2017

Feature detection and matching

COMPUTER VISION – LAB 3 & 4

JOSEP FAMADAS ALSAMORA / JORDI RIU VICENTE

UNIVERSITAT DE BARCELONA

UB | Facultat de Matemàtiques i Informàtica

Table of contents

Table of Figures

1- Feature detection

Figure 1: SIFT features on a grayscale image

Question 1:

In *Figure 1* we have selected 4 different features observed at different scales and positions:

Feature 1: It is detected when the image size is reduced drastically as we can see from the size of the circle. This is due to the fact that when the image is at the original size a 16x16p window could not observe any variation in intensity as it is a flat wall, but, when the image is scaled the roof edge and the door one appears in the 16x16p window and therefore we detect a feature.

Feature 2: This case is similar to the previous one, but given the fact that the plain area is smaller, so it is the feature window because the top and bottom edges are detected with less reduction.

Feature 3: As we can observe this is a crucial feature because it detects not only an edge but a corner and it delimits the skyline. Again, we only need a very small reduction because the feature center is near the corner.

Feature 4: We have selected this feature to remark that this is not an important one. It is selected nearly at the original size and only shows a detail of the roof which is repeated hundreds of times.

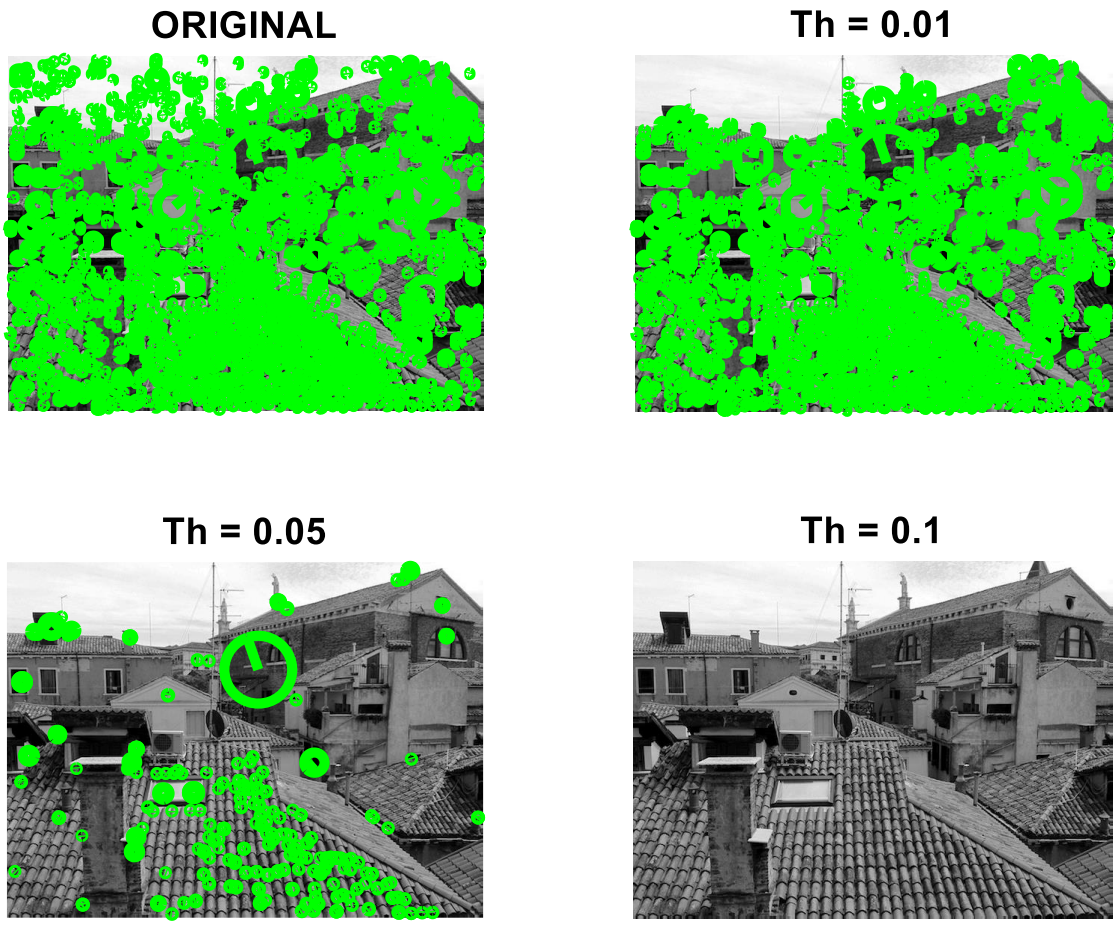

Figure 2: SIFT features with different DoG thresholds

Question 2:

As it can be observer in *Figure 2*, a threshold of 0.01 is not really significative and just discards as features the ones detected at the sky with no threshold.

