pixel kế tiếp của đường seam cần xóa nằm sát bên trái

| | |
|-------|-----------|
| a_i | a_{i+1} |
| b_i | b_{i+1} |

Ở pixel b_i ta cần tìm pixel ở hàng trên có giá trị nhỏ nhất ở hàng trên liền kề với nó để tính được giá trị kết quả mới vào b_i . Có 2 trường hợp

$$b_i \begin{cases} b_i = b_i + a_i \\ b_i = b_i + a_{i+1} \end{cases}$$

Trường hợp 2: Pixel kế tiếp của đường seam cần xóa nằm giữa

| | | |
|-------|-----------|-----------|
| a_i | a_{i+1} | a_{i+2} |
| b_i | b_{i+1} | b_{i+2} |

Ở pixel b_{i+1} ta cần tìm pixel ở hàng trên có giá trị nhỏ nhất ở hàng trên liền kề với nó để tính được giá trị kết quả mới vào b_{i+1} . Có 3 trường hợp

$$b_{i+1} \begin{cases} b_{i+1} = b_{i+1} + a_i \\ b_{i+1} = b_{i+1} + a_{i+1} \\ b_{i+1} = b_{i+1} + a_{i+2} \end{cases}$$

Trường hợp 3: Pixel kế tiếp của đường seam cần xóa nằm sát bên phải

| | |
|-------|-----------|
| a_i | a_{i+1} |
| b_i | b_{i+1} |

Ở pixel b_{i+1} ta cần tìm pixel ở hàng trên có giá trị nhỏ nhất ở hàng trên liền kề với nó để tính được giá trị kết quả mới vào b_{i+1} . Có 2 trường hợp

$$b_{i+1} \begin{cases} b_{i+1} = b_{i+1} + a_i \\ b_{i+1} = b_{i+1} + a_{i+1} \end{cases}$$

Sau khi tính toán ta dùng phương pháp Backtracking để tìm đường seam nhỏ nhất dựa trên các trường hợp trên.

Phần 4: Năng lượng chuyển tiếp:

Việc tăng cường năng lượng phía trước được thực hiện bằng cách tính đến năng lượng được tạo ra từ các cận kề mới sau khi đường nối được loại bỏ. Thuật toán ban đầu được sửa đổi bằng cách thêm năng lượng chuyển tiếp bổ sung này trong quá trình tạo ma trận chi phí tích lũy.

Khi quyết định cận kề trên cùng (trên cùng bên trái, trên cùng giữa, trên cùng bên phải) trong ma trận chi phí tích lũy có giá trị năng lượng nhỏ nhất, năng lượng chuyển tiếp từ các láng giềng mới tương ứng sẽ được thêm vào giá trị chi phí tích lũy của từng cận kề trên cùng.

Các lợi ích của việc sử dụng năng lượng chuyển tiếp có thể được nhìn thấy dưới đây trong cách bảo tồn hình dạng góc cạnh của các ngọn núi.

Đầu vào (trên cùng)

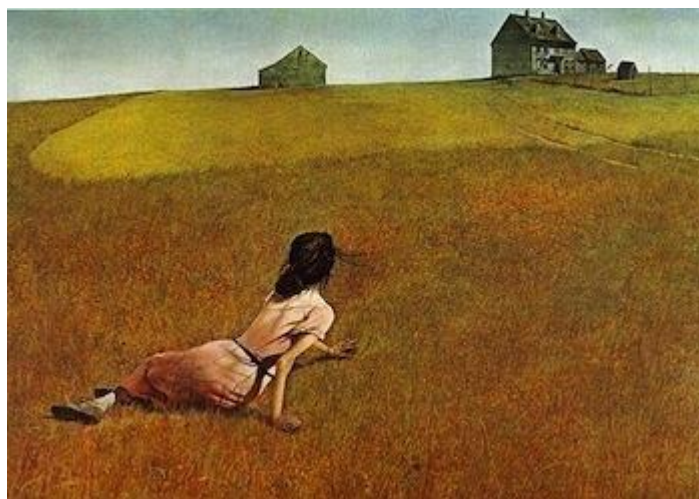


Năng lượng chuyển tiếp (trái) so với Không năng lượng chuyển tiếp (phải)

