

ĐẠI HỌC QUỐC GIA TP. HCM
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN



BÁO CÁO ĐỒ ÁN CUỐI KÌ
XÓA ĐỐI TƯỢNG SỬ DỤNG SEAM CARVING

Giáo viên hướng dẫn: TS. Nguyễn Vinh Tiệp

Nhóm thực hiện:

- 1. Nguyễn Nhật Trường - 20522087**
- 2. Lại Chí Thiện - 20520309**
- 3. Lê Thị Phương Vy - 20520355**
- 4. Lê Trương Ngọc Hải - 20520481**

TP. HỒ CHÍ MINH, 2022

MỤC MỤC

TỔNG QUAN ĐỒ ÁN.....	1
Chương 1. Giới thiệu đề tài.....	1
1.1 Tình hình thực tại	1
1.2 Giới thiệu bài toán.....	1
Chương 2. Sơ lược về Seam Carving.....	2
2.1 Energy map	2
2.2 Seams	3
2.3 Đường seam tối ưu	4
Chương 3. Một cách tiếp cận của bài toán xóa đối tượng	4
Chương 4. Thử nghiệm	6
Chương 5. Đánh giá	8
Chương 6. Kết luận và hướng phát triển trong tương lai.....	10
Chương 7. Cấu trúc đồ án	10

DANH MỤC HÌNH

Hình 1: Original image và energy map (trái và phải).....3

Hình 2: Minh họa đường seam3

Hình 3: Minh họa về việc tạo ra removal masking5

Hình 4: Minh họa về việc tạo ra protective masking6

Hình 5: Ảnh trước và sau khi thử nghiệm xóa đối tượng (chữ PCCC, cửa thông hơi)
.....8

Hình 6: Minh họa về việc xóa đối tượng không có mask bảo vệ8

TỔNG QUAN ĐỒ ÁN

Trong chương này, chúng tôi giới thiệu chung về đồ án, đầu tiên giới thiệu đề tài và giới thiệu sơ lược seam carving, tiếp đến là một cách tiếp cận của bài toán xóa đối tượng và kết quả nghiệm thu cơ bản. Phần tóm tắt từng chương trong đồ án được trình bày ở chương cuối cùng.

Chương 1. Giới thiệu đề tài

1.1 Tình hình thực tại

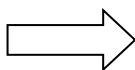
Trong những năm gần đây, dưới sự bùng nổ của các mạng xã hội, nhu cầu chia sẻ các thông tin đa phương tiện như hình ảnh, video càng tăng lên một cách chóng mặt. Kéo theo đó là sự phát triển của các công cụ chỉnh sửa hình ảnh video ngày càng hiện đại, tân tiến hơn, giải quyết được nhiều vấn đề hơn. Trong đó, thao tác xóa vật thể trong ảnh trở thành một trong những thao tác được sử dụng nhiều nhất.

1.2 Giới thiệu bài toán

Xóa đối tượng trong hình ảnh là một bài toán xử lý ảnh thuộc lĩnh vực thị giác máy tính, bài toán nhận đầu vào là một hình ảnh và thông tin về đối tượng cần xóa trong hình ảnh vừa được đưa vào. Trong bối cảnh phát triển mạng mẽ của công nghệ thông tin nói chung và thị giác máy tính nói riêng, các phần mềm công cụ chỉnh sửa hình ảnh, video đã và đang trang bị cho mình những công nghệ xóa đối tượng trong ảnh rất tốt, cho ra kết quả rất vượt trội so với trước đây. Để có được những công cụ với hiệu suất vượt trội cần phải dựa vào thuật toán nền tảng tốt, từ đó phát triển thành những ứng dụng một cách hoàn chỉnh.



