
BBM418 Computer Vision Laboratory

Programming Assignment 2 – Image Panorama Stitching

Mehmet Giray Nacakci

21989009

Department of Computer Engineering
Hacettepe University
Ankara, Turkey

b21989009@cs.hacettepe.edu.tr

Overview

In this assignment, the aim was to stitch different images of the same scene into a panorama. These images cannot be directly concatenated; since they have their scale, rotation, translation, shear, and perspective different, relative to each other.

1 Feature Extraction

Each image in a dataset is passed from three different feature extraction algorithm. So that common features could be matched later.

OpenCV implementation of SIFT, SURF, ORB algorithms are used. The quality of the results at the end will be discussed later. The computational performance is discussed below:

For SIFT and ORB, `maximum_number_of_features = 1500` parameter is used. The best 1500 features are used. Less gave worse looking panoramas, more caused computational overdue.

For SURF, `hessian_threshold = 400` parameter is used, since it is the value recommended by OpenCV documentation.

SIFT detected 82616 features in 9 seconds. That is 9178 features/second

SURF detected 217156 features in 10 seconds. That is 20739 features/second

ORB detected 89127 features in 1.4 seconds. That is 61282 features/second

By the results, ORB is the fastest, then SURF, then SIFT.

2 Feature Matching

Each image in an image set is matched pair-wise with the first image. First image, because, in most image sets, objects in first image looks the most perpendicular to the image plane. I thought using it as a reference would be the most intuitive and precise while calculating homography later.

matching 1.png with 2.png,
1.png with 3.png,
...,
1.png with n.png

OpenCV Brute Force Matcher calculates distances between the two images' features using the keypoint descriptors obtained in step1. Lowest distance responses are best matches. Then, I eliminate the noise by taking only the best 80% of the matches returned by the brute-force-matcher.

3 Finding Homography

OpenCV function `findHomography` is used with RANSAC setting, **between each image pair**, to figure out how `img_2` can be transformed into `img_1`, such that the matching features could overlap if we put them on top of each other after transformation.

4 Stitching the Panorama

- First, the Panorama Canvas is initialized with `first_image`.
- Then, the `second_image` is applied homography transformation. Yet, the transformed image matrix can hold negative index values that are actually some parts of the image, since it is rotated or bigger than its original size. And not using those indices lead to image being cropped. No, we want all of it.

Thus, a translation (figured out by perspective transforming the corners of the image to find new widest index values) is applied to the homography matrix itself, and then the new homography is applied to the `second_image`.

`new_homography = (translation)(homography)`

`img_2_transformed = warpPerspective(img_2, new_homography)`

To fit the transformed second image into the canvas, canvas is added black (empty) padding to each side up to its new size.

- Finally, the empty (black) pixels in the canvas is replaced by transformed second image. Thus, pixels on canvas which are coming from the first image are not manipulated.

In next steps, assign this whole canvas as the first_image, and add next new image as second_image. Thus, each image in the image set is iteratively added on top of the existing panorama canvas one by one. Since, intentionally, no additional blending or edge smoothing applied, stitched between images are visible.

Panorama Results. For the given image sets (6 folders with 6 images), rendering panorama for SIFT, SURF, ORB separately; the total time takes around 3 minutes.

5 Results

Below are all (18 cases: 6 image sets x 3 algorithms) the results of Feature Extraction (feature points drawn but too small to see) and Feature Matching (lines are drawn between matched features) plotted. And the Resulting Panorama.



