Computational Photography

Final Portfolio

Akemi Matsuoka

CS6475 - Fall 2019

Assignment #1: Epsilon Photography

Image 1

Image 2

Image 3

Image 4

Final Artifact

In this assignment, multiple images were captured with only one element changing. This final artifact shows a single moving object in 4 different positions. The ISO, exposure and aperture were the same in all 4 images. . All the images were blended into one image using the Overlay method and a bitwise operation was also performed.

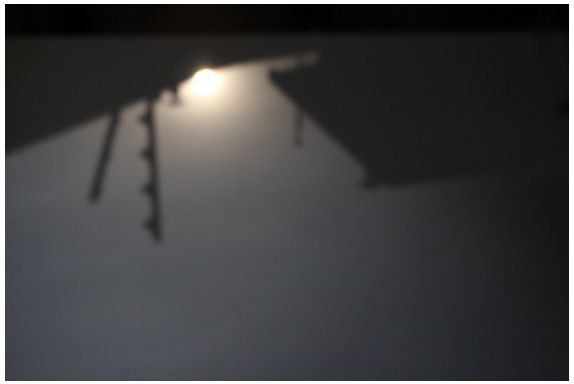
Initially, the images were taken by a mobile camera at home. I faced several problems with image quality such as White balance variations (Cool and Warmer colors variation between pictures); Motion blur due to the camera movement when shooting pictures without a tripod. Due to these problems, a scenario with proper conditions and a DSLR were used to perform this assignment. The camera used was Canon T5i and manually adjusted to preserve image quality. The camera was also attached to a tripod.

Camera settings:

Exposure: **1/30s Aperture: f/4.5 ISO: 1600**
Focal length of 20mm

Assignment #2: Camera Obscura

The Scene



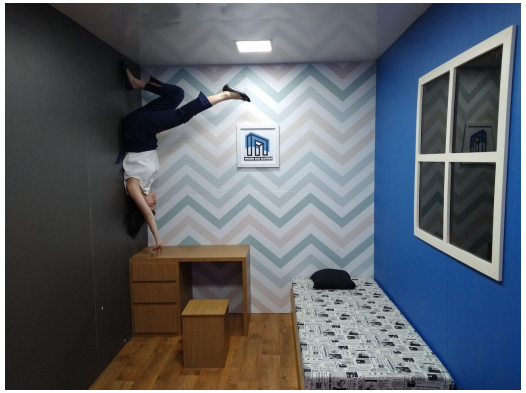
The Final Image

The Set Up

Above & Beyond

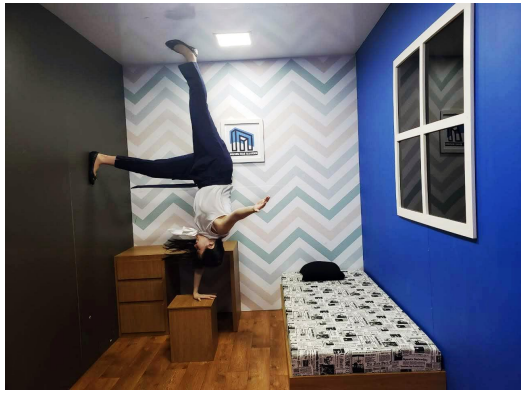
The purpose of this assignment is to build a Camera Obscura, show its process of construction and register the images captured. The camera parameters were f/5, shutter speed of 2 seconds and ISO 1600. For the A&B artifact, aperture f/8, shutter speed of 10s, and ISO 100 to perform light painting. The setup of the Camera Obscura requires a lot of effort, materials to cover the window from light, tripod, a good camera and planning to take picture on sunny day. I was satisfied with the results because the final image produced is inverted, as seen on class.