input



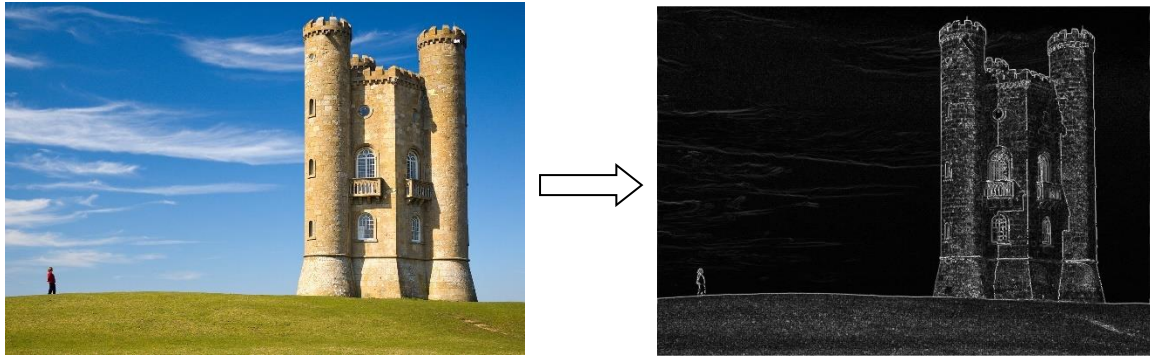
output

Chương 2. Sơ lược về Seam Carving

Seam carving là một thuật toán thay đổi kích thước của hình ảnh nhưng vẫn giữ được các nội dung quan trọng, nó hoạt động bằng cách liên tục xác định một đường seam có chứa thông tin ít quan trọng trong hình ảnh và sau đó cắt các hình ảnh đó ra khỏi bức ảnh một cách có chọn lọc. Các đường seam có thể nằm dọc hoặc nằm ngang. Đường seam dọc là một đường dẫn gồm các pixel gồm các pixel nối từ trên xuống dưới trong một hình ảnh với một pixel trong mỗi hàng được minh họa như hình 2. Các đường seam ngang cũng tương tự như vậy nhưng sẽ là các đường nối từ phải sang trái. Hàm năng lượng (energy map) sẽ đánh giá một pixel bằng cách đo độ tương phản của nó với các pixel lân cận.

2.1 Energy map

Đầu tiên chúng ta sẽ cần số liệu về những vùng nào trong hình ảnh là không quan trọng. Mục đích của hàm năng lượng (energy function) là gán cho mỗi pixel một giá trị $e(i, j)$ để chỉ ra tầm quan trọng của nó trong hình ảnh. Các pixel không có sự biến thiên lớn so với các pixel xung quanh nó có giá trị năng lượng thấp, ngược lại các pixel có giá trị biến thiên cao so với các pixel xung quanh (ví dụ như cạnh vật thể) có giá trị năng lượng cao. Ta cần định nghĩa hàm năng lượng để tính năng lượng điểm ảnh từ các điểm ảnh của ảnh gốc. Ta có thể tính năng lượng của ảnh thông qua đạo hàm của từng điểm ảnh theo các hướng $e_I = \left| \frac{\delta I}{\delta x} \right| + \left| \frac{\delta I}{\delta y} \right|$.



Hình 1: Original image và energy map (trái và phải)¹

2.2 Seams

Xác định một đường seam dọc là một đường nối các pixel trong hình ảnh từ trên xuống dưới, mỗi pixel trong đó thuộc một hàng riêng biệt của hình ảnh. Tương tự, một đường seam ngang là một đường từ phải sang trái, mỗi pixel trong đó thuộc một cột riêng biệt của hình ảnh. Trong cả hai loại đường seam nếu hai pixel nằm cạnh nhau thì pixel này phải thuộc 8 pixel xung quanh của pixel kia. Khi xóa một đường seam, các pixel di chuyển lên trên (hoặc sang trái) để bù đắp cho đường vừa xóa. Ngược lại khi chèn thêm một đường seam, các pixel di chuyển xuống dưới (hoặc sang phải) để dành chỗ cho đường được thêm vào. Chúng ta có thể thay đổi kích thước ảnh ban đầu bằng cách xóa hoặc thêm các đường seam.



