CS381V : Visual Recognition - HW1

Ashish Bora

February 2016

1 Introduction

In this assignment we explore local feature matching using SIFT[1] and its application to instance recognition. We are given a template image, and our task is to detect if the template object is present in another scene image or not.

Please see README.txt for instructions on how to run code for specific parts and change parameters. The README file also explains how the resulting images are organized. Here we present representative experimental results and some analysis.

2 Feature extraction and matching

For template as well as scene image, we extract SIFT features. For every template feature, we find the scene feature nearest in terms of Eucledean distance in SIFT space.

3 Reducing noise in matching

Here we aim to automatically remove bad matches that occur, possibly due to imperfections in SIFT or the mactching process. For any given SIFT feature, let NND and SNND be the distance to the first and second nearest neighbour (in the SIFT space) respectively. We explore the following two strategies.

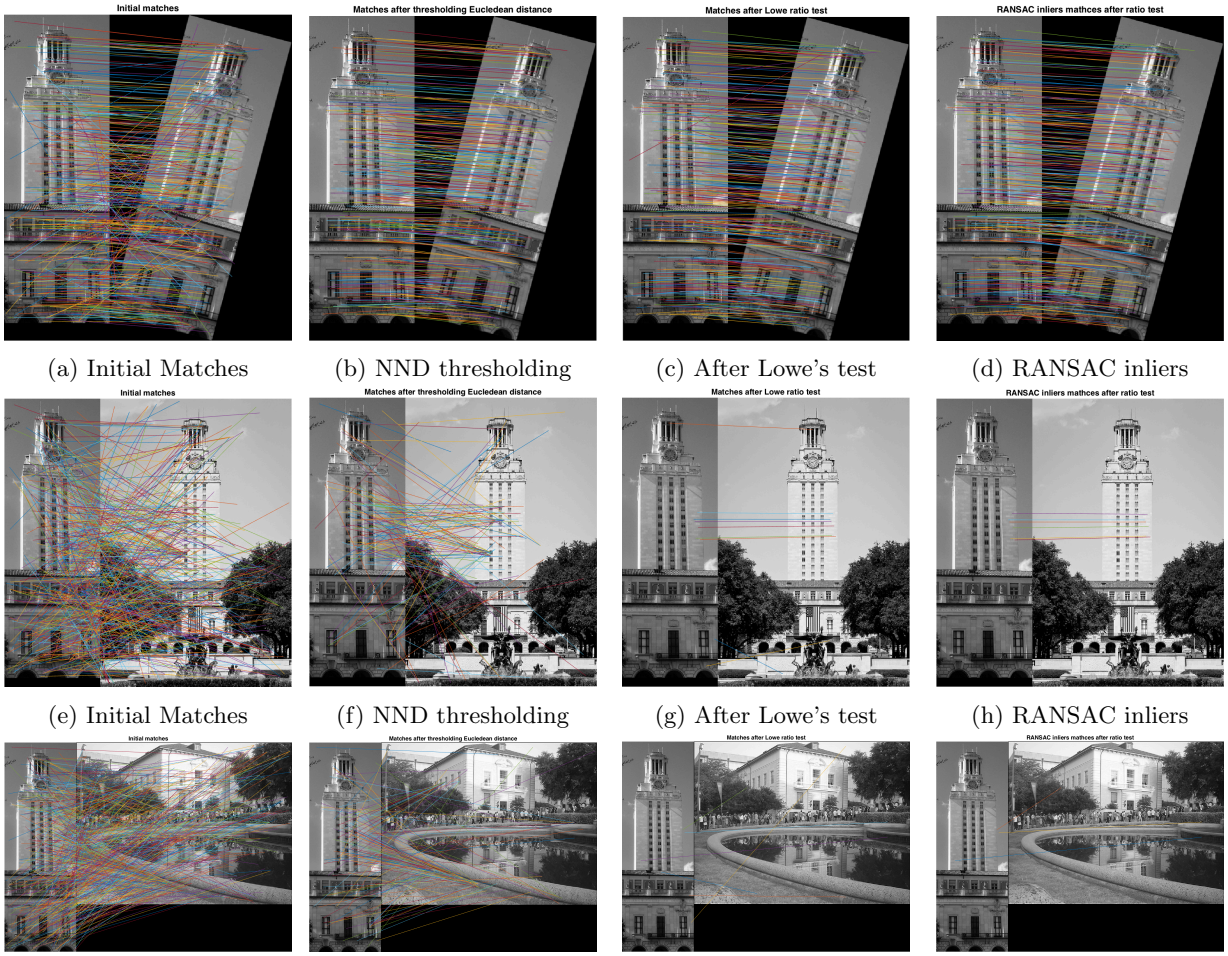
1. **Thresholded nearest neighbours :** If NND is larger than a threshold, we discard that match. The intuition here is that true matches will have a very small distance and false matches will have a comparatively larger distance. Thus we can threshold near the average.
2. **Lowe's ratio test :** If NND/SNND is larger than a threshold, we discard that match. The intuition here is that outliers will have very similar NND and SNND since they will be far away from all other points.

