

Project I: Feature Detection and Matching

Wenzhen Zhu

Feature Descriptor

simple feature

Simple feature descriptor is based on a 5*5 window. For each pixel, we centered there and push all the 25 values into a vector called window. If it's out of boundary, we assign the value as 0. Then the window vector is our feature f's data.

my feature

My feature is also based on a 5*5 window. However, we care about the mean and standard deviation of the 25 size vector. I assign the mean, and standard deviation to the feature f's data.

Design Approach

Hard Code

number	1	2	3
feature type	simple	harris	my_feature
match type	SSD	ratio	□

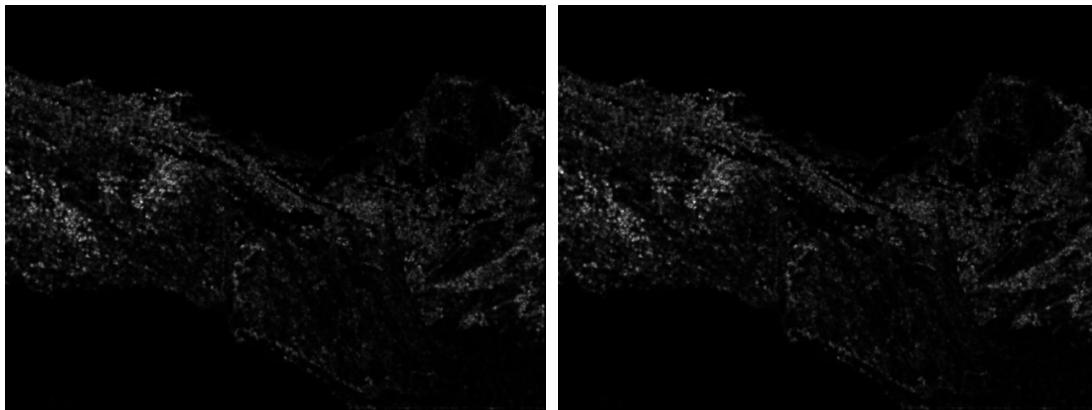
computeFeatures

Has 3 feature type, it compute features via 3 options, simple, Harris, or myfeature.

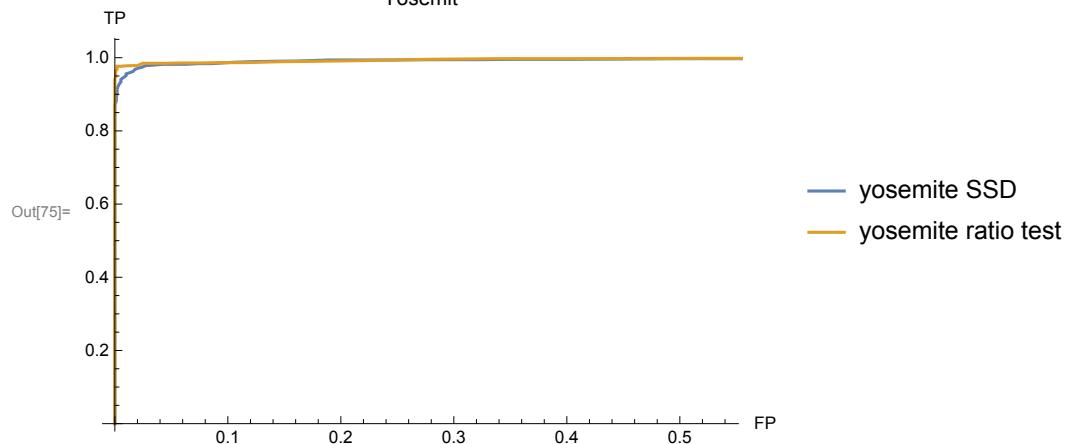
computeLocalMaxima

I looped through the image, with 5*5 window, for each loop, we get a vector contains the 25 pixel's intensity. I used max_element to find the maximum. Then I use the window for loop to write the image into a destination image based on threshold and the relationship with the local maxima.