Question 3:

In *Figure 2* we have selected two other thresholds. In the image we observe that, as expected, the larger the threshold value, the lesser detected features. This happens because at the original picture the features for which the determinant of its DoG (Difference of Gaussian) matrix is very small are accepted. These features could appear because of noise so a threshold is defined to avoid them.

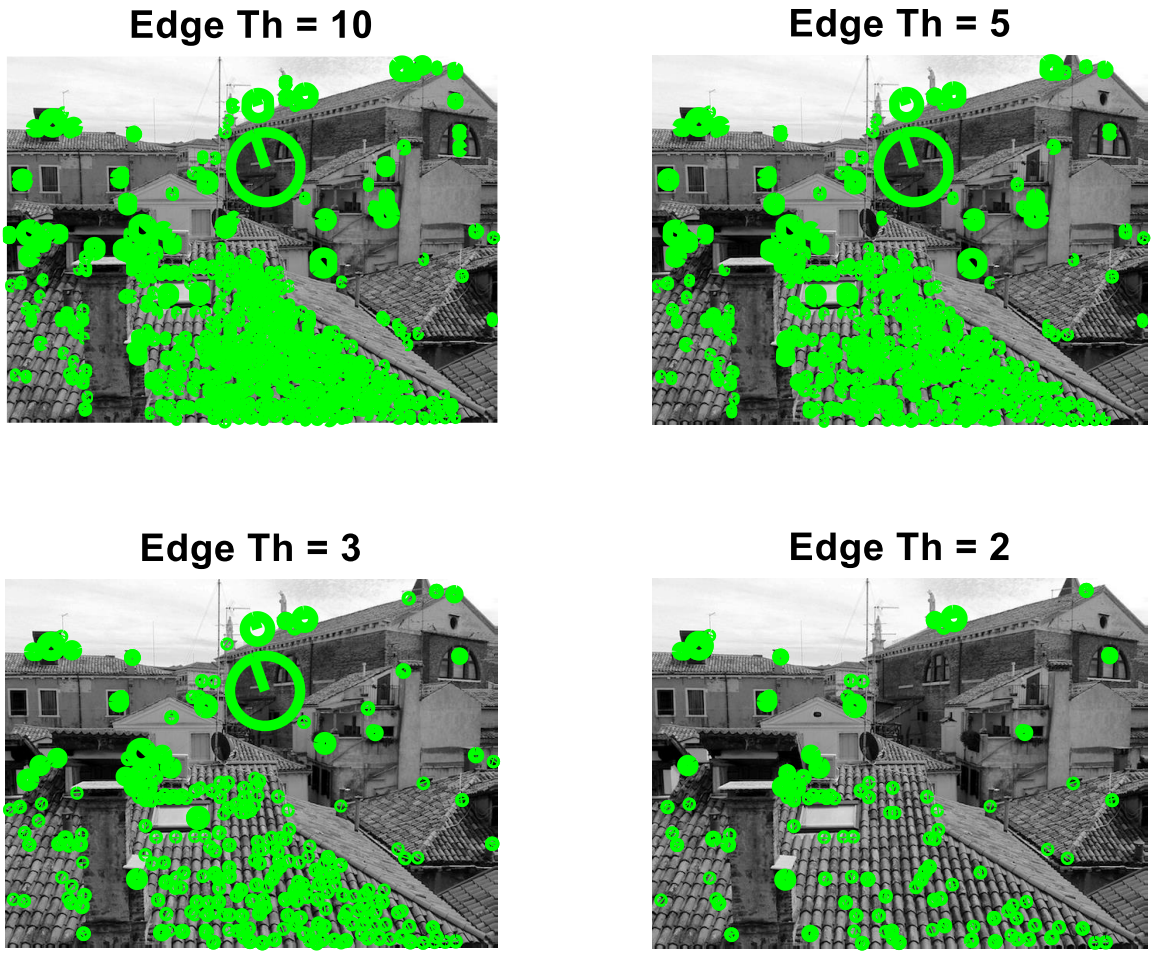

Figure 3: SIFT features with different Edge thresholds

Question 4:

The edge threshold takes into account a value that gives the grade of roundness of an edge. Large values mean a straight edge, and the lower the value is, the higher the curvature in a short space until the lowest values which correspond to sharp corners.