4 Matching with RANSAC

We assume that scene image object is an affine tranform ($Ax + b$) of the template image (along the lines of [1]). We estimate the parameters A and b using RANSAC.

5 Results and Observations

In Fig 1 we show show the results when these methods are applied to the given scene images. For the first test, we use 0.8∗`mean`(NND) as a threshold. For the second test, we set the threshold at 0.6. Finally we apply RANSAC on the matches that survive Lowe's test with inlier threshold of 5 on the squared distance.



(a) Initial Matches (b) NND thresholding (c) After Lowe's test (d) RANSAC inliers

(e) Initial Matches (f) NND thresholding (g) After Lowe's test (h) RANSAC inliers

(i) Initial Matches (j) NND thresholding (k) After Lowe's test (l) RANSAC inliers

Figure 1: Reducing noise in matching

Observations:

1. We can see that the intial matching has a lot of false matches.
2. There is still a lot of clutter left after thresholding NND. This is particularly visible in the second and third scene image.
3. Lowe's test is able to throw away most of the wrong matches. There are still a few non-matches left (as in the first example), but these are very few as compared to the first method. On the third scene image, Lowe's test is able to throw away almost all the matches, which is good since this is a negative exmaple.

In summary, Lowe's ratio test performas better than thresholding NND. We thus use this method for noise removal before the final affine-RANSAC verification. RANSAC removes inconsistent matches further. For first and the second scene image, it can be seen that only those matches that can be explained by a similar

affine transform remain. For the third image, all but 3 matches are left, which is the minimum since we are fitting with 3 datapoints.

6 Object Detection

If an object is present and if we detect enough number of features in both template and scene image, we can expect a lof of them to be consistent with a single affine tranform. If the object is not present, it would be very unlikely for many matches that survive the ratio test to also agree on a single affine transform. Thus, this suggests that the number of RANSAC inliers is a good indicator of presence of an object.

For the scene images, we run the detection algorithm with RANSAC inlier threshold of 6. For third image the result is negative. For the first two images, we get a positive result. For these cases, we tranform the template corners and display it overlayed on the scene image in Fig 2. We also report the number of RANSAC inliers.

(a) Template Image (b) ransac-inliers = 375 (c) ransac-inliers = 7

Figure 2: Detection results - UT Tower

Observations :

As can be seen, the template detection works pretty well . Here is a potential idea to get a better matching:

1. The scale of every feature is available to us. The magnitude of eigenvalues of A will give us an idea about the relative scale between the images. Same relative scale should hold for true inliers as found by RANSAC. So this can be used to modify the inlier criteria in RANSAC to get a more robust matching.
2. For non-deformable objects, both eigenvalues of A should have similar magnitude. This can be used as another sanity check.

