EQ2425 Analysis and Search of Visual Data

EQ2425, Project 1

Chunyu Wang chunyuw@kth.se

Jingwen Liu jingwenl@kth.se

September 16, 2024

Summary

Images contain robust interest points. To test the repeatability of keypoint detectors and descriptors, we modified the original image by rotating and rescaling. We applied fixed threshold, nearest neighbor and nearest neighbor distance ratio keypoint matching methods, and compared the results between SIFT and SURF.

1 Introduction

Repeatability refers to the good matches of keypoints between original figure and modified one. To test the robustness between SIFT and SURF, we modified the original figure by rotating and rescaling it, and plot the repeatability vs. those factors.

We then tested the robustness of image feature matching for both methods. we applied fixed threshold, nearest neighbor, and nearest neighbor distance ratio matching methods between two figures for the same object.

2 Problem Description

2.1 Robustness of Keypoint Detector

In this section, we would apply SIFT and SURF to the same image. We tune the parameters, peak and edge thresholds of the SIFT keypoint detector, and the strongest feature threshold of the SURF, to generate only a few hundreds of keypoints for both methods.

We then make adjustments to the original figure by rotating and scaling it under multiple times. We would then conduct SIFT and SURF again and generate new keypoints with the same parameters. We compare the newly generated keypoints with the original ones to check the repeatability over the changes.

2.2 Image Feature Matching

In this section, we implement and compare three feature matching algorithms—"fixed threshold," "nearest neighbor," and "nearest neighbor distance ratio"—to match two images captured from different perspectives of the same scene. We then analyze and discuss the performance of each algorithm. First, we extract a few hundred SIFT features from the test images. Show the feature keypoints superimposed on top of obj1 5.JPG and obj1 t5.JPG. For the following

three algorithms, we plot side-by-side views of the query image obj1 5.JPG and the database image obj1 t5.JPG with matched feature points connected by lines. Then we visually examine the matched features and comment on the performance of these algorithms.

The "fixed threshold" matching algorithm matches feature points between two images by calculating the distance between them. During implementation, feature points and their descriptors are first extracted from both images, and then the distance between each feature point in one image and all feature points in the other image is computed. If the distance between two points is smaller than a predefined threshold, they are considered a match. The "nearest neighbor" matching algorithm identifies the best match for each feature point in one image by finding the feature point in the other image that has the smallest distance in terms of their descriptors. For each point, the closest match is selected as the corresponding point. The "nearest neighbor distance ratio" matching algorithm improves upon the nearest neighbor approach by comparing the distance of the best match to the distance of the second-best match. If the ratio of these two distances is below a predefined threshold, the match is considered reliable. Finally, we extract a few hundred SURF features from the test images. Then we use the "nearest neighbor distance ratio" matching algorithm and compare the result to SIFT.

3 Results

3.1 Robustness of Keypoint Detector

3.1.1 Part a

We set $SIFT\_peak\_threshold = 0.17, SIFT\_edge\_threshold = 5, SURF\_hessian\_threshold = 7500$. The SIFT and SURF methods are implemented to the original image with these parameters. 436 and 351 key points were generated with SIFT and SURF respectively.

Figure 1: SIFT features of original image.

Figure 2: SURF features of original image.

3.1.2 Part b

In this part, we rotate the original image and apply SIFT and SURF with the same parameters. We match the key points from the rotated images to the original one with respect to each method, and plot the repeatability over rotation angles. From 3, we noticed that SIFT changes

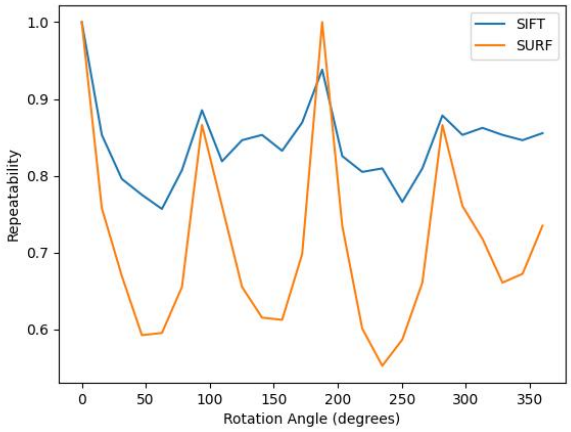

Figure 3: Rotation vs. Repeatability.

less than SURF when rotation angle increases, indicating SIFT is more robust than SURF under rotations.

3.1.3 Part c

In this part, we scale the original image and apply SIFT and SURF with the same parameters. We match the key points from the scaled images to the original one with respect to each method, and plot the repeatability over scaling factors. From 4, we noticed that SIFT changes

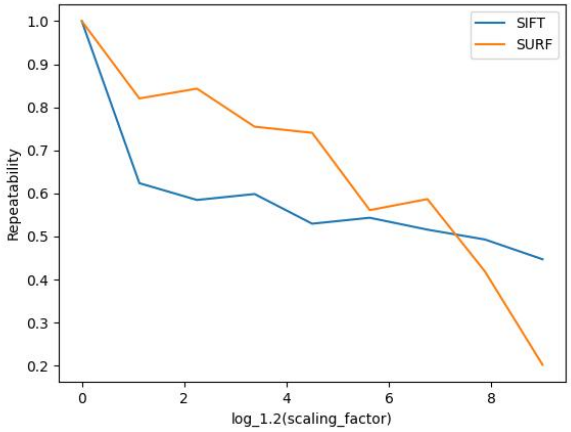

Figure 4: Scaling vs. Repeatability.

much when scaling factor increases by a little but keeps steady when it continues increasing, while SURF performs better first but worse as the scaling factor increases more. We conclude that SIFT is more robust to scaling than SURF.

