Computer Vision and Image Analysis
Assignment Sheet 3 - 04.11.2014

Next exercise group: 07.11.2014

- Deadline for exercise: **11.11.2014, 13:30**, 30% point loss per day late.
  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.

Exercise 3.1 (structure tensor, 10+5 points)

(a) Write a script to compute the structure tensor of an image. To reduce your workload, the following functions are provided with the exercise sheet:

- `dgx.m, dgy.m` compute Derivative of Gaussian kernels to be used with `imfilter`.
- `st_eigensystem.m` computes Eigenvalues and the Eigenvector corresponding to the larger Eigenvalue (direction across the edge) from the structure tensor components.

Compute the Eigenvalues and Eigenvectors of the structure tensor for a checkerboard pattern and another image of your choice, and visualize both similar to the examples given in the lecture (useful Matlab functions: `checkerboard`, `quiver`).

(b) Write another function to compute the Harris corner detector response given either the Eigenvalues or the structure tensor components directly (your choice). Visualize the response function. Then, perform non-maxima suppression - for this, we have provided another script `non_maxima_suppression.m` -, threshold the result and visualize the corners in the image. Try to achieve a result which looks similar to the examples in the lecture. For convenience, we provide the function `visualize_mask.m` which can mark the detected locations in the input image for visual feedback.

Exercise 3.2 (normalized cross-correlation, 10 +5 points)

(a) Write a function to compute the normalized cross-correlation between two images (should be a relatively simple modification of the convolution code using for-loops, if you do not care about speed).

Combine it with e.g. non-maxima suppression and thresholding (see exercise 2.1) to detect the pebble `pebble.png` in the image `beach.png` (actually it is sufficient to just find the location of the global maximum).

(b) In `pebble_masked.png` you again find a pebble, but this time, its background has been removed. In `pebble_mask.png`, you will find a binary image for the foreground (white pixels).

Modify the cross-correlation function so that it is able to deal with this situation, then implement it. Detect the second pebble in the image using the method above and the modified function.