Assignment #3: Blending

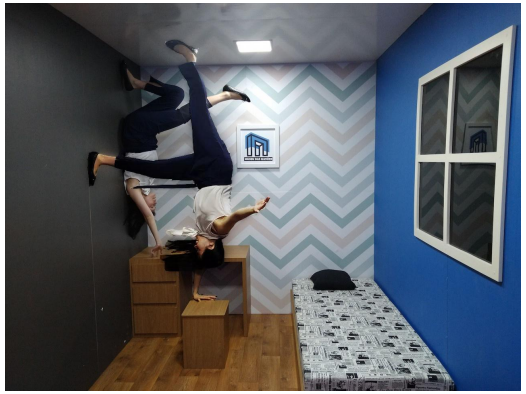


Black Image

Mask Image



White Image



Final Artifact

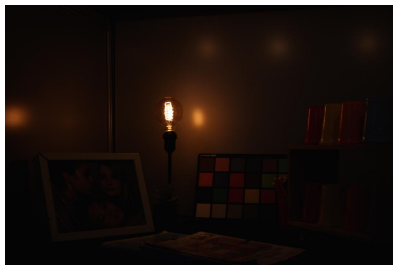
For this assignment, it was produced a pyramid blending code and a laplacian pyramid blend. The input was two images and a mask. The mask will only allow portions of the “white” image to be blended into the black image. Here, we can see the main subject from the white image was masked and blended with the black image. The final artifact was satisfactory, due to the additional use of gamma correction. It was performed this operation to correct the illumination variation from the input images.

Assignment #4: Panoramas

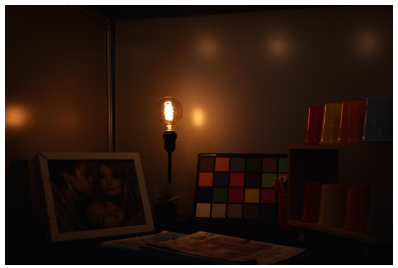
The purpose of this assignment was to construct a panorama of a sample image set. To perform this, features between images were matched,, homography was calculated, the images were warped onto the panorama canvas, and finally the images were alpha blended together to output the panorama. In the final artifact, the boundaries between the images 2 and 3 were blended well, while the images 1 and 2 were not. The main error was when I photographed the scene 2, where we can notice a slight angle difference between images 1 and 2. A more sophisticated feature matching would solve this problem.

Assignment #5: HDR

Img 1

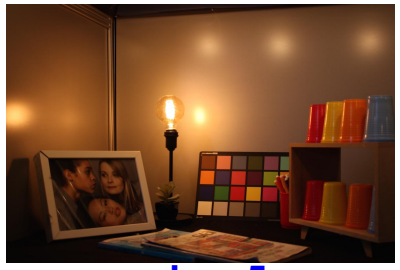


Img 2



Img 3

Img 4



Img 5

Img 6

In this assignment, an High Dynamic Range image was computed programmatically using sets of different images with exposure variating. The HDR image was calculated by weighting the pixels in each of the input photos, sampling pixel intensities, computing a response curve, and finally computing a radiance map to provide a normalized output image. The main challenge in this project, was to show the lamp details, which we can see in Img 1.

- Camera settings: Exposure Time: ([1/125, 1/60, 1/30, 1/15, 1/8, 1/4])
- Aperture: f/4.5
- ISO: 400

Assignment #5: HDR

The overall result, details of the light bulb and objects worked well. The first output was opaque. After applying tone mapping the color was more balanced and warm (like the original scenario) throughout the image. I tested global tone mapping using gamma correction with different parameters. The best result was when gamma = 0.6.

Assignment #6: Video Textures

In this assignment, video textures were implemented to create a seamless loop in a video. It was done by breaking the video down into frames and using them to calculate transition matrices. These matrices were then used to find frames with similar transition points to create a seamless loop.

Overall, I was happy with my results. If we compare against the original source video, the final output gif seems pretty smooth. On the original video, there is one part the flame is shorter and tilted to the right. On the output, it loops reasonably well, keeping the main subject more stable than the original. While the result from my own video, it shows a good transition between start and end. This result was achieved after finding a good alpha value. A higher value for alpha was tried for this video and there was an abrupt cut in the video. The results for the two texture gifs produced can be found in the links below:

- Working link for candle video texture gif - https://drive.google.com/open?id=1KUQqFMf32L9lLlbGIxqg17ox7UALCUd6
- Working link for my own video texture gif - https://drive.google.com/open?id=17WXRiNSmVOcvZR4ApLPKV87IHPVK-NSW
- Working link to the frames (folder) - https://drive.google.com/open?id=1wlAb9lB1D3xy58zvNwQkJLprDwqZpSp1

