Seam carving final project - Computer vision

Nexhi Sula

May 2023

1 Introduction

As we are aware, with the advancement of technology, developers are deploying more and more applications and websites for the needs of people and making their lives easier. Images are the most important part when it comes to these applications and since we use different kinds of phones, image size is critical. There are different techniques for adapting an image for different devices such as cropping or standard scaling but at the end the image ends up losing information only in the periphery when we perform cropping. These techniques don't seem to be the best way when it comes to resizing since we lose important content from the image. Shai Avidan and Ariel Shamir introduced seam carving that can change the size of an image by gracefully carving-out or inserting pixels in different parts of the image. As they explained in their paper Seam carving uses an energy function defining the importance of pixels. A seam is a connected path of low energy pixels crossing the image from top to bottom, or from left to right. For this final project I am going to implement seam carving for image resizing using the same techniques as they did. So, the main steps of the implementation are:

- Computing the energy map of the image or in other word finding the edges of the image
- Calculating the energy map matrix which will store the least energy of the pixels and the path of these values.
- Finding the seam with the lowest energy and then removing it from the image.
- Comparing the original image with the resized one.

2 Data description

My data for this project consist in different images and trying to remove as many pixels as possible and I have used the same images as in the paper and using other images from `https://www.flickr.com/` form the same website that the authors used.

3 Approach

As mentioned above for implementing seam carving, I have used several steps and I will explain each of them in detail.

3.0.1 Edge detection/energy map computation

For detecting the edges, the first thing that I did was converting the image in grayscale and blurring it using the Gaussian filter. To compute the energy map for the image, so in other words finding the energy value for each pixel I tried several methods such as Canny edge detection, Laplacian of Gaussian, and Sobel filter. From the results that I will present below the Sobel filter was the best approach for detecting edges and computing the energy map because it convolves the image with two 3x3 matrices, one for horizontal and one for vertical edges.

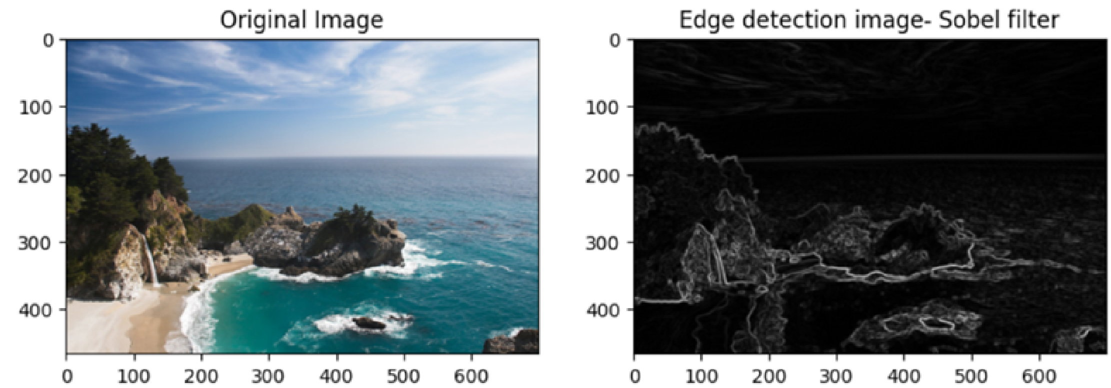

Figure 1: Sobel filter for edge detection.

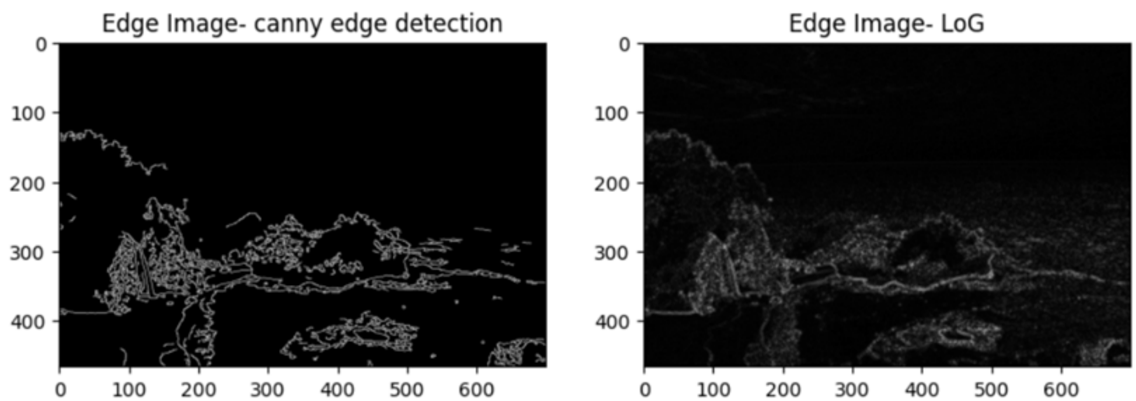

Figure 2: LoG and Canny filter for edge detection.