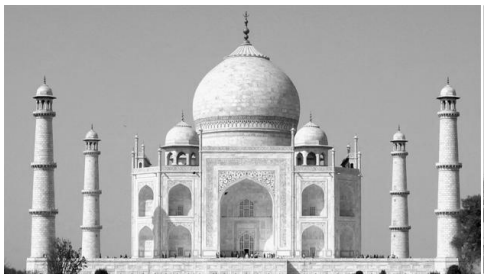
7 Additional Experiments

7.1 The Taj Mahal

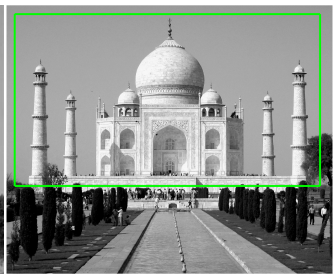
The Taj Mahal is a challenging monument from detection point of view because of the following:

1. Symmetry and self-similarity (pillars etc) (See Fig 3e)
2. Clear water and reflections around the monument (See Fig 3d)
3. Very similar (dome like structures) around the mounument (See Fig 3f)

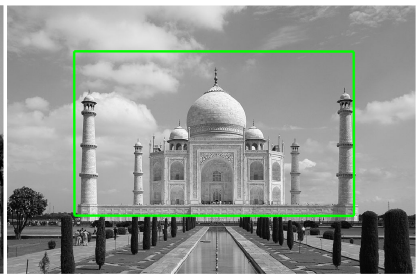
We run the detection algorithm on images of the Taj Mahal taken from various viewpoints [2]. We report some successes followed by interesting failure cases in Fig 3.



