Computer Vision

Exercises of Lab 2

Exercise 2.1: Multi-scale Harris detector

Open Exercise2.1.m and read through the code.

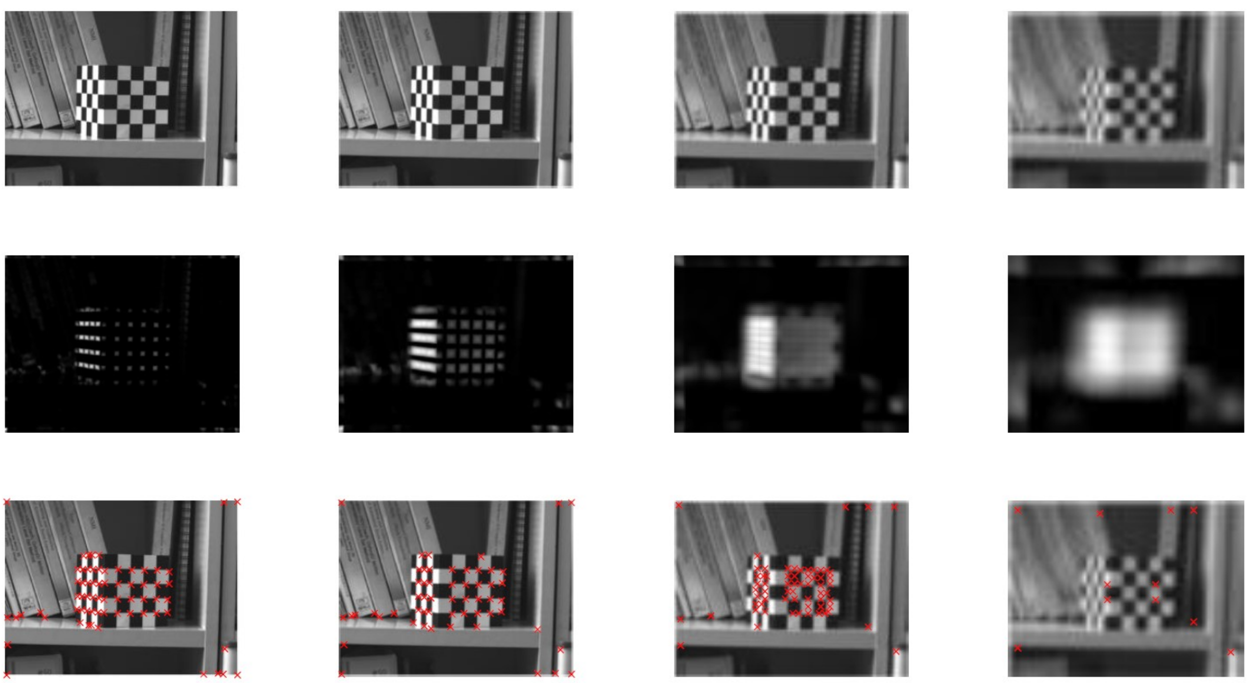
What do the different images in the top row of Figure 1 represent?

Implement the Harris corner measure inside the for-loops in line 38. Use the definition on slide 48.

```
for x = 1:size(I,1)
    for y = 1:size(I,2)
        % Exercise: Find the Harris corner measure (Harris operator), R


        % R(x,y) = ???
    end
 end
```

The result should look something like the following:

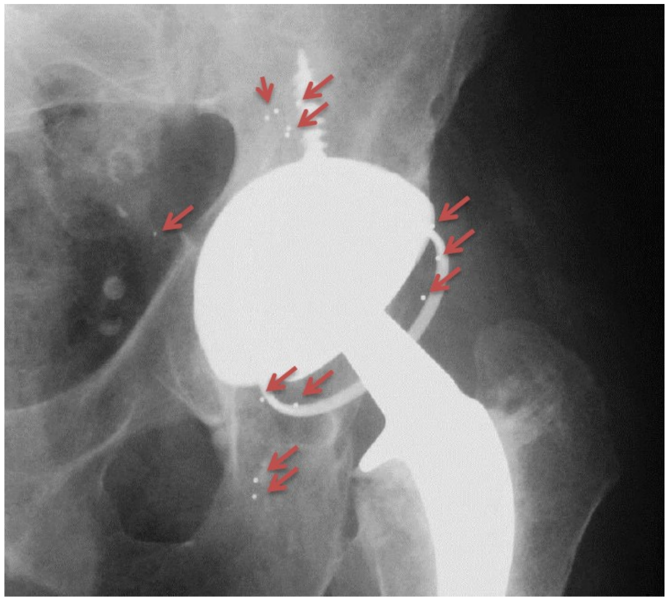


The middle row shows the corner response function (R), and the bottom row shows the detected corners. How are the corners detected from R?

If time allows, experiment with the feature detection and extraction functions in Matlab's Computer Vision Toolbox.