3.0.2 Computing energy map and directions of the pixels with the lowest energy

This was the hardest part of the implementation because the authors in the papers have used dynamic programming to find the energy map. It was a challenge because I haven't used it before but with some good online tutorials, I managed to do it. So, I created another matrix with the same dimension of energy matrix that I found at edge detection step and copied the very bottom row of it to the new matrix. Starting from that row and going up to the energy matrix, I calculated the energy map, getting the lowest energy compared to the three pixels below the current pixel. While computing the minimum value of energy, at the same time I saved the directions or the path of these values. This will help me later to find the seam with the lowest energy.

So, the black regions (right image) in figure 3 are those with the lowest energy value and the brightest ones have the most important content which we need to preserve when removing the pixels. The reason why we see the triangle form in the image is because of the dynamic computing that we used for computing the energy map. The brighter the region the higher the value of energy.

3.0.3 Finding the seam with the lowest energy and removing it from the image.

For finding the seam I have used the directions matrix that I computed from the previous steps where I saved the paths with the lowest energy value. First I wrote a function that uses this matrix and a random pixel at the top row of the image and after that calling the function with the minimum element from the energy map (figure 4).

After finding the seam, I have manually removed the pixels from the image when initializing the new image sizes as below:

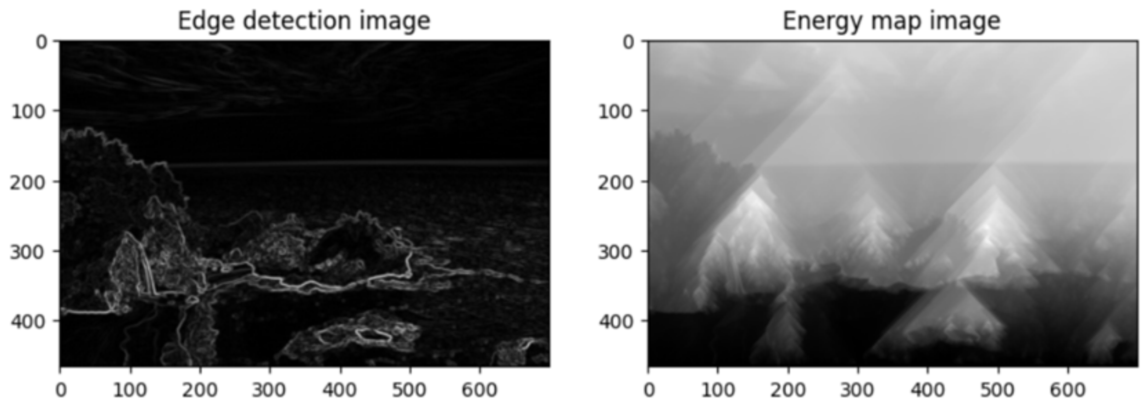

Figure 3: Energy map

```
[17] def remove_seam(img, seam):
       h, w, _ = img.shape
       new_image = np.zeros((h, w-100, 3), dtype = np.uint8) #creating a new image but with a reduced number of pixels for the width

       for i in range(h): #going though each row of the image and remove the pixels for the seam found earlier
         j = seam [i]
         new_image[i, :, :] = np.concatenate((img[i, :j, :], img[i, j+100:, :] ), axis = 0)  #Concatenating both sides of the image without the seams

       return new_image
```

Figure 4: Code for the seam with the lowest energy

4 Results

Below are the results with different images and removing different pixels from them. So for the first image which is the same that the authors have used in their paper I have used removed 100 pixels and then 200 pixel. When running the code I did the same for the other images as well but I am showing here the results only when I removed 200 pixels (figures from 5-8)

5 References

1. Original paper - Seam carving for content-aware image resizing https://dl.acm.org/doi/abs/10.1145/1275808.1276390?casa_token=fXsCunEycxwAAAAA:uppc1ns_tyfzpEzinrx0nq4Up6PdtvUG9cOPHxoTytWJ vvlT1A