Midterm Project

The goal of this project is to replicate the results from **Seam Carving for Content-Aware Image Resizing** by **Shai Avidan** and **Ariel Shamir** and I**mproved Seam Carving for Video Retargeting** by **Michael Rubinstein, Ariel Shamir**, and **Shai Avidan**

The purpose of this project was to explore content aware resizing of images. A process called seam carving was implemented in order to reduce or expand image seams with the lowest energy. The original seam carving method proposed can create visible artifacts. It happens because this method ignores the energy inserted in the photo. In the improved seam carving, the authors select the seam that introduces the least amount of energy into the picture This is called forward energy. To incorporate this through a dynamic programming approach, we simply need to change our cost function.

Figure 1: Seam Removal. From left to right: Original image and Image resized using seam removal by 50%

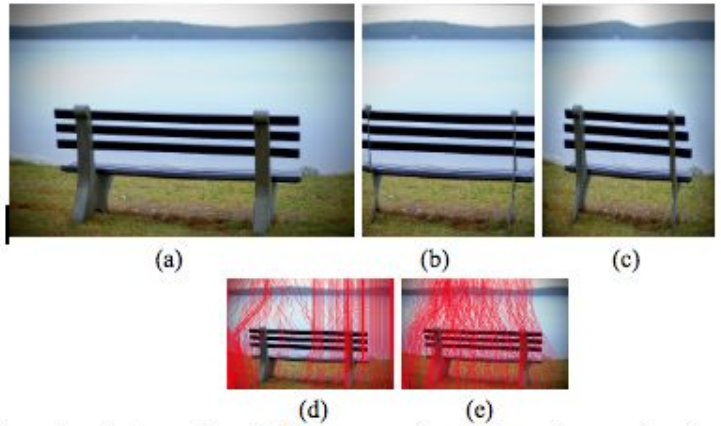


Figure 2: Comparison between the original seam carving backward energy showing artifacts (b) and its seam in (d). The image using forward energy (c) and its seams in (e).

Midterm Project

We can reduce an image's width by calculating the vertical seams with the lowest accumulated energy in the image. Then, we delete the number of seams needed that have the lowest total energy. If we want to increase image size, we extend the images with the the lowest energy seams. I considered their definitions and implementation very confusing. In particular, the backward energy function was not specified which method they used. I tested several methods and Sobel Gradient gave me better results for this project.

Overall, my final images look similar to the results presented in the paper. However, they are not identical. The seams presented in Figures 2 are close to the result of the original work. However, the seams from Figure 3 are not the same as shown in paper, which was the method that I had more difficulties to understand. In Figures 2 and 4, both cases demonstrated that forward energy performed better, showing fewer artifacts. You can see more details in this Video link: https://youtu.be/Arcpwwxrmtw