3.2 Image Feature Matching

(a) We first extract a few hundred SIFT features from the test images and show the feature keypoints superimposed on top of the images. We can see in Figure5 and Figure6, we get 436 and 645 keypoints from images obj1 5.JPG and obj1 t5.JPG.

Figure 5: SIFT features of query image.

Figure 6: SIFT features of database image.

(b) We implement the "fixed threshold" matching algorithm to the keypoints gained in (a). After trying some thresholds, we found that the result is good when the threshold is 170. The matching results are show in Figure7, the number of matches is 42. When the threshold is too small, for example 150, the result is as shown in the Figure8, with too few matching points. When the threshold is set too high, for example 190, the result is as shown in the Figure9, with more matching points but too many mismatches.

Figure 7: Fixed Threshold Matching Algorithm on SIFT with threshold 170.

(c) We implement the "nearest neighbor" matching algorithm. The result is shown in Figure10. We can see that the matching lines in the images are more cluttered. There are both more matches and mismatches. The "nearest neighbor" algorithm finds a match for every point, so there will be a lot of mismatches caused by those points that only appear once.
(d) We implement the "nearest neighbor distance ratio" matching algorithm. After trying some thresholds, we found that the result is best when the threshold is 0.8. The matching results are show in Figure11, there are 81 matches and 2 mismatches. When the threshold is too high, for example 0.85, the result is as shown in the Figure12, with too many mismatching points.

Figure 8: Fixed Threshold Matching Algorithm on SIFT with threshold 150.

Figure 9: Fixed Threshold Matching Algorithm on SIFT with threshold 190.

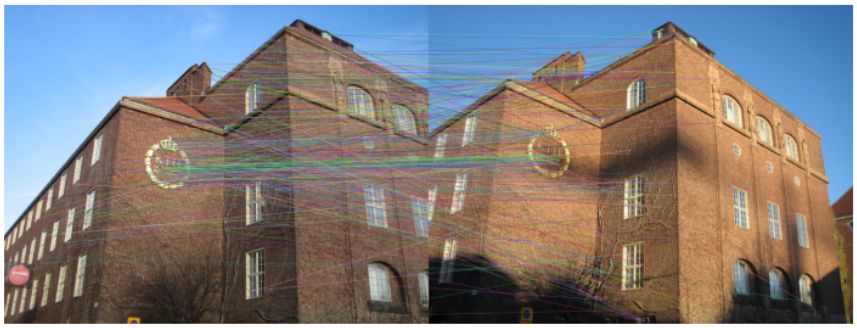

Figure 10: Nearest Neighbor Matching Algorithm on SIFT

When the threshold is set too low, for example 0.7, the result is as shown in the Figure13, with too few matches, only 54 matches.
(e) We extract a few hundred SURF features from the test images and use the "nearest neighbor distance ratio" matching algorithm. The result is shown in Figure14. We adjusted the threshold from 0.65 to 0.85, the results all show many mismatches. And the number of correct matches is not as many as SIFT features. So we can conclude, SURF is not as effective as SIFT when using the "nearest neighbor distance ratio" algorithm.

Figure 11: Nearest Neighbor Distance Ratio Matching Algorithm on SIFT with threshold 0.8.

Figure 12: Nearest Neighbor Distance Ratio Matching Algorithm on SIFT with threshold 0.85.

Figure 13: Nearest Neighbor Distance Ratio Matching Algorithm on SIFT with threshold 0.7.

4 Conclusions

In conclusion, the SIFT algorithm performs better than SURF for both parts. SIFT performs worse than SURF when scaling factor increases a bit, but it has overall better performance and robustness than SURF.

The overall performance of the SIFT algorithm is good when combined with different matching algorithms such as fixed threshold, nearest neighbor, and nearest neighbor distance ratio.

Figure 14: Nearest Neighbor Distance Ratio Matching Algorithm on SURF with threshold = 0.8.

When using the "nearest neighbor distance ratio" matching algorithm, SURF results in more mismatches compared to SIFT. The SIFT algorithm delivers superior accuracy in the context of this project.

Appendix

Who Did What

The Robustness of Keypoint Detector section is mainly conducted by Chunyu Wang, but discussed together. The Image Feature Matching section is mainly conducted by Jingwen Liu, but discussed together.

References

[1] Rafael C. Gonzalez and Richard E. Woods, *Digital Image Processing*, Prentice Hall, 2nd ed., 2002

[2] Tobias Oetiker et al., *The Not So Short Introduction to LaTeX 2ε*, Available: http://tobi.oetiker.ch/lshort/lshort.pdf, Last accessed: March 17, 2009