CS 435 – Computational Photography
Assignment 3

In this assignment you will demonstrate your ability to perform seam carving for re-targeting images.

Grading Scheme:

1. Theory Questions (45pts)
2. Naïve Cropping and Stretching (15pts)
3. Energy Functions (10pts)
4. Optimal Seam Discovery (15pts)
5. Seam Removal (15pts)

Theory Question(s)

1. In lecture we derived the equations for 1D Poison blending to obtain an equation in the form $Af = b$. At the end of that material we briefly discussed how this could be extended to 2D (so that we can use it for images). Now let's do the math to support the creation of the A matrix and the b vector!

   For simplicity let's assume a 2×2 square mask. Below are two images, the one we're copying *to* and the one we're copying *from.* The masked area is indicated in green.

1	0	2	2	1
4	3	5	1	2
4	4	4	4	6
4	5	2	0	2
2	3	3	0	3
1	0	0	5	2

Image we're copying ***from***

4	5	6	1	2
1	2	5	3	3
3	3	3	6	4
2	4	2	5	1
4	5	3	1	0
3	3	1	1	0

Image we're copying ***to***

   a. (10pts) First let's establish the cost function Q for this 2D case. You may assume squared error, therefore the cost with regards to pixel (i, j) will be:

$$J_{i,j} = \left(\left(f_{i,j} - f_{i-1,j}\right) - \left(t_{i,j} - t_{i-1,j}\right)\right)^2 + \left(\left(f_{i,j} - f_{i,j-1}\right) - \left(t_{i,j} - t_{i,j-1}\right)\right)^2$$

   Computing the cost over our entire region (for the example above) we have:

$$J = \sum_{i=2}^{4} \sum_{j=2}^{4} J_{i,j}$$

   Find the equation for J using the values found in the matrices above.

   b. (10pts) Next compute the partial derivative of this cost function with regards to our knowns (there are four unknown pixel values so there should be four equations…).

c. (5pts) Finally, to minimize the partials, set them equal to zero and arrange in a matrix format such that $Af = b$

Hint: You should be able to write A such that for any given row there is an 8 on the pixel corresponding to that partial, and a -2 for the columns on that row that are neighbors to that pixel.

d. (5pts) Solve for f using Matlab

2. If the image below is the *gradient* of an image

2	3	4	5	1
1	0	2	2	1
4	3	5	1	2
4	4	4	4	6
4	5	2	0	2
2	3	3	0	3

a. (10pts) Construct the optimal seam matrix if we assume vertical seams

b. (5pts) What is the optimal seam?

Programming Introduction

For this assignment we are going to implement some image retargeting approaches, allowing us to change the size and aspect ratio of an image.

For this assignment, select two images of interest to you so that we can multiple example I/O of each approach.

Part 1: Crop and Rescale

First let's start with some naïve approaches!

For each of your images:

1. Make the images larger, while maintaining the aspect ratio
2. Make the images smaller, again while maintaining the aspect ratio
3. Change the aspect ratio

Do this all in color! You **may not** use any Matlab functions (for instance, `imresize`) to do these changes for you. The desired width and height should be any valid value, i.e. $w, h \in \mathbb{Z}_0^+$. One decision you'll have to make is what to do when one pixel doesn't perfectly map to another pixel's location. Approaches include selecting the nearest location, interpolating values in some region, etc...

For your report provide:

1. Sample I/O for both your test images (8 total images).
2. A description of your approach to naively resizing an image. Including citations if appropriate.

Part 2: Energy Function

Before we get into actually implementing seam carving algorithms, let's take a look at the energy functions of your images.

For each of your images, compute its energy and visualize this as an image.

Some implementation details:

- Do this in grayscale
- Pre-smooth your grayscale image prior to getting the gradients (or do this in one step). Choose parameters that make sense for you (and report them!).
- Pad your images such that the convolution processes don't change the size of the image.
- You **may** use Matlab functions like `conv2, rgb2gray`
- Note: You may need to go back and forth between double and uint8 data types for computations and viewing, respectively.

Original image and its associated energy function

Part 3: Optimal Seam

Ultimately we want to do some seam carving. To do this we must be able to find the optimal seam! Since this process is typically done via dynamic programming, create functions like:

```
function M = getOptimalSeam(E,M,i)
      %E is energy function
      %M is current seam matrix
      %i is the current row being processed

function P = backtrack(M)
      %M is the seam matrix
      %P is the optimal path
```

Now for each of your images:

1. Find the optimal seam. Since this is more easily done via dynamic programming, create
2. Superimpose on your **color** image the optimal seam in red.

Additional Details:

- We will do vertical seam carving, starting at the top of the image.
- You'll likely have to think about how to handle the edge cases.

Original image with the optimal vertical seam shown in red.

Part 4: Seam Carving

Let’s use seam carving to reduce the aspect ratio. Keeping the height fixed, allow the user the chose a width, w', that is $0 < w' \leq w$, where w is the original width.

Additional Details:

- We will do vertical seam carving, starting at the top of the image.
- Depending on how optimally you write your code (leveraging efficient Matlab operators instead of doing loops, etc..) it may take a while to reduce from $w \rightarrow w'$. So at the very least first run your code with only a small lowering of the width.

For each of your images:

1. C*reate* a **video** showing the seam removal process. Each frame should depict the current color image with the current optimal seam superimposed (like in the previous part). ***Note: You may need to "pad" your frames so that they all have the same size in order to render as a movie.***
2. Show the before and after images.

Original image and one with a smaller width

Submission

1. Assignments must be submitted via Bd Learn
2. Submit a single compressed file (zip, tar, etc..) containing:
    a. A PDF file containing:
        i. Your answer to the theory question(s).
        ii. Your two original images.
        iii. Your six cropped and rescaled images (Part 1).
        iv. Your two energy function images (Part 2).
        v. Your two optimal seam images for Part 3.
        vi. Your two before-and-after images for Part 4.
    b. A README text file (**not** Word or PDF) that explains
        i. Features of your program
        ii. Name of your entry-point script
        iii. Any useful instructions to run your script.
    c. Your source files
    d. The chosen images that you are processing.
    e. The videos generated for Part 4.