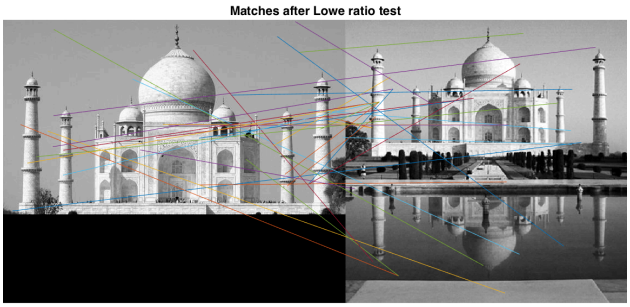
(a) Template Image



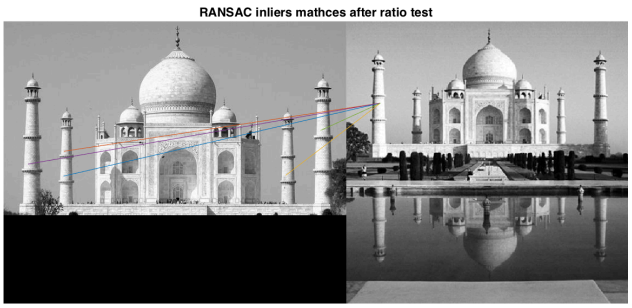
(b) success



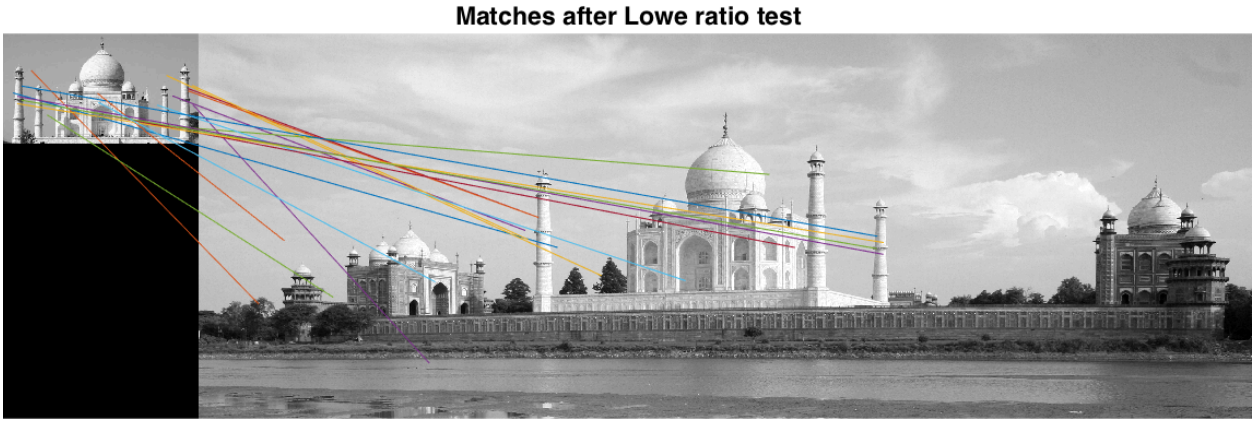
(c) success



(d) reflections



(e) low eigenvalue problem



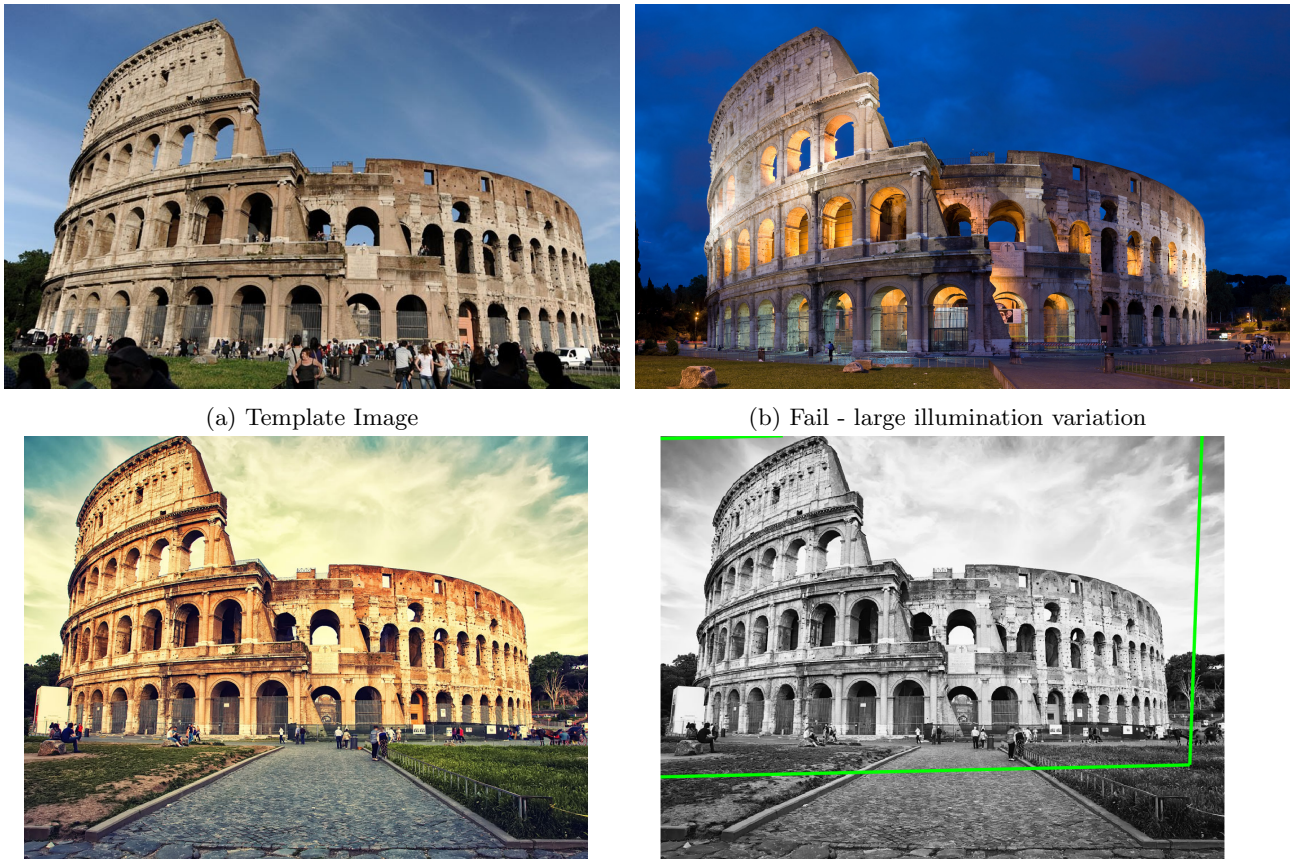
(f) other similar structures

Figure 3: Detection results - Taj Mahal

A particular issue is that of low eigenvalues of A which happens if many template faetures are all matched to a small region in the scene. This issue can be seen in action in Fig 3e. One way to avoid this problem is

to use the knowledge of object scales. We can assume that the images we are working with are not larger than $5k$ resolution (5000×5000 pixels) and not smaller than 20×20 pixels. Thus, the scale ratio can at most be 250 and hence the absolute value of eigenvalues of A must be within $\left[\frac{1}{250}, 250\right]$. This test can also be combined with the previously mentioned scale matching to give a better detection algorithm.

7.2 The Colossuem



(a) Template Image

(b) Fail - large illumination variation