Hình 2: Minh họa đường seam²

¹ <https://andrewdcampbell.github.io/seam-carving>

² <https://karthikkaranth.me/blog/implementing-seam-carving-with-python/>

Việc xác định đường seam nào được chọn để xóa hoặc thêm vào sẽ dựa vào energy map. Đường seam được chọn là đường seam tối ưu, một đường seam được gọi là tối ưu khi năng lượng của nó là nhỏ nhất. Năng lượng của một đường seam sẽ bằng tổng năng lượng các pixel của nó.

2.3 Đường seam tối ưu

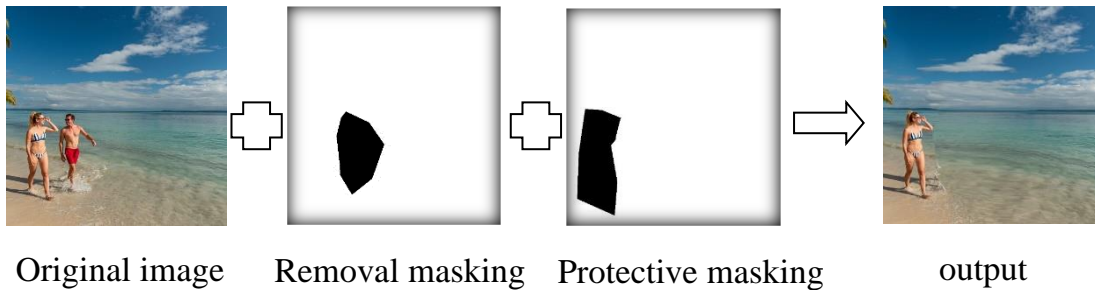
Chúng ta có thể tìm đường seam tối ưu bằng phương pháp quy hoạch động. Bước đầu tiên ta lướt qua hình ảnh từ hàng thứ hai đến hàng cuối cùng và tính toán năng lượng tối thiểu tích lũy M cho tất cả các đường seam được kết nối cho mỗi entry (i, j) . Cách để tìm được kết quả tối ưu cho bài toán khi sử dụng chiến lược quy hoạch động:

1. Tìm đường seam tối ưu từ biên trên của ảnh đến mỗi điểm ảnh (i, j) .
 - Gọi $M[i, j]$ là giá trị năng lượng nhỏ nhất đi từ biên trên của ảnh đến điểm ảnh (i, j) .
 - $M[1, j] = e(1, j)$ với $e(i, j)$ là năng lượng điểm ảnh tại (i, j) .
 - $M[i, j] = \min(M[i - 1, j - 1], M[i - 1, j + 1]) + e(i, j)$.
2. Ở biên dưới của ảnh, ta tìm đường seam tối ưu (tổng giá trị năng lượng thấp nhất thông qua bảng phương án M) và đi ngược về để tìm đường đi tối ưu.