SIFT Panorama : v_bird



SURF MATCHED v_bird 1.png 2.png



SURF MATCHED v_bird 1.png 3.png



SURF MATCHED v_bird 1.png 4.png



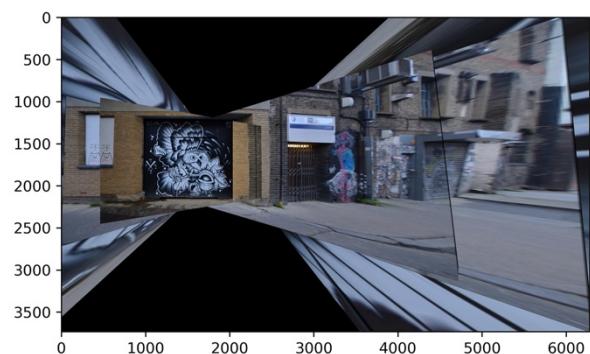
SURF MATCHED v_bird 1.png 5.png



SURF MATCHED v_bird 1.png 6.png



SURF Panorama : v_bird



ORB MATCHED v_bird 1.png 2.png



ORB MATCHED v_bird 1.png 3.png



ORB MATCHED v_bird 1.png 4.png



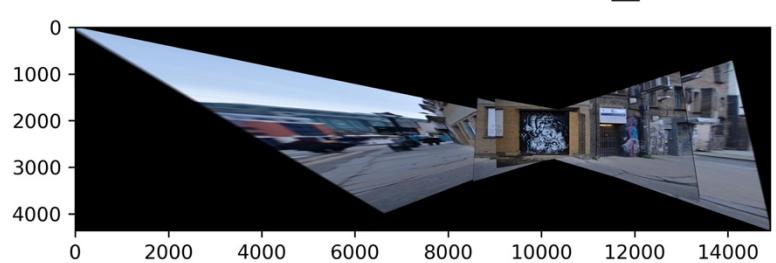
ORB MATCHED v_bird 1.png 5.png



ORB MATCHED v_bird 1.png 6.png



ORB Panorama : v_bird



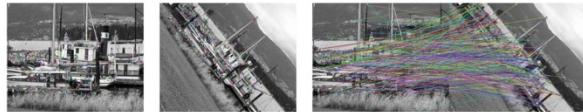
SIFT MATCHED v_boat 1.png 2.png



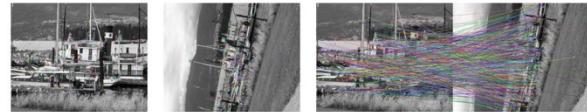
SIFT MATCHED v_boat 1.png 3.png



SIFT MATCHED v_boat 1.png 4.png



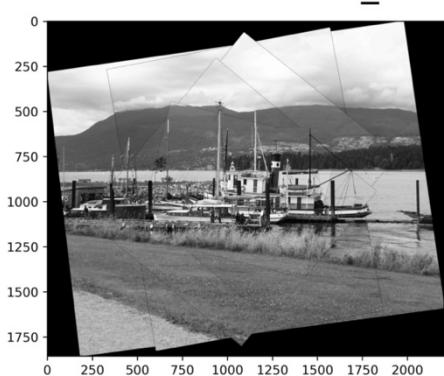
SIFT MATCHED v_boat 1.png 5.png



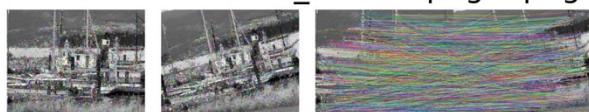
SIFT MATCHED v_boat 1.png 6.png



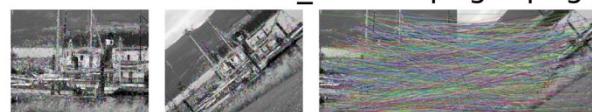
SIFT Panorama : v_boat



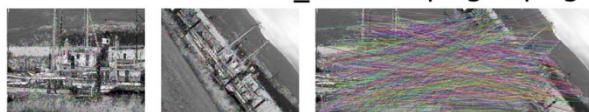
SURF MATCHED v_boat 1.png 2.png



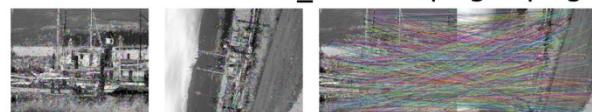
SURF MATCHED v_boat 1.png 3.png



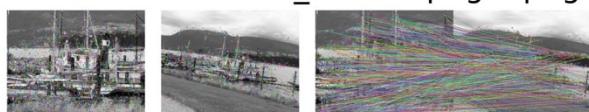
SURF MATCHED v_boat 1.png 4.png



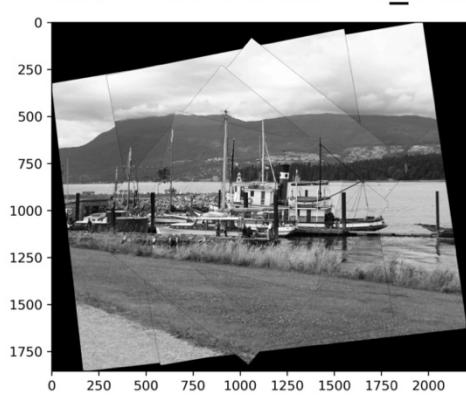
SURF MATCHED v_boat 1.png 5.png



SURF MATCHED v_boat 1.png 6.png



SURF Panorama : v_boat



ORB MATCHED v_boat 1.png 2.png



ORB MATCHED v_boat 1.png 3.png



ORB MATCHED v_boat 1.png 4.png



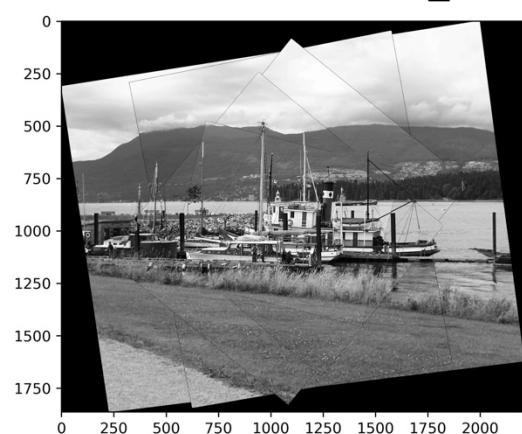
ORB MATCHED v_boat 1.png 5.png



ORB MATCHED v_boat 1.png 6.png



ORB Panorama : v_boat



SIFT MATCHED v_graffiti 1.png 2.png



SIFT MATCHED v_graffiti 1.png 3.png



SIFT MATCHED v_graffiti 1.png 4.png



