Assignment 1

This exercise is part of the course assignment. **Deadline for the assignment 30.11.2022 at 23:59**

The topic of this assignment is multiresolution analysis and image blending. The corresponding chapters in the book are 3.5.3 and 3.5.5. For this assignment you should return

- The files gauss_pyramid.m, laplace_pyramid.m, from_laplacian.m and main1.m. Each of these files should contain your name and student number (of both students if you work in pairs).
- Your answers to the questions in the analysis part. At the end of the course, you should return a single pdf containing the answers to all questions of the assignments. The report should contain also your name and student number (of both students if you work in pairs).

Coding part (5p)

Before you start coding, run the script demo.p, to get a taste of the final result. Your tasks are:

1. Implement the function gauss_pyramid(I, N), which takes an image I (grayscale or color) and computes the Gaussian pyramid with N levels, where the first level is the original image. At each level the image size is halved. (**Hint:** matlab function imresize). The output of the function should be a cell array of length N. (**Hint:** doc cell to read the documentation of matlab cell arrays.)
2. Implement the function laplace_pyramid(I, N), which computes the Lapalce pyramid with N levels from the image I. The last level of the pyramid should be the same of the Gaussian pyramid. Again, the function should work for both color and grayscale images and the output should be a cell array of length N. (**Hint:** use the function you implemented in the previous step. Again, imresize is your friend)
3. Implement the function from_laplacian(pyr), which takes as input a Laplace pyramid and reconstructs the original image from it.
4. When you are done, open the script main.m and complete the missing parts.

(a) Create the mask orange_mask, which extracts the right side of the image and sets the left size to zero. Similarly, create the mask apple_mask, which extracts the left side of the image and sets the right to zero.

(b) Perform naive blending, i.e. compute the weighted sum of the images.

(c) Using the functions you have implemented before, compute the laplacian pyramids of the two images.

(d) Compute the Gaussian pyramids of the two masks.

(e) Create a new cell blended_pyramid, which has the same size of the pyramids you created before.

(f) At each level, blend the corresponding levels in the laplacian pyramids, using the corresponding masks from the gaussian pyramids.

(g) Reconstruct the blended image from the blended_pyramid.

(h) Visualize the results

Note! You are given the files sol_gauss_pyramid.p, sol_laplace_pyramid.p, sol_from_laplacian.p which contain the model solutions of the corresponding functions. These functions can be called but the source code is hidden. These might be helpful in debugging your implementation.

Analysis part (5p)

Explain how pyramid blending (steps d-g in the previous section) works and why it performs better than naive blending. Your report should contain the answers to the following questions

- What information is contained in the laplace pyramid and what information is contained in the Gaussian pyramid?
- Why do we take the Laplace pyramid of the image but the Gaussian pyramid of the mask?
- What are the advantages of performing blending at multiple resolution levels? In the main script, try different values for the number of levels, what do you observe? (**Hint:** try values between 1 and 10)