Chương 3. Một cách tiếp cận của bài toán xóa đối tượng

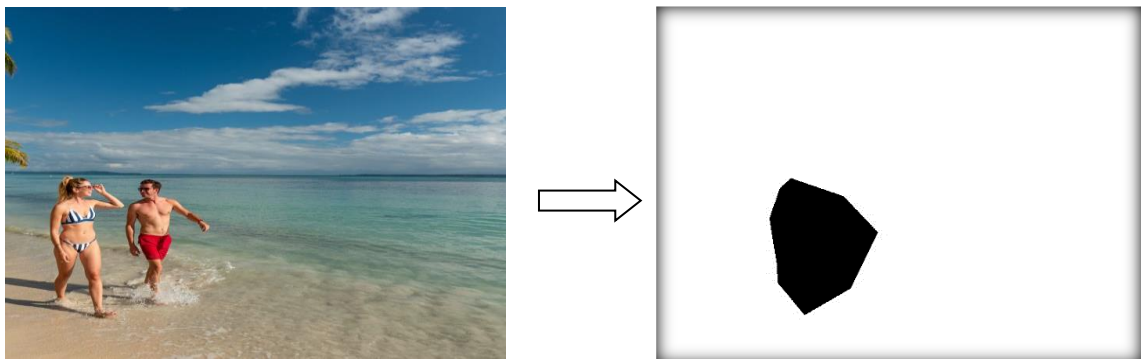
Bài toán xóa đối tượng được thực hiện bằng việc trước tiên phải tạo ra một binary mask trong quá trình xử lý trước khi đối tượng bị loại bỏ. Đầu tiên, khu vực mask được đo lường để có thể quyết định liệu có tốt hơn để loại bỏ đối tượng bằng các đường seam từ trên xuống dưới hay từ trái sang phải. Sau đó, khi tạo energy map của hình ảnh, các khu vực dưới vùng mask có trọng số với giá trị âm rất cao. Điều này sẽ dẫn đến đường seam tối ưu sẽ được định tuyến qua vùng mask. Loại bỏ đường seam tối ưu và sau đó được thực hiện như bình thường. Sự khác biệt duy nhất là khi loại bỏ các đường seam tối ưu khỏi hình ảnh thì đường seam tối ưu cũng bị loại bỏ khỏi mask để tính toán năng lượng của bước tiếp theo. Sau khi vùng cần

xóa được loại bỏ hoàn toàn, sẽ thực hiện chèn đường seam được sử dụng để đưa hình ảnh trở lại kích thước ban đầu.



Seam removal

Chúng tôi sẽ đánh dấu đối tượng mục tiêu sẽ bị loại bỏ bằng việc tạo ra removal masking. Sau đó các đường seams được loại bỏ khỏi hình ảnh cho đến khi tất cả pixel được đánh dấu là biến mất. Ở removal masking chúng ta sẽ cung cấp cho vùng này một mức năng lượng âm lớn. Tiếp đến, liên tục loại bỏ các đường seam cho đến khi removal masking được loại bỏ ra khỏi bức ảnh.

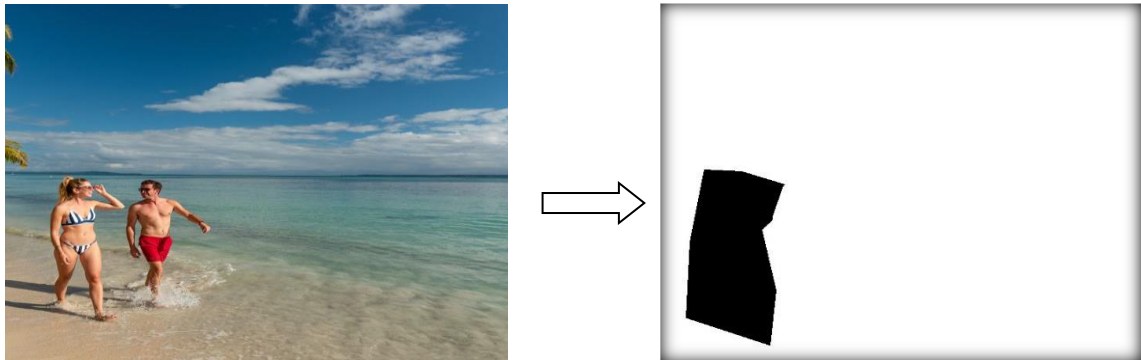


Hình 3: Minh họa về việc tạo ra removal masking

Protective Masking

Chúng tôi muốn chỉ định các vùng trong hình không nên được sửa đổi. Do đó việc tạo ra một protective masking là thực sự cần thiết. Điều này có thể thực hiện bằng cách cung cấp một binary mask cho thuật toán xóa đối tượng nhằm bảo vệ đối tượng mà chúng ta mong muốn. Ngược lại với removal masking, các khu vực màu

đen của mặt nạ có thể được cung cấp các giá trị năng lượng dương trong energy map tương ứng để ngăn chặn mạnh mẽ đường seam tối ưu đi qua khu vực này.



