Computer Vision 1
Assignment 4: Image Alignment and Stitching

Minh Ngo[1] 10897402, Riaan Zoetmulder[2] 6072909

March 13, 2016

University of Amsterdam
[1]minh.ngole@student.uva.nl, [2] riaanzoetmulder@gmail.com

1 Image alignment

1.1 SIFT descriptor

SIFT descriptor can be an useful feature for the assignment task of image alignment and stitching because of its invariance to rotation and shifting.

The SIFT method firstly gets rid details without introducing new false positives by progressively blurring out versions of an image using Gaussian blurring. Features should still be detectable even when they aren't as clearly visible. It then proceeds to make sure that features are invariant to size of the image. It creates *octaves* of images that are resized and progressively blurred out.

To find key points that are scale invariant difference of Gaussian is applied to consecutive transformations mentioned previously. Once this is done, minima and maxima are found by checking whether the value of a pixel is a the smallest or highest value among neighbors. Only key points that are corners are considered.

Later points that are positioned along an edge or don't have enough contrast are gotten rid by analyzing the hessian matrix and thresholding an intensity measure. Finally, a small window around a point is divided to smaller regions to compute a histogram of orientations. To make the feature rotation independent it's firstly normalized and later subtracted by the key point's rotation to make other rotations to be relative to key points orientation. In addition, big values are thresholded to make it more illumination independent.

1.2 Keypoint Matching

The first part of this assignment consisted of matching keypoints that were found with SIFT. We did it by applying the *VL_UBCMATCH* function available in the *vl_feat* library that matches descriptors from two available sets together if the distance measurement between them is less than a threshold.

To visualize matches, firstly images are concatenated together. We sampled 50 matched candidates and visualized key points and drew lines between them. The results are displayed in figure 1. It can be noticed that with a decrease of the overlapping region between images more false matches appeared [Fig 2].

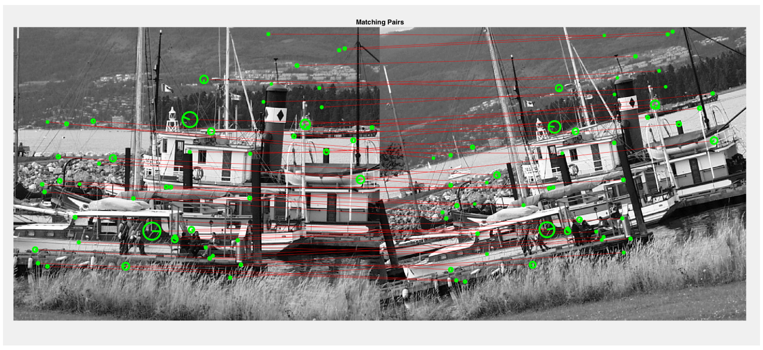

Figure 1: Keypoints for the 2 images with different orientations.

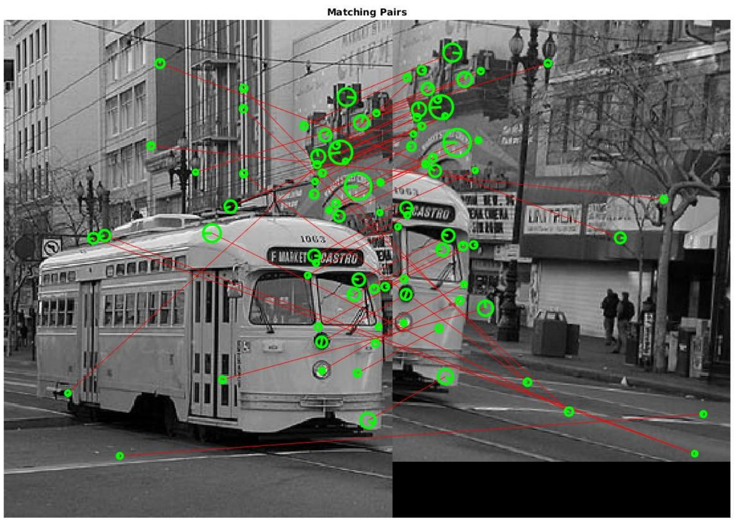

Figure 2: Keypoints matches for 2 images with a bus. It can be seen that keypoints extracted from the head of the bus have been matched correctly, but not keypoints from the street.

