Fatih Mehmet YİĞİTEL
Barış SİNAPLI
Assist. Prof. Aslı GENÇTAV
CMPE 362 – Spring 2023

June 11th, 2023

Digital Image Processing – Assignment 4

Introduction

In this report, image compositing, which is a sub-field of image processing, is discussed. Image compositing is a technique that allows creating larger, higher quality images using multiple small images. To do this, it uses multiple images with overlapping fields of view. In this assignment, we will focus on the results we obtained using this technique using different inputs and this process. Among the tools we have used for this task are Python, which we have used in previous image processing studies, and OpenCV, a library containing image processing tools that we will use.

Process

Image compositing consists of many steps. First, we uploaded the images using *"cv2.imread"* and then changed the images to grayscale with *"cv2.cvtColor"*. This is because the feature detection step works best on grayscale images.

Next, we used the *Scale-Invariant Feature Transformation (SIFT)* method on the images. SIFT detects important points in the image and creates a 128-dimensional feature vector for each point. To do this, we used the *"detectionAndCompute"* method in the *"cv2.SIFT_create"* class.

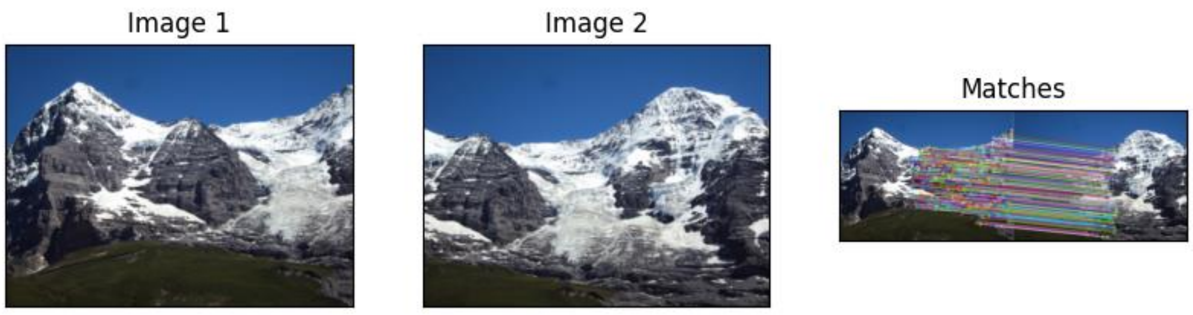
After determining the features of the images, we started the matching part of these features. To do this we had to use a Brute-Force matcher algorithm built using the *"cv2.BFMatcher"* class. This allowed us to use the K-nearest neighbor (KNN) algorithm with a k-value of 2 done via the *"knnMatch"* method. In the next step, we ran a ratio test to find the best matches and visualized those matches using the *"cv2.drawMatchesKnn"* function.

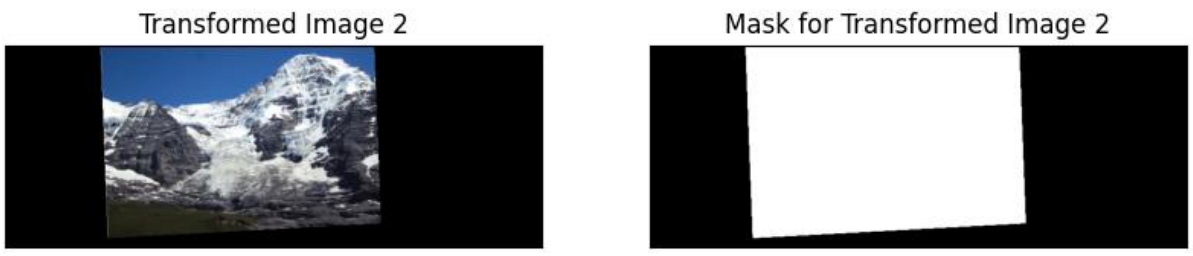
We started registrating the images with both the features we found and the matches. In this step, we needed to estimate the transformation matrix required to produce correct results using the *"cv2.estimateAffine2D"* function. Because it helped in the step of converting this matrix to the second image later using the *"cv2.warpAffine"* function. We also created and transformed a mask image that matches the size of the second image.

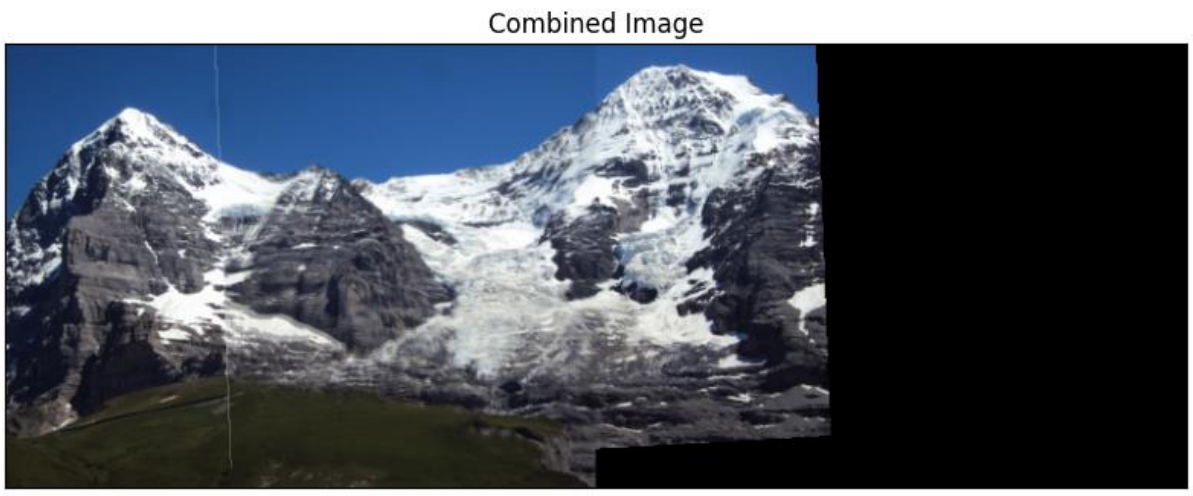
In the last step, we brought together registered images. We enlarged the first image to match the size of the second converted image. We also created a mask for the first image to show where the pixels are. Finally, we combined the two images by averaging the pixel values in the areas where they overlap.

Results and Discussion

After this process, we were able to create a single combined image using the contents of the two images and render it as desired with two different inputs. In this process, we made many attempts to get the right result, and finally, we were able to combine the inputs correctly with the right features. We were able to accurately predict the conversion matrix thanks to the removal of weak matches with the ratio test, and we were able to get visually appropriate results with strong matches.







The quality of the combined image that emerges as the final product demonstrates how well the process works. It helped us to properly overlay images by demonstrating the accuracy of compositing via the transform matrix, and the method of averaging pixel values in overlapping areas also helped reduce visible seams between images, improving the overall quality of the final image. As a result of this homework, we have better grasped the principles and techniques behind the operation of the image combining method in the field of image processing, and as a result of our efforts, we have produced the desired results in the homework. Also, the high quality of the images we used in this assignment also helped us to get these quality results. Because if we used very little overlapping, pattern-like or missing inputs, our homework would not yield correct results. For this reason, the effect of the images we have taken from the database given to us in the homework should not be denied. Thank you for your attention.