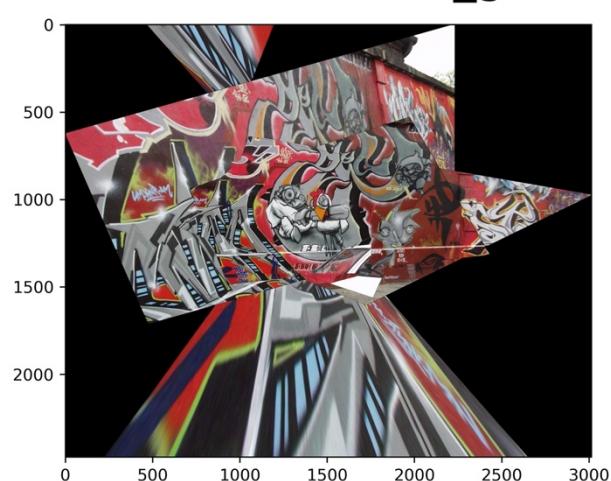
SIFT MATCHED v_graffiti 1.png 5.png



SIFT MATCHED v_graffiti 1.png 6.png



SIFT Panorama : v_graffiti



SURF MATCHED v_graffiti 1.png 2.png



SURF MATCHED v_graffiti 1.png 3.png



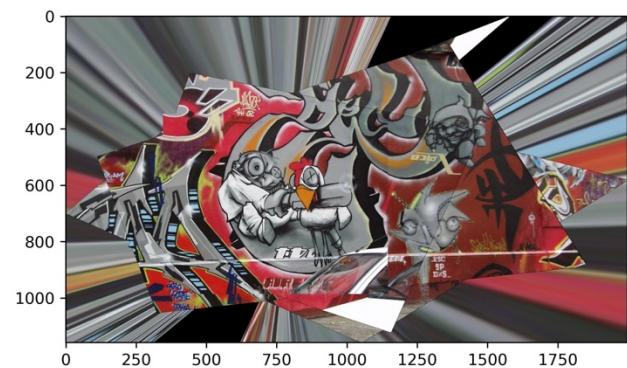
SURF MATCHED v_graffiti 1.png 4.png



SURF MATCHED v_graffiti 1.png 5.png



SURF Panorama : v_graffiti



ORB MATCHED v_graffiti 1.png 2.png



ORB MATCHED v_graffiti 1.png 3.png



ORB MATCHED v_graffiti 1.png 4.png



ORB MATCHED v_graffiti 1.png 5.png



ORB Panorama : v_graffiti



ORB MATCHED v_graffiti 1.png 6.png



SURF MATCHED v_soldiers 1.png 2.png



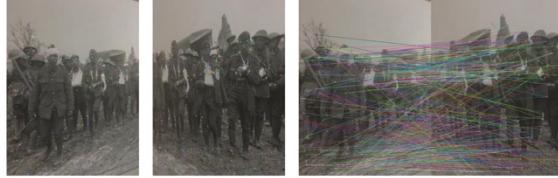
SURF MATCHED v_soldiers 1.png 3.png



SURF MATCHED v_soldiers 1.png 4.png



SURF MATCHED v_soldiers 1.png 5.png



SURF Panorama : v_soldiers



SURF MATCHED v_soldiers 1.png 6.png



ORB MATCHED v_soldiers 1.png 2.png



ORB MATCHED v_soldiers 1.png 3.png



ORB MATCHED v_soldiers 1.png 4.png



ORB MATCHED v_soldiers 1.png 5.png



ORB MATCHED v_soldiers 1.png 6.png



ORB Panorama : v_soldiers



ORB MATCHED v_weapons 1.png 2.png



ORB MATCHED v_weapons 1.png 3.png



ORB MATCHED v_weapons 1.png 4.png



ORB MATCHED v_weapons 1.png 5.png



ORB MATCHED v_weapons 1.png 6.png



ORB Panorama : v_weapons



6 Discussion: SIFT vs SURF vs ORB Panorama Quality

v_bird: SIFT > SURF > ORB

v_boat: SIFT ~ SURF ~ ORB

v_graffiti: SIFT > SURF > ORB

v_circus: SIFT ~ SURF ~ ORB

v_soldiers: SIFT ~ SURF ~ ORB

v_weapons: SIFT ~ SURF ~ ORB

In the imageset result that quality differed,
SIFT > SURF > ORB is the trend.

For others, no significant difference observed.