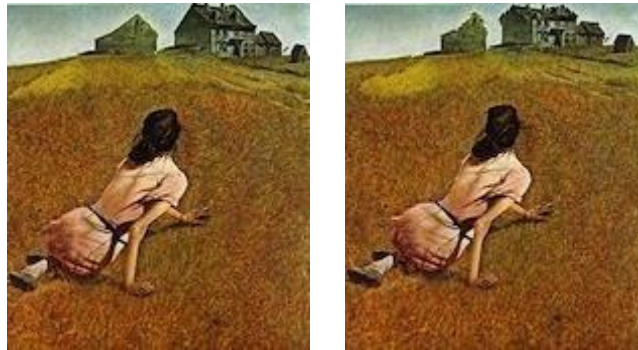


Ở đây, hình ảnh với năng lượng phía trước có một ngôi nhà trông đẹp hơn nhiều.

Đầu vào (trên cùng)



Năng lượng chuyển tiếp (trái) so với Không năng lượng chuyển tiếp (phải)



Một lần nữa, hình dạng góc cạnh của những ngọn núi được bảo tồn.

Đầu vào (trên cùng)



Năng lượng chuyển tiếp (trái) so với Không năng lượng chuyển tiếp (phải)**Phần 5: Chèn đường nối:**

Chèn đường may có thể được coi là thực hiện ngược lại việc loại bỏ đường may. Nếu n đường may được chèn, thì việc loại bỏ đường may trước tiên phải được thực hiện cho n đường may.

Tọa độ và thứ tự của các đường nối tối thiểu này được ghi lại trong quá trình loại bỏ. Hình ảnh gốc không thực sự được sửa đổi trong quá trình này, vì mục đích chỉ là để tìm tọa độ và thứ tự của các đường nối tối thiểu.

Cuối cùng, các tọa độ được sử dụng để chèn các đường nối mới vào hình ảnh ban đầu theo đúng thứ tự mà chúng đã được loại bỏ.

Giá trị pixel của các đường nối mới được tính từ giá trị trung bình của các cận kề trên / dưới hoặc trái / phải.

Một số ví dụ về chèn đường may được hiển thị bên dưới.

Đầu vào (trên) so với Đã thay đổi kích thước (dưới)



Đầu vào (trên cùng) so với Đã thay đổi kích thước (dưới cùng)



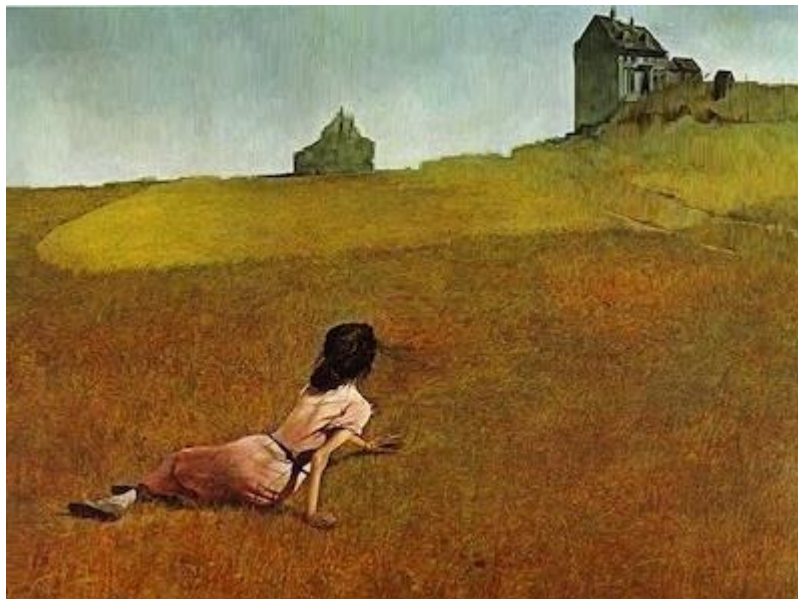
Điều quan trọng cần lưu ý là **không** nên sử dụng năng lượng chuyển tiếp để chèn đường may.