Hình 4: Minh họa về việc tạo ra protective masking

Mô tả thuật toán xóa đối tượng sử dụng seam carving

Bước 1: Giảm năng lượng tại removal masking (giảm xuống bé hơn 0).

Bước 2: Tăng năng lượng tại protective masking.

Bước 3: Tính toán đường seam theo chiều rộng và chiều cao của bức ảnh và năng lượng của chúng.

Bước 4: Dựa vào năng lượng để quyết định xóa đường seam nào.

Bước 5: Quay lại Bước 3 đến khi tất cả các pixel được đánh dấu bằng removal masking biến mất.

Trên đây chỉ là ý tưởng thuật toán, các thông số sẽ được thể hiện rõ hơn chương 4.

Chương 4. Thử nghiệm

Để thuận tiện cho quá trình tạo removal masking và protective masking chúng tôi có thêm file mask.py giúp hỗ trợ vẽ các polygon. Cụ thể các vẽ các polygon như nhau:

- Nhấn chuột trái để tạo các điểm tạo nên các polygon: các điểm mới tạo sẽ được nối với điểm trước đó.
- Khi hoàn thành 1 polygon nhấn chuột phải: điểm cuối cùng sẽ được nối với điểm đầu tiên.

- Khi đã hoàn thành mask nhấn phím q để hoàn thành.

Trong class Mask của mask.py chúng tôi có thêm hàm Deleted_union giúp xóa phần mask giao giữa removal masking và protective masking trong removal masking (truyền thêm tham số `rm = True`) hoặc protective masking.

Quá trình xóa được diễn ra như sau:

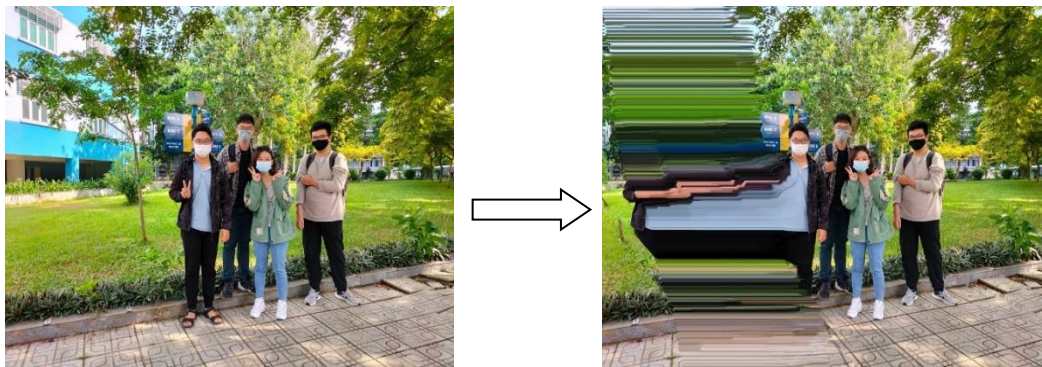
- Năng lượng tại những điểm được đánh dấu trên removal masking sẽ được nhân với $-(\text{năng lượng tại điểm đó})^2$
- Năng lượng tại những điểm trên sẽ tiếp tục được trừ cho $(\text{năng lượng tại điểm đó})^2$
- Năng lượng tại những điểm được đánh dấu trên protective masking sẽ được tăng thêm 255
- Ta tính toán đường seam tối ưu và năng lượng của nó theo chiều dọc và ngang của bức ảnh sao đó xóa đi đường seam có năng lượng nhỏ hơn, quá trình này được lặp lại đến khi tất cả các pixel được đánh dấu trong removal masking được xóa.

