
Lab 2 - Neighborhood Processing & Filters

Jeroen Baars - 10686320

Bob van den Hoogen - 10420169

Introduction

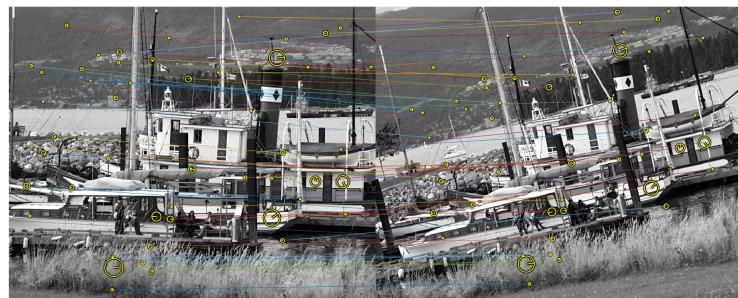
In this practice we will write a function that takes two images as input and computes the affine transformation between them. In the next part, we will write a function that takes two images as input and stitches them together.

1 1 Image Alignment (60pts)

Question 1

For the matches, the separate function plotmatches.m is used.

Figure 1: Matching points between two images



In figure 2 can the transformation with nearest neighbor implementation be seen. In figure 3 are the same transformation parameters used but now with the built in matlab function. In figure 4 and figure 5 the same methods are used but the other image of the pair is transformed.

Figure 2: Boat 1 is transformed to boat 2 with own affine transform



Figure 3: Boat 1 is transformed to boat 2 with matlab affine transform



Figure 4: Boat 2 is transformed to boat 1 with own affine transform



Figure 5: Boat 2 is transformed to boat 1 with matlab affine transform



Question 2 1. We need to solve the affine transformation for 50 matches.

2. This depends on the dataset, there is an formula for getting the number of iterations which is dependent on the dataset.

$$k = \frac{\log(1 - p)}{\log(1 - w!n)} \quad (1)$$

where k is the amount of iterations, p is the probability of choosing only inlier's and w is the probability of choosing an inlier as point.

2 2 Image Stitching (40pts)

Question 1

We didn't come to this part, we got blind to how to perform this application. Probably was an easy part if getting the 'eureka' moment.

References