(c) High illumination variation

(d) Successful detection

Figure 4: Detection results - Colossuem

8 Optional sections

1. Fig 4c is the image of the Colossuem which we try to match against the template image (Fig 4a). Notice the high illumiation variation between the two. Despite that we successfully find the match as shown in Fig 4d. In Fig 4b we show very high degree of illumiation variation (day vs night and internal lighing). In this case we are not able to detect the object.
2. In Fig 5, we show that when we try to match the Taj Mahal with the Colossuem, we get as many as 13 matches after the Lowe's test, but the RANSAC affine verification step is able to eliminate them, thus avoiding false positive.

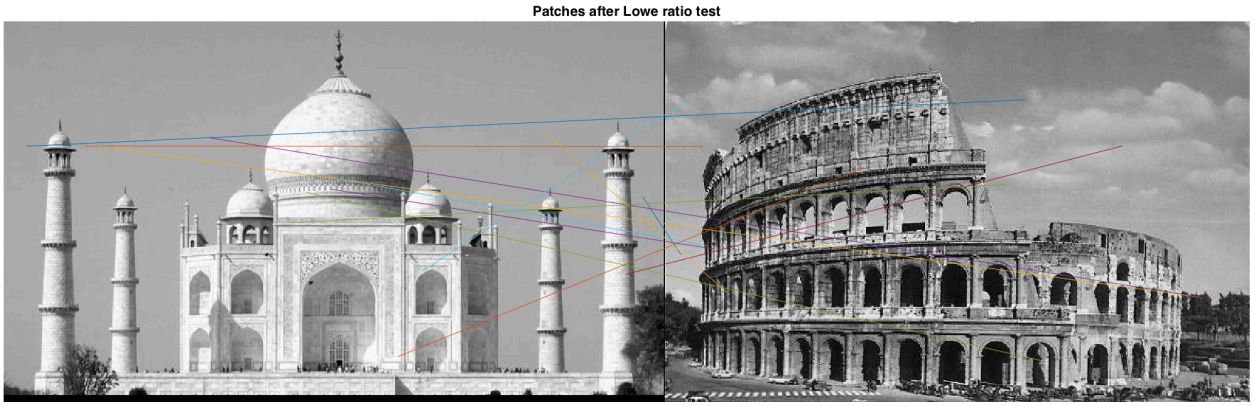


Figure 5: False matches survive Lowe's ratio test

References

[1] Lowe, David G. "Object recognition from local scale-invariant features." Computer vision, 1999. The proceedings of the seventh IEEE international conference on. Vol. 2. Ieee, 1999.

[2] Images of the Taj Mahal and the Colosseum obtained from Google Image Search.