1.3 Best Transformation using RANSAC

To determine 6 variables required for the affine transformation we need at least 6 linear independent equations that correspond to matches of 3 keypoints. We sampled N keypoints several iterations and choose the affine transformation that produce less outliers (keypoints that are matched but are far away from each other).

For the report we have aligned img1.pgm to img2.pgm and vice rsa. Results of these experiments are presented in figures 3 and 4 respectively.

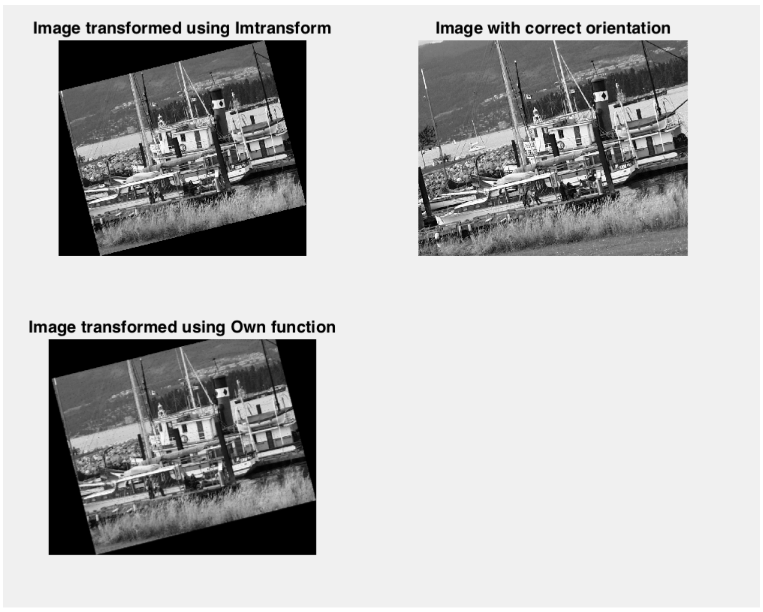
Image transformed using Imtransform
Image with correct orientation
Image transformed using Own function

Figure 3: best transformation to convert image 1 to the same orientation as image 2.

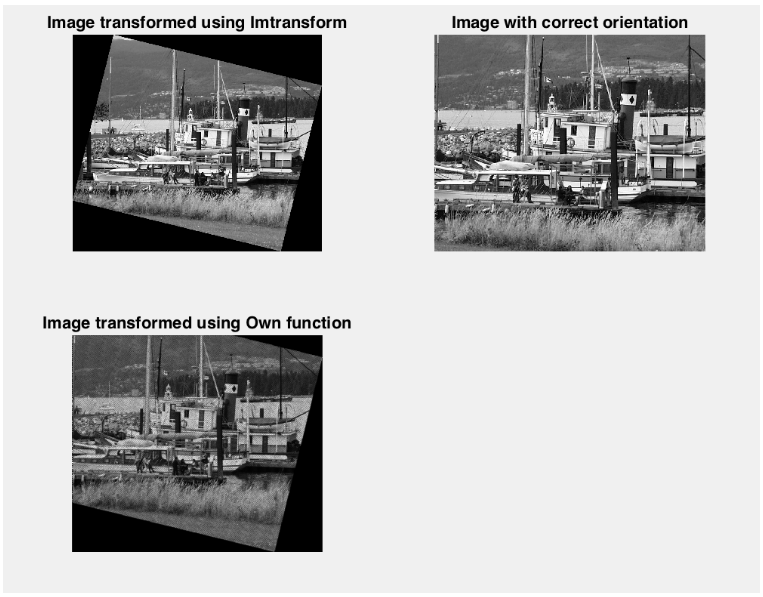
Image transformed using Imtransform
Image with correct orientation
Image transformed using Own function

Figure 4: best transformation to convert image 2 to the same orientation as image 1.