Điều này là do năng lượng chuyển tiếp từ các nước láng giềng mới không liên quan đến hình ảnh mở rộng.

Trên thực tế, thuật toán ban đầu đủ để ước tính năng lượng từ các cận kề mới trong việc chèn đường nối, vì đường may mới chỉ là mức trung bình.

So sánh lợi ích của năng lượng chuyển tiếp và không có năng lượng chuyển tiếp trong chèn đường may được hiển thị bên dưới. 50 pixel đã được chèn vào mỗi kích thước.

Năng lượng chuyển tiếp (trên) so với Không năng lượng chuyển tiếp (dưới)





Phần 6: Loại bỏ đối tượng:

Loại bỏ đối tượng được thực hiện bằng cách đầu tiên tạo một mặt nạ nhị phân trong quá trình xử lý trước đối tượng cần loại bỏ.

Đầu tiên, diện tích của vùng bị che được đo để có thể quyết định xem việc loại bỏ vật thể đó hiệu quả hơn bằng cách loại bỏ các đường nối từ trên xuống dưới hoặc các đường nối từ trái sang phải.

Sau đó, khi tạo bản đồ năng lượng của hình ảnh, các khu vực bên dưới vùng bị che có giá trị âm rất cao.

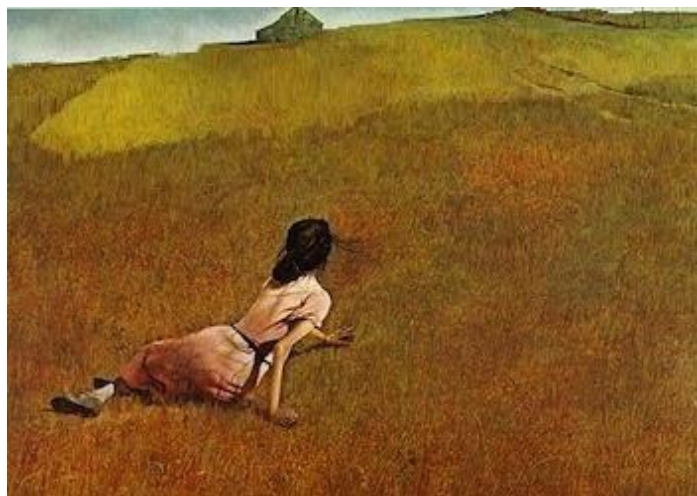
Điều này đảm bảo rằng đường may tối thiểu sẽ được định tuyến qua vùng được che. Loại bỏ đường may tối thiểu sau đó được thực hiện như bình thường. Sự khác biệt duy nhất là khi loại bỏ đường may tối thiểu khỏi một hình ảnh, đường may tối thiểu tương tự cũng phải được loại bỏ khỏi mặt nạ để việc tính toán năng lượng của bước tiếp theo được chính xác.

Sau khi vùng bị che đã được loại bỏ hoàn toàn, chèn đường nối được sử dụng để trả lại hình ảnh cho nó 's kích thước ban đầu.

Năng lượng chuyển tiếp được sử dụng trong quá trình loại bỏ đường may nhưng không được sử dụng trong quá trình chèn đường may.

Dưới đây là một ví dụ về loại bỏ đối tượng.

Đầu vào / Mặt nạ (trên cùng) so với Đối tượng đã xóa (dưới cùng)



Phần 7: Kết quả từ hình ảnh gốc:

Đầu vào (trên) so với Đã thay đổi kích thước (dưới)



Đầu vào (trên cùng) so với Đã thay đổi kích thước (dưới cùng)



Trường hợp thất bại:

Đây rõ ràng là một trường hợp thất bại, bởi vì Larry Bird và Magic Johnson đã trở nên méo mó thay vì nên.

Điều này có thể là do một số khu vực trên da và đồng phục của các cầu thủ có năng lượng thấp hơn so với các khu vực đám đông ở hậu cảnh. Gradient ở những khu vực này tương đối mượt mà hơn so với trong đám đông.