Quá trình resize ảnh được diễn ra như sau:

- Copy hình ảnh ban đầu
- Chọn đường seam tối ưu theo chiều dọc hình ảnh copy sau đó xóa đường seam đó hình ảnh copy và thêm đường seam đó vào ảnh gốc, quá trình này được lặp lại đến khi chiều dọc ảnh gốc bằng với kích thước trước khi xóa vật thể.
- Tương tự với chiều ngang, chọn đường seam tối ưu theo chiều ngang hình ảnh copy sau đó xóa đường seam đó hình ảnh copy và thêm đường seam đó vào ảnh gốc, quá trình này được lặp lại đến khi chiều dọc ảnh gốc bằng với kích thước trước khi xóa vật thể.



Hình 5: Ảnh trước và sau khi thử nghiệm xóa đối tượng (chữ PCCC, cửa thông hơi)



Hình 6: Minh họa về việc xóa đối tượng không có mask bảo vệ

Chương 5. Đánh giá

- **Nhận xét kết quả:**

Dựa vào kết quả thử nghiệm, chúng tôi nhận thấy rằng kết quả xóa đối tượng được thực hiện khá tốt trên nhiều ảnh khác nhau. Bên cạnh đó, cũng còn một số hạn chế cần được giải quyết như protective masking cần phải được bảo vệ tốt hơn, đôi khi protective masking được sử dụng nhưng vẫn bảo vệ không được tốt do đường seam tối ưu bắt buộc đi qua khu vực này. Ngoài ra, việc resize chưa tốt cũng là yếu tố ảnh hưởng trực tiếp đến hiệu quả của việc xóa đối tượng.

- **Điểm mạnh:**

Bằng việc loại bỏ những pixel có năng lượng thấp, hay còn có thể nói là những pixel kém quan trọng so với những pixel khác, thuật toán Seam Carving hoàn toàn có thể bảo toàn được kích thước và hình dạng của vật thể “quan trọng” trong lúc resize những vùng “ít quan trọng”.

Khác với những thuật toán trước, Seam Carving suy xét “mức độ quan trọng” của từng pixel để có thể tìm được đường seam tối ưu nhất rồi từ đó giải quyết vấn đề mình đang hướng tới thay vì áp dụng các kỹ thuật kéo dài (stretching) hoặc cắt xén (cropping) – những kỹ thuật thô sơ gây mất mát hoặc biến dạng vật thể trong ảnh.

- **Điểm yếu:**

Thuật toán rõ ràng quá phụ thuộc vào các phương pháp phát hiện vật thể hoặc các thao tác thủ công từ người dùng (tạo protective masking), tức là, nếu không có protective masking, vật thể quan trọng sẽ dễ bị tác động bởi thuật toán, gây ra biến dạng, đứt gãy hoặc lờm chờm. Thêm vào đó, khi đường seam bắt buộc phải đi qua vùng có protective masking, khi đó protective masking vẫn được sử dụng nhưng nó vẫn không bảo vệ đối tượng mà mình mong muốn.

Trong một số trường hợp, quá trình dùng Seam Carving để resize, đưa ảnh sau khi xóa về đúng kích thước ban đầu có thể làm cho những chi tiết xung quanh bị méo mó. Nguyên nhân là vì thuật toán dùng đường Seam để thêm vào ảnh các đường có mức năng lượng thấp nhất, tuy nhiên vì đây là bước thêm các pixel, nên các đường Seam sẽ không bị mất đi và còn được thuật toán thực hiện nhân bản, hệ quả là cùng một đường Seam sẽ được thêm vào nhiều lần, làm cho các chi tiết của hình ảnh bị mất tự nhiên, nếu các pixel được thêm thuộc những đối tượng khác, có thể làm biến dạng đối tượng đó, kéo dài đối tượng đó trong ảnh.

Đôi khi việc xóa một đường seam có mức năng lượng thấp nhất lại có thể vô tình tạo ra một đường seam khác có mức năng lượng cao hơn. Một giải pháp có thể được sử dụng là mô phỏng “nhìn trước” bức ảnh sau khi thực hiện xóa đường seam và kiểm tra mức năng lượng của bức ảnh có tăng lên hay không, nếu có ta sẽ thực hiện với một đường seam khác.

Việc xét từng pixel dẫn đến xác suất những pixel “kém quan trọng” cũng có thể nằm trong vật thể của ảnh dẫn đến trường hợp thuật toán “vô tình” xóa đi mất (đáng ra không nên bị xóa). Ngoài ra, việc tính toán trên từng pixel tiêu tốn không ít thời gian, dẫn đến tốc độ thuật toán là không hề nhỏ.

Chương 6. Kết luận và hướng phát triển trong tương lai

Trong đồ án này, chúng tôi sử dụng seam carving cho việc xóa đối tượng trong một bức ảnh. Người dùng sẽ đánh dấu đối tượng mục tiêu cần được loại bỏ và sau đó các đường seam sẽ được xóa khỏi hình ảnh cho đến khi tất cả các pixel được đánh dấu là biến mất. Nhìn chung, seam carving là một cách tiếp cận hiệu quả cho việc chỉnh sửa hình ảnh. Đôi khi việc xóa đối tượng và resize chưa hoạt động tốt, có thể kể đến một số hạn chế của nó như: vật thể xung quanh bị kéo giãn, một phần của vật thể bị cắt đi. Mặt khác, seam carving không thể tự hoạt động trên toàn bộ ảnh mà không có sự hỗ trợ từ các phương pháp phát hiện vật thể hoặc tác động của người dùng (cung cấp mask cho vật thể).

Nhận thấy rằng có một số hạn chế được kể như trên, trong tương lai nhóm chúng tôi sẽ tiến hành cải tiến hàm resize, xây dựng protective masking có khả năng bảo vệ đối tượng (object) tốt hơn và thử nghiệm đầu vào là một video.

Chương 7. Cấu trúc đồ án

Báo cáo gồm 6 chương với các nội dung như sau:

- **Chương 1: Giới thiệu đề tài.** Giới thiệu về bài toán tìm hiểu và tình hình thực tại của đề tài.
- **Chương 2: Sơ lược về seam carving.** Trình bày sơ lược về ý tưởng và các hàm được sử dụng trong seam carving.
- **Chương 3: Cách tiếp cận của bài toán xóa đối tượng.** Trình bày về ý tưởng của bài toán xóa đối tượng sử dụng seam carving. Giải thích về các mask sẽ được sử dụng. Mô tả thuật toán xóa đối tượng sử dụng seam carving
- **Chương 4: Thử nghiệm.** Trình bày về quy trình chuẩn bị source code và tiến hành thực nghiệm trên các ảnh đầu vào. Trình bày về các thông số đã được sử dụng.
- **Chương 5: Đánh giá.** Trình bày về điểm mạnh và điểm yếu của seam carving. Đưa ra nhận xét chung về kết quả đã thử nghiệm trên các ảnh đầu vào.

- **Chương 6: Kết luận và hướng phát triển trong tương lai.** Tổng kết các nội dung quan trọng cần được và đưa ra một số cải tiến sẽ được thực hiện trong tương lai.

TÀI LIỆU THAM KHẢO

<http://cs.brown.edu/courses/cs129/results/proj3/taox/>

https://en.wikipedia.org/wiki/Seam_carving

<https://vnoi.info/wiki/cs/imageprocessing/Seam-Carving.md>

<https://courses.cs.washington.edu/courses/cse332/20wi/homework/seamcarving/>

<https://andrewdcampbell.github.io/seam-carving>

<https://inst.eecs.berkeley.edu/~cs194-26/fa18/hw/proj4-seamcarving/imret.pdf>

<https://github.com/Spellstaker/Seam-Carving>

[https://github.com/erwin24092002/CS231---
Computer_Vision/tree/main/%5BASSIGNMENT%2306%5D%20Seam%20Carvin
g](https://github.com/erwin24092002/CS231---Computer_Vision/tree/main/%5BASSIGNMENT%2306%5D%20Seam%20Carving)

<https://perso.crans.org/frenoy/matlab2012/seamcarving.pdf>

-----**Hết**-----