1.4 Number of iterations and matches required.

We did and experiment to check the influence of hyperparameters (number of interations in RANSAC, size of keypoint samples N) on the fraction of inliers obtained from our experiments. The results are reported in figure 5. As can be seen, lower sample sizes are always related to a lower number of inliers found, it takes at least an N of 3 to get a good proportion of inliers to total data points. Furthermore, having fewer iterations can also cause a lower amount of inliers, especially when it is combined with a smaller sample size. Having a higher sample size but fewer iterations can cause fewer inliers because the chance that a data point in the sample is an outlier becomes bigger. As such the RANSAC algorithm won't be able to correct by resampling the dataset.

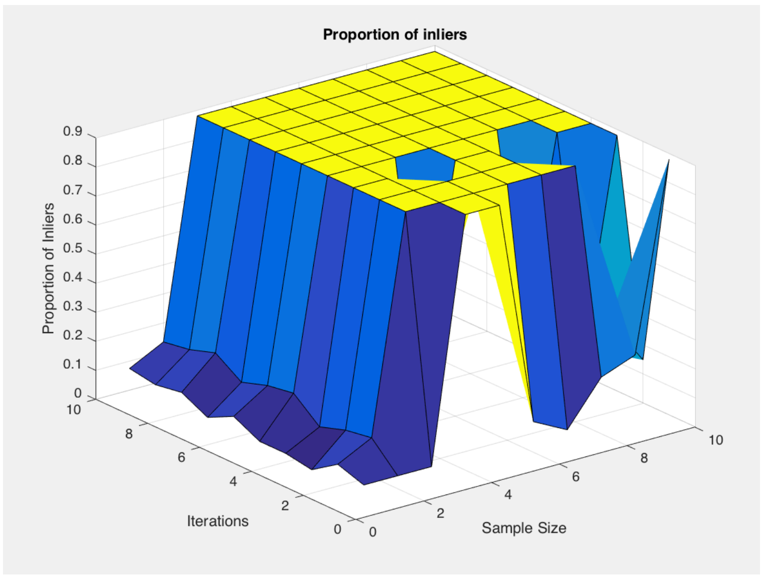

Figure 5: best transformation to convert image 2 to the same orientation as image 1.

2 Image stitching

Let we have images A and B. We do nearest neighbor interpolation for the image A after transformation to fill unknown gaps between transformed pixels. For the purpose of stitching, the bounding box C of images together in the coordinate system of the image B is computed. In the case of having a negative coordinate for the origin corner, images will be shifted to be in the coordinate system where all pixels have positive coordinates, a new stitched result will have the size of the computed bounding box. We fill pixels with values that are unknown in the image B by values from the transformed image A [Fig 6].

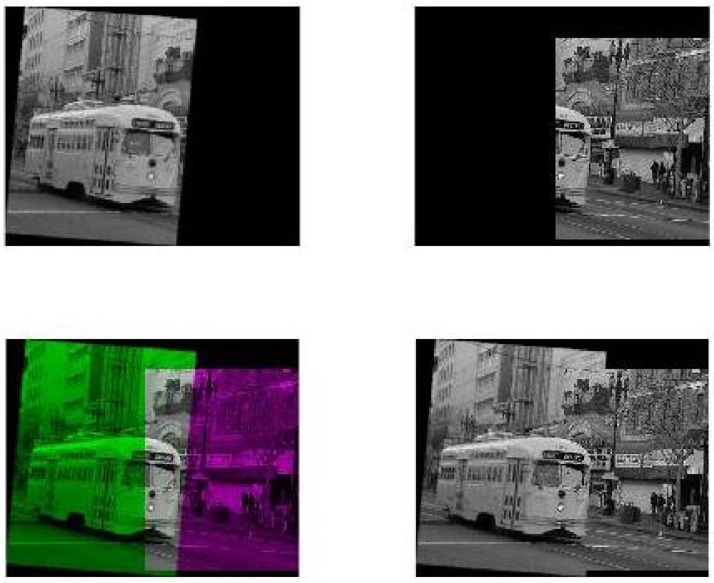

Figure 6: Stitched bus results. Transformed left.jpg (Left-Top), right.jpg (Right-Top), left.jpg and right.jpg matching (Left-Bottom), stitched result (Right-Bottom). All images in the coordinate system of the bounding box C.