Exercise 2.2: Blob detector

This exercise is inspired by a recent research project. Your task is to detect metal bone markers in an X-ray image of a hip implant. (The markers are injected into the patient's bone and used as a reference to monitor if the implant loosens over time).



First, download VLFeat binary package (including a Matlab toolbox) from vlfeat.org and extract the binary (e.g. using [7-Zip](7-Zip)).

Open Exercise2.2.m and change the variable *VLFEATROOT* to the unpacked path at your hard drive. Run the code and inspect the figures carefully. Figure 1 presents a Gaussian pyramid and Figure 2 represents a pyramid of Difference of Gaussians (DOGs).

Why does Figure 2 only have 5 subdivision/suboctaves per octave (ignore the dark blue), whereas Figure 1 has 6?

The first step of the SIFT algorithm extracts interest points (blobs). This is done by detecting local extrema in the DoG pyramid. Each pixel in a suboctave is compared to its eight neighbors in the current image and nine neighbors in the suboctave above and below. Study the code that finds local extrema and inspect Figure 3. Here, the local extrema (red=maxima, blue=minima) are plotted on top of the 1st suboctave of the 1st octave in scale space.

Find out which scale (octave and suboctave) that best detects the bone markers. Change the variables octave and suboctave to visualize the extrema.

Some of the extrema found by the blob detector represent insignificant points such as edges. To remove such undesired feature points, the SIFT algorithm uses the principles underlying the Harris corner detector. Your task here is to implement a Harris corner detector that operates on the DoG pyramid (not the original image!). For this, calculate the Harris corner measure for each extremum in the DoG pyramid and remove the blob if the measure is below the threshold set by the variable, HarrisThreshold.

Figure 4 illustrates all blobs unfiltered (left) and the remaining blobs after edge removal (right).

Exercise 2.3: SIFT

Your task here is to stitch (combine) two aerial photos into one image using SIFT-based feature matching.

Open Exercise2.3.m and change the variable *VLFEATROOT* to the unpacked path of VLFeat at your hard drive.

Run the code and inspect the figures carefully.

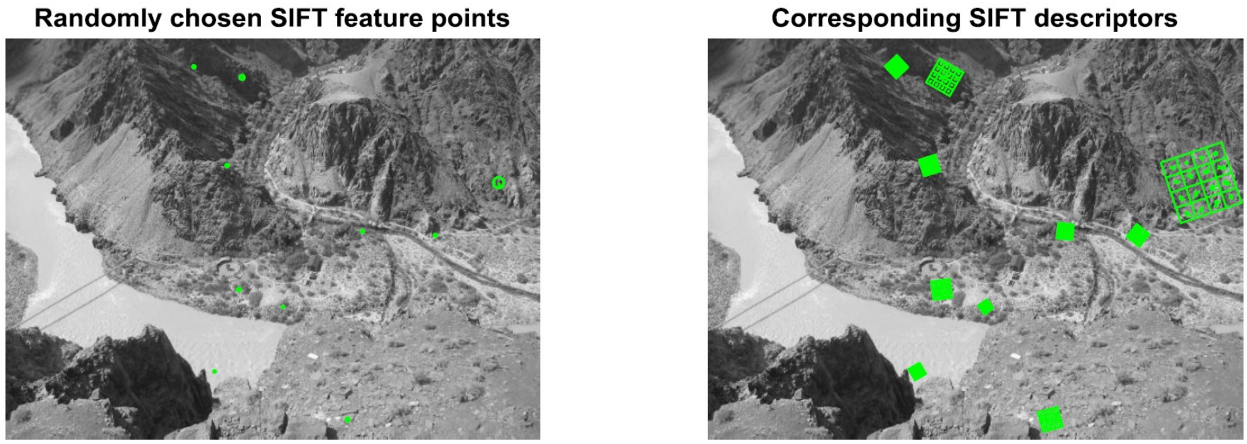


Figure 1 illustrates 10 randomly chosen SIFT feature points, their scales, and their corresponding SIFT descriptors.

Figure 2 illustrates all the matched features between the two images. How can these be used to stitch (combine) the images into one?

In Figure 3, this is done using all matches for estimating a global transformation from one image to the other. A global transformation maps a point p in one image to a point p' in the other image:

$$\boldsymbol{p}' = T(\boldsymbol{p})$$

Why does this procedure fail?

In Figure 4, a concept known as RANSAC (Random Sample Consensus) finds a subset of feature point matches that agree about a global transformation. In Figure 5, this subset is used to estimate a new and better transformation.

Try using your own images to make a panorama of two images. When will this fail?

Exercise 2.4: Patch descriptors

Open Exercise2.4.m and change the variable *VLFEATROOT* to the unpacked path of VLFeat at your hard drive.

Run the code and try to explain what Figure 2 and 3 illustrate. It may help to set the number of visualized features (nFeatures) to 1.

Hint: See the CovDet tutorial http://www.vlfeat.org/overview/covdet.html

The code in Exercise2.4.m uses the default (SIFT) to detect feature points. If time allows, replace the SIFT by another feature detection algorithm.