Computer Vision and Image Analysis
Assignment Sheet 5 - 18.11.2014

Next exercise group: 21.11.2014

- Deadline for exercise: **27.11.2014, 18:00**, hard deadline.
  Please submit solutions either on our Ilias web page (preferred method), or via e-Mail in a single ZIP-Archive to `ole.johannsen@uni-konstanz.de`.
- You may work in groups of up to three students, make sure all participants are clearly mentioned or assigned to the submission in Ilias.

> Note: (1) in contrast to what I said in the lecture, I have moved the SIFT library exercise to next week, since we only discuss the algorithm fully on Friday. Instead, I believe it is for now more useful to get a better idea of the LoG or DoG scale space and blob detection. (2) Exercise 5.1 is somewhat different from what you are used to. If you think it's stupid or way too much effort, just skip it.

Exercise 5.1 (the SIFT paper, 10 points)

At some point in his education, every student should try to delve into primary sources (i.e. research papers) in addition to textbooks. If you haven't yet, the SIFT paper is a good starting point, since it is quite a famous one and will be explained in detail in the lecture.

(a) Read the SIFT paper (provided under 'Additional material' on Ilias as `lowe_sift_2004.pdf`) at least casually, and try to understand as much of what is going on as you can. Relate it to what you have learned in the lecture and see on the slides.

It might be a good idea to try this exercise before Friday, when we continue the review of the paper - it is not necessary that you already understand everything.

If you wish, you may skip Section 2 (which is very hard to understand without lots of background) as well as everything related to object detection, in particular Sections 7 and 8, which I will not talk about in the lecture.

(b) Explain the following in your own words (or formulas). Don't write essays, three-five sentences should easily be sufficient for each point.

(i) If an octave is divided into s intervals, why are there $s + 3$ filtered images which need to be computed for the octave, as claimed on page 7? Explain it with a short example: if the octave covers the range $[\sigma, 2\sigma]$ and $s = 2$, at which scales exactly are the 5 images computed?

(ii) Is it always better to use larger values of s, i.e. a finer and more accurate discretization of scale space? What are the effects, and which trade-offs are there?

(iii) Page 11 tells us that we need to solve a 3×3 linear system in order to compute the refined location $\hat{\mathbf{x}}$ of the scale-space extremum. What exactly is this linear system, i.e. what are the entries of the 3×3 matrix A and vector b when the system is written as $A\hat{\mathbf{x}} = b$?

(iv) Suppose that as usual, when using a spatial Gaussian weighting kernel, that it is cut off when the distance from the center becomes larger than three standard deviations. A feature is detected at scale $\sigma = 1$ pixel. How many sample pixels, i.e. measurements of orientation, contribute to the angular histogram when estimating feature orientation?

(v) How many entries has the feature vector for the descriptor as shown in figure 7? How many for the actual descriptor used in the experiments?

Exercise 5.2 (blob detection using Difference of Gaussians in scale space, 10+5+5 points)

(a) The sample code for the exercise has a skeleton script `blobs.m`, which contains an outline of what you have to do to detect the scale space extrema corresponding to blob-shaped features (as used in SIFT). The code also visualizes the features in the original image, and shows off a really cool visualizion of scale space. Complete the missing parts of the code to get a detection result which looks roughly like this.

(b) Use the code to compute the scale space for the synthetic blob image provided in `blobs.png`. Repeat the blob detection above for this sample image. Verify how accurate the estimates for the circle radii are, and try to improve them by making code adjustments, e.g. changing the factor for the radius or selecting different scale space parameters.

(c) If you have implemented the code exactly as asked for in the comments, you will only detect dark blobs. Change the code so that it detects bright as well as dark blobs, and demonstrate the difference on an example image.