The threshold discards the edges with low curvature and leaves only the corners.

As we can observe in *Figure 3* we can appreciate that with a threshold of 2, the remaining features are mainly sharp corners.

2- Feature matching

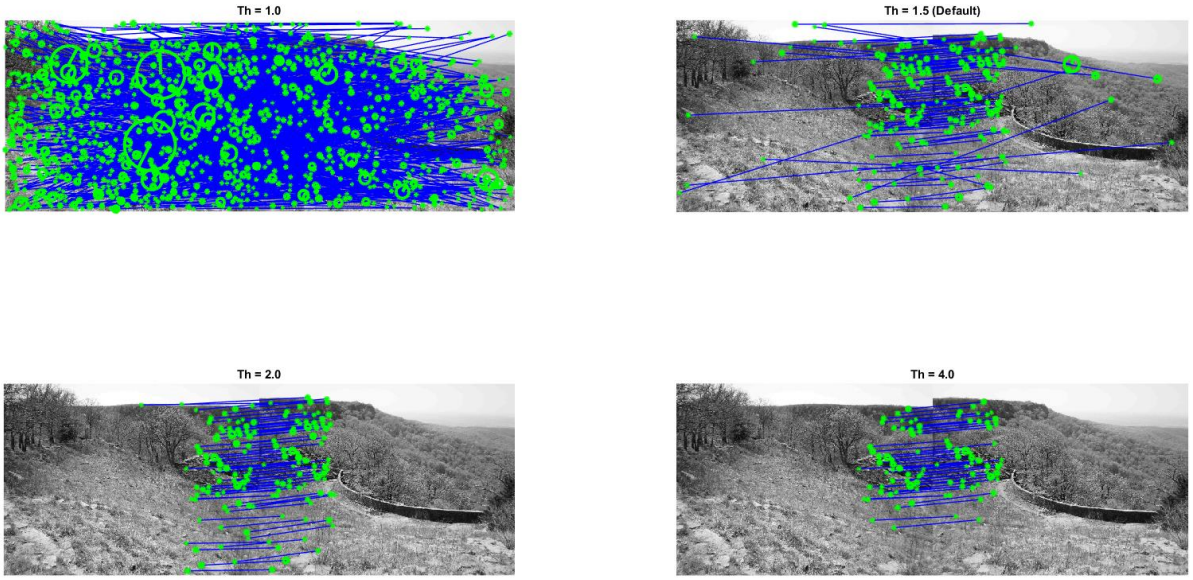

Figure 4: Matching between left and right image features

Question 5:

The ubcmatch function matches a keypoint of an image with the one from the other image that is most similar to it (minimum distance between them). Even if a keypoint has no match on the other image, the function will find a mismatch for it, so to make sure that the match of two keypoints is a true match we define a threshold. The threshold ensures that the distance between a keypoint and its closest match and between the same keypoint and its second closest match follows:

$$\frac{d(keypoint - closest)}{d(keypoint - 2nd\ closest)} > Threshold$$

If this relation is satisfied, the match is taken as a true match. As we can observe in *Figure 1*, as it was expected, the higher the threshold, the more restrictive are the matches.

Figure 5: Linear translation with no threshold

Question 6:

We can appreciate in *Figure 5* that a linear translation computed with the default threshold is not really good, for example, one can appreciate the blue arrows that signal the same point in both images but they are not connected by the model.

On the other hand, if we compute the linear model taking into account only the best matches (the ones obtained with a threshold of 4 in the previous question) the results are far better.

Figure 6: Linear translation with a threshold of 4

Even though this model is better, we think that the picture not only responds to a linear translation but also to a resize, because it looks like the second picture has been taken from a little bit further from the rocks.

3- Panorama creation

Figure 7: Panoramic image result of superposing the two previous ones

When creating the panoramic image using the method proposed in the document, we can appreciate that the matching is not perfect mainly because, as commented before, in our opinion these two images do not correspond to an exact linear translation.

In the zip file delivered we have computed the image using 10, 20, 30 and 40 matches to compute the mean displacement and some differences are appreciated.