Ngoài ra, thuật toán thực hiện công việc kém trong việc bảo toàn hình dạng của chân của họ, đặc biệt là nơi các chân chồng lên nhau.

Một cải tiến thuật toán có thể có sẽ là một cách nào đó để nói với thuật toán đó rằng những khu vực này là quan trọng để thay vào đó, đám đông sẽ bị loại bỏ.

Đầu vào (trái) so với Đã thay đổi kích thước (phải)



Phần 8: Demo chương trình:

Với demo này chúng em xin phép thực nghiệm với việc tăng giảm ảnh đi 100px với các hình thức tăng (giảm) chiều dọc và chiều ngang ảnh.

Ảnh gốc:



Seam Carving theo chiều dọc:



Seam Carving theo chiều ngang:



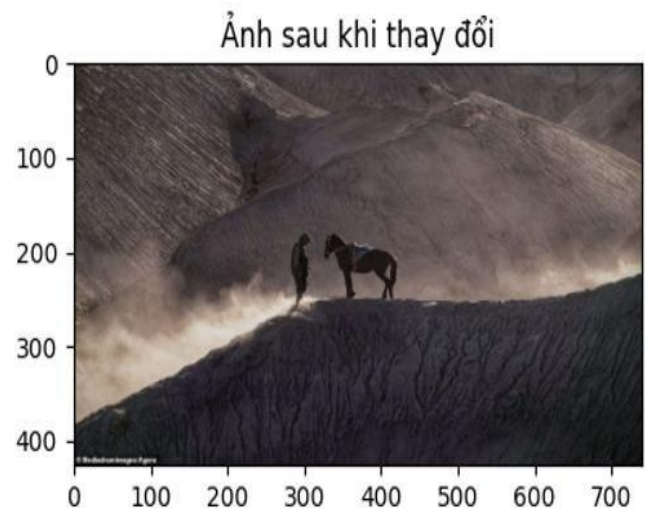
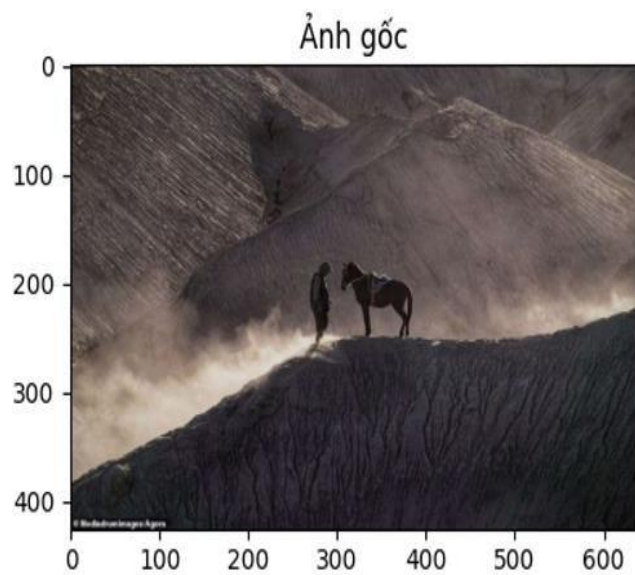
Tăng chiều dọc ảnh lên 100px:



Tăng chiều ngang ảnh lên 100px:



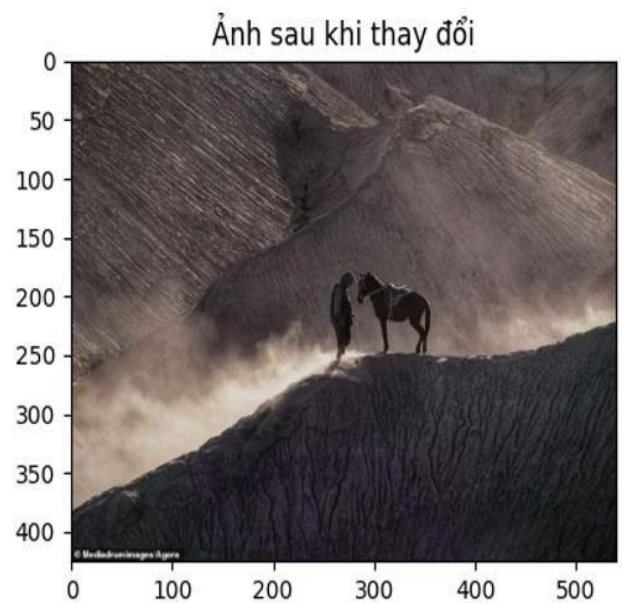
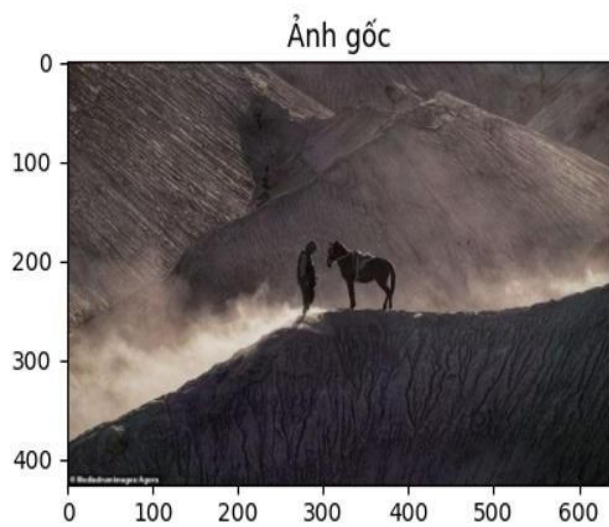
So sánh:



Giảm chiều ngang ảnh đi 100px:



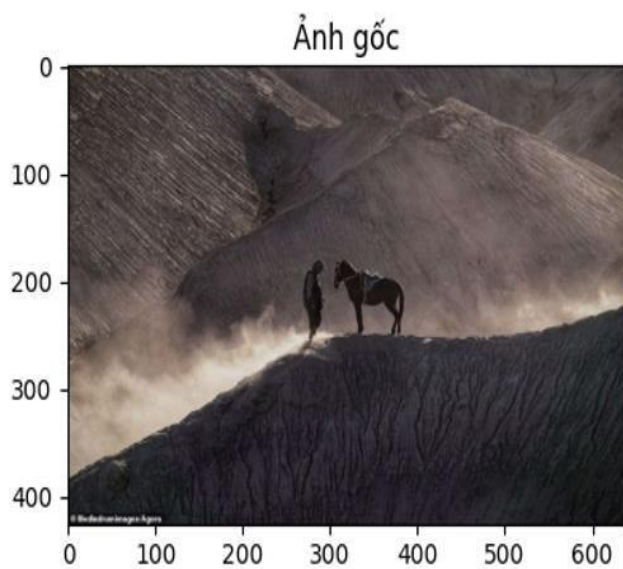
So sánh:



Giảm chiều dọc ảnh đi 100px:



So sánh :



Phần 9: Tài liệu tham khảo:

1. https://karthikkaranth.me/blog/implementing-seam-carving-with-python/?fbclid=IwAR20jeK8HAIuQOO3K4nBC28gWWYEcqX-jbZN10qs_Zzz6P5TPcZ4D9GOedk
2. <https://vnoi.info/wiki/cs/imageprocessing/Seam-Carving.md>
3. [Avidan, Shai; Shamir, Ariel \(July 2007\). "Seam carving for content-aware image resizing | ACM SIGGRAPH 2007 papers".](#)
4. https://en.wikipedia.org/wiki/Seam_carving

KẾT LUẬN

Trong báo cáo này, chúng em đã tìm hiểu và thiết kế ra được một chương trình để minh họa các thuật toán trong môn Xử lý ảnh và Ứng dụng.

Qua đề tài lần này có thể giúp chúng em nắm vững thêm những kiến thức để sau này đi làm hoặc học sâu hơn.

Xin chân thành cảm ơn thầy Mai Tiến Dũng đã giúp đỡ chúng em rất nhiều để hoàn thành đề tài này.