Yosemite

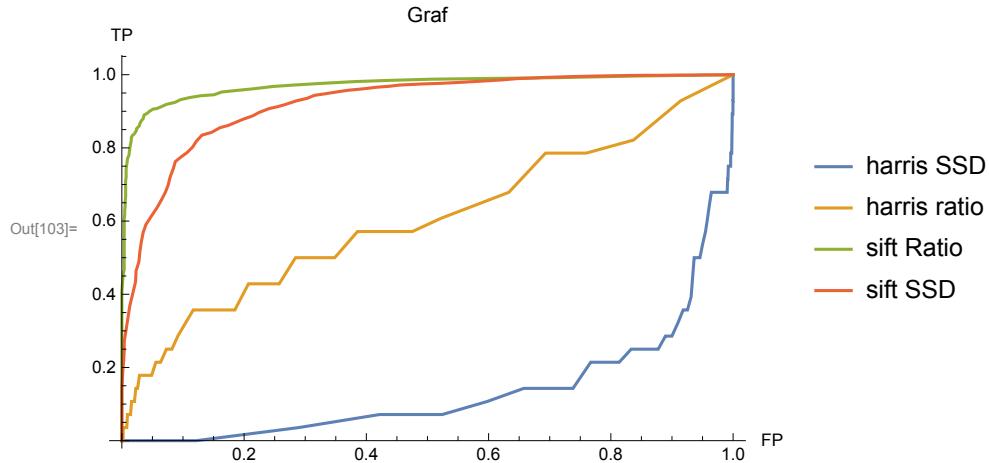


Yosemit



Graf





Average AUC

Simple Feature

Use **SSD** match (match_type = 1)

□	wall	graf	bikes	leuven
matching image 1 with image 2	0.625	0.638	0.6	0.526
matching image 1 with image 3	0.531	0.604	0.83	0.708
matching image 1 with image 4	0.479	0.520	0.427	0.429
matching image 1 with image 5	0.622	0.540	0.44	0.571
matching image 1 with image 6	0.421	0.422	nan	0.570
average AUC	0.537	0.54	0.567	0.561
threshold	0.5	0.1	0.5	0.5

Use **ratio** match (match type = 2)

□	wall	graf	bikes	leuven
matching image 1 with image 2	0.618	0.638	0.552	0.7064
matching image 1 with image 3	0.601	0.604	0.571	0.6696
matching image 1 with image 4	0.489	0.520	0.206	0.6649
matching image 1 with image 5	0.507	0.540	0.917	0.6625
matching image 1 with image 6	0.544	0.422	0.427	0.6423
average AUC	0.552	0.540	0.534	0.669
threshold	0.02	0.1	0.05	0.02

my features

Use **SSD** match (match type = 1)

□	wall	graf	bikes	leuven
matching image 1 with image 2	0.44	0.67	0.52	0.63
matching image 1 with image 3	0.51	0.63	0.54	0.56
matching image 1 with image 4	0.65	0.52	0.21	0.51
matching image 1 with image 5	0.57	0.56	0.67	0.54
matching image 1 with image 6	0.53	0.47	0.53	0.69
average AUC	0.54	0.57	0.494	0.58
threshold	0.5	0.5	0.5	0.5

Use ratio match (match type = 2)

□	wall	graf	bikes	leuven
matching image 1 with image 2	0.59	0.62	0.69	0.65
matching image 1 with image 3	0.64	0.80	0.77	0.51
matching image 1 with image 4	0.38	0.73	0.05	nan
matching image 1 with image 5	0.67	0.53	0.32	0.28
matching image 1 with image 6	0.52	nan	nan	nan
average AUC	0.56	nan	nan	nan
threshold	0.15	0.45	0.55	0.6

Description

My ratio test is extremely slow then the threshold is set to be small. it's nested 4 layer for-loop. And for finding local maxima, I went through the 5*5 window twice, first is to compute local maxima, 2nd time is to write pixel into the destination image. I need more time to come up with a better idea to implement this.

Demo