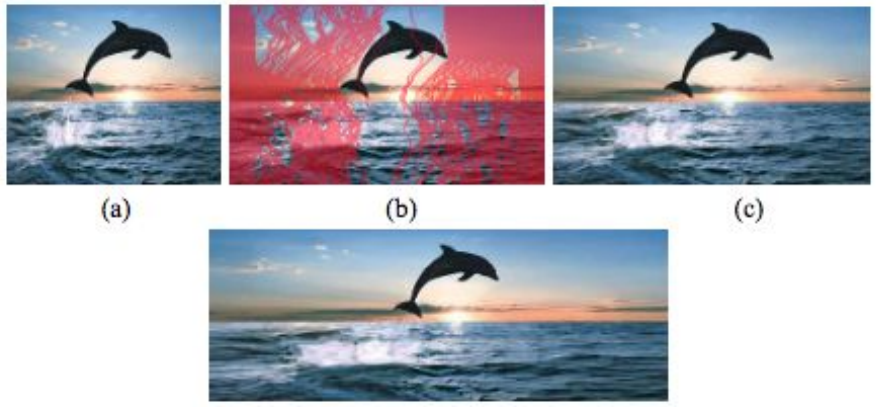


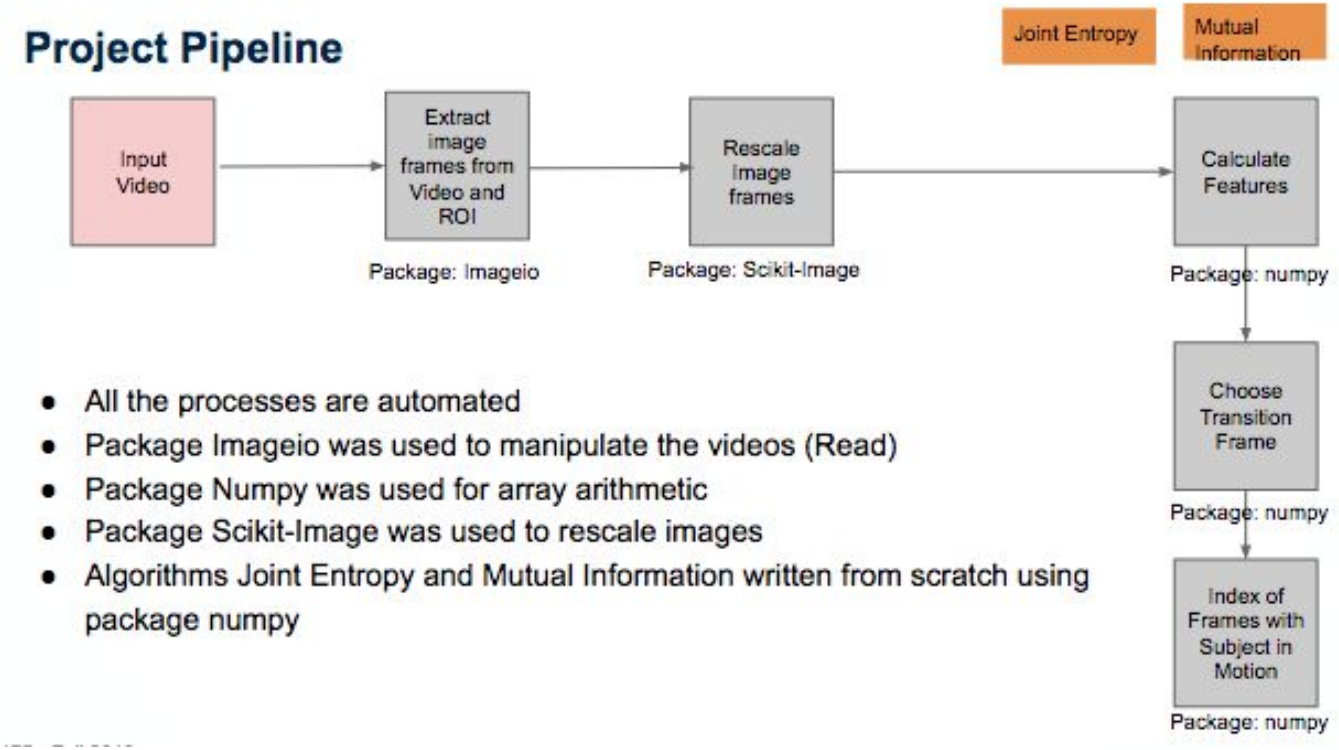
Figure 3: Seam insertion. Seams that will be added in the same order of removal (b), resulting in the image (c) and increasing the width again by 50% in (d)

Figure 4: Comparison between the original seam carving method showing artifacts (b) and its seam in (d). The image using forward energy (c) and its seams in (e).

Final Project

Goal

This project will calculate unique features in the video to detect cuts in a video. The paper **Shot detection in video sequences using entropy based metrics** by **Z. Cernekova ; C. Nikou ; I. Pitas** was implemented to detect abrupt cuts and soft cuts which were studied in the Video Processing lecture.



Final Project

The feature implemented can automatically capture the changing content in a video. In this example, the input was a video with a ball in motion. After processing, the output represents the identified changes. One of the challenges was to understand the basic concepts that I've worked with. I had to review all the theories related to Entropy to implement the algorithm. I also found it challenging to understand the Co-occurrence matrix and had to study this concept from extra resources.

Since we are working with videos, the camera lens can refocus during the video. It can lead to brightness changes. Both features, Mutual Information, and Joint entropy are sensitive to illumination changes leading the algorithm to show false positives in the results (It means the algorithm can detect a frame as a false abrupt cut). You can see the videos link and also the output images.