2. Seam Carving. EECS 194-26: Image Manipulation and Computational Photography. UC Berkeley https://inst.eecs.berkeley.edu/~cs194-26/fa17/upload/files/projFinalAssigned/cs194-26-adm/seamCarving.html

3. Blog - Implementing Seam Carving with Python https://karthikkaranth.me/blog/implementing-seam-carving

4. Dynamic Programming https://www.youtube.com/watch?v=aPQY__2H3tE

5. Blog - Seam Carving Algorithm : A Seemingly Impossible Way of Resizing An Image https://www.analyticsvidhya.com/blog/2020/09/seam-carving-algorithm-a-seemingly-impossible-way-to-resiz

6. Blog -Seam Carving https://sites.coloradocollege.edu/blockfeatures/2020/11/08/seam-carving/

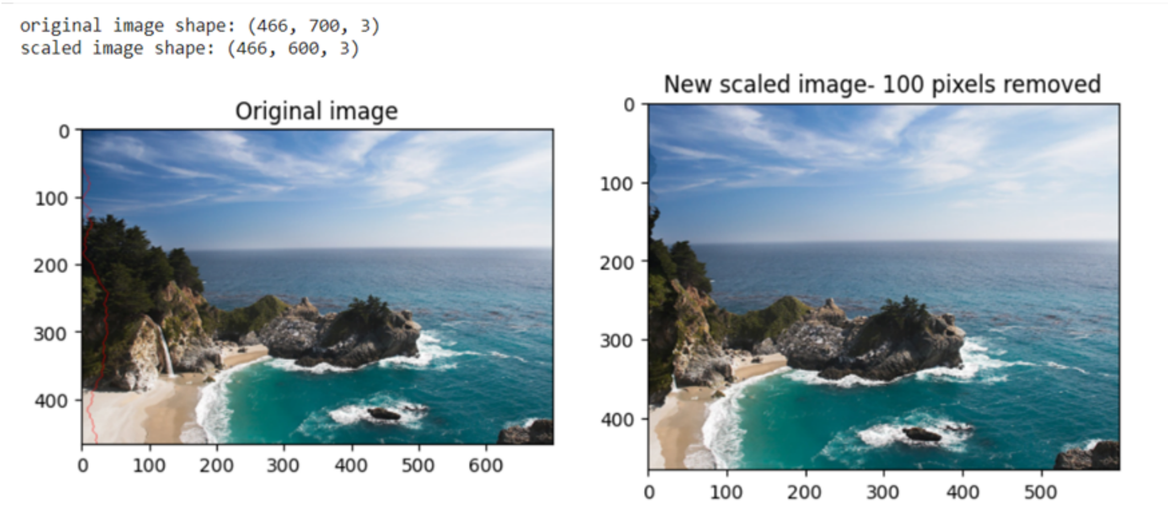

Figure 5: Comparison between the original and scaled image for 100 pixels

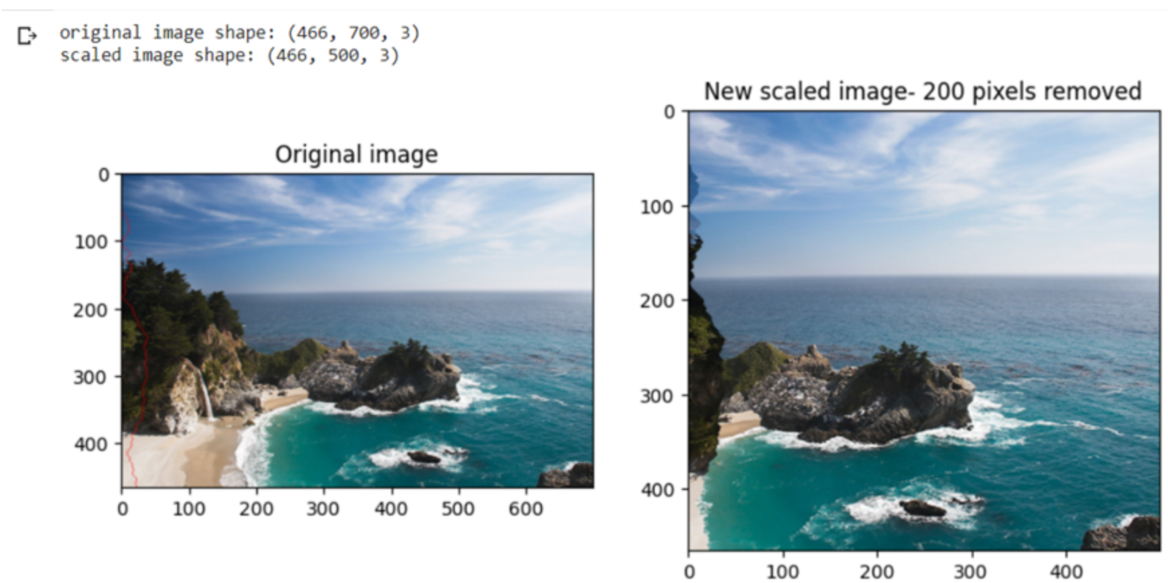

Figure 6: Comparison between the original and scaled image for 200 pixels

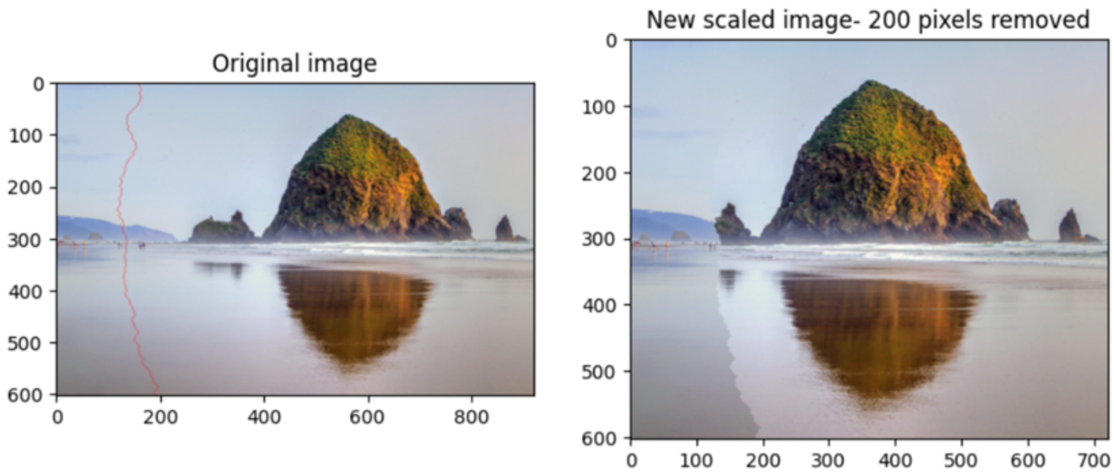

Figure 7: Comparison between the original and scaled image for 200 pixels

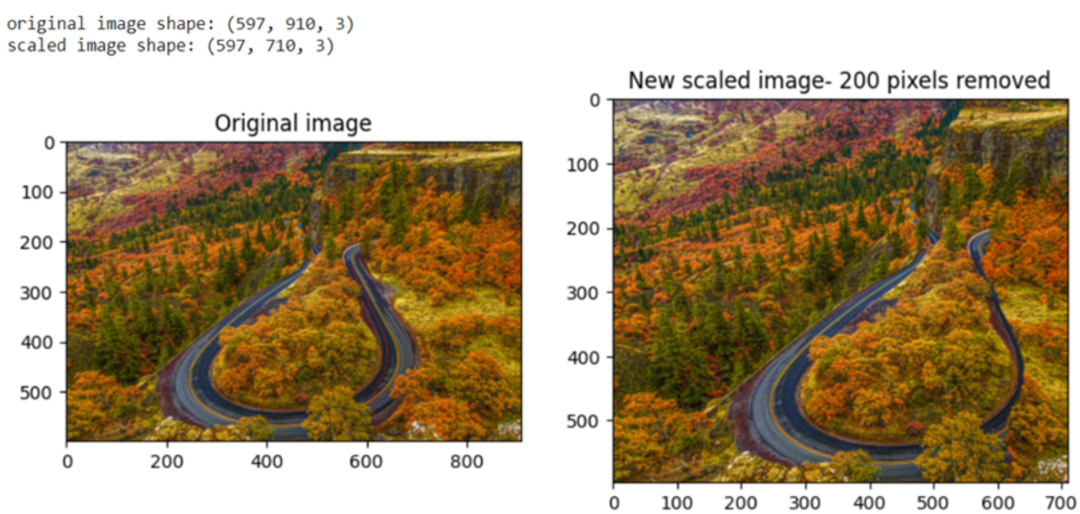

Figure 8: Comparison between the